

НАСТАВНИ ПЛАН И ПРОГРАМ ЗА ГИМНАЗИЈУ ЗА УЧЕНИКЕ СА ПОСЕБНИМ СПОСОБНОСТИМА ЗА РАЧУНАРСТВО И ИНФОРМАТИКУ

/ **нолошка компетенција**

ЦИЉЕВИ ОБРАЗОВАЊА И ВАСПИТАЊА ЗА ГИМНАЗИЈУ ЗА УЧЕНИКЕ СА ПОСЕБНИМ СПОСОБНОСТИМА ЗА РАЧУНАРСТВО И ИНФОРМАТИКУ

Циљ образовања и васпитања за гимназију за ученике са посебним способностима за рачунарство и информатику је да се путем стицања функционалних знања, овладавања вештинама, формирања ставова и вредности, у оквиру предвиђених наставних предмета, обезбеди:

– пун интелектуални, емоционални, социјални, морални и физички развој ученика у складу са њиховим способностима, потребама, интересовањима;

– стицање знања, овладавање вештинама и формирање вредносних ставова који доприносе развоју информатичке писмености неопходне за даље школовање, живот и рад у савременом друштву;

– унапређивање језичке, уметничке, културне, научне, техничке и математичке писмености, што је неопходно за наставак образовања и професионални развој;

– подршка развоју међупредметних компетенција.

Међупредметне компетенције обезбеђују ученицима успешно сналажење у свакодневном животу и раду. Оне су резултат великог броја активности које се остварују у свим наставним предметима и током целокупног гимназијског школовања. За њихов развој важни су не само садржаји већ и методе којима се оне проучавају – требало би да доминирају истраживачке и интерактивне методе, рефлексивна, критичко мишљење, кооперативно учење и друге методе које јачају учешће ученика у образовно-васпитном процесу.

Компетенција комуникације на српском (или матерњем) језику

Ученик је у стању да разуме, користи и критички размишља о идејама, чињеницама, осећањима и ставовима које изражавају други на српском (матерњем) језику у усменом и писаном облику. У стању је да изрази и тумачи сопствене мисли, ставове, осећања у усменом и писаном облику на српском (матерњем) језику. Ученик користи српски (матерњи) језик да би учествовао у различитим друштвеним и културним ситуацијама на примерен, конструктиван и креативан начин.

Компетенција комуникације на страном језику

Ученик је у стању да разуме, користи и критички размишља о идејама, чињеницама, осећањима и ставовима које изражавају други на страном језику у усменом и писаном облику. У стању је да изрази и тумачи сопствене мисли, ставове, осећања у усменом и писаном облику на страном језику. Ученик користи страни језик да би учествовао у различитим друштвеним и културним ситуацијама на примерен, конструктиван и креативан начин.

Информатичка компетенција

Ученик компетентно и критички користи технологије информационог друштва у разноврсним ситуацијама и активностима да би остварио личне циљеве и активности, конструктивно и промишљено учествовао у животу заједнице. На адекватан начин користи предности рачунара и друштвених мрежа у удруживању са другима и покретању акција чији је циљ ширење корисних информација или пружање помоћи и подршке онима којима је то потребно, градећи при томе правилне ставове према коришћењу рачунара, без злоупотребе и претеривања које угрожава физичко и ментално здравље.

Математичка компетенција

Ученик је у стању да користи, опише и објасни математичке појмове и процедуре, предвиди појаве, доноси математички засноване одлуке и решава проблеме у различитим ситуацијама. Ученик може да идентификује и разуме улогу коју математика има у савременом животу и активно, конструктивно и критички учествује у животу заједнице и доприноси њеном развоју.

Ученик је у стању да користи и примењује знање и научне методе при препознавању научних проблема, формулише научна објашњења појава и изводи на чињеницама засноване закључке о научно релевантним питањима. Ученик разуме науку као најзначајнији облик људског сазнања и делатности за развој друштва, поседује свест о начинима на који наука и технологија обликују живот у савременом друштву и разуме колико је важно имати одговоран однос према употреби научних достигнућа за данашње и будуће генерације.

Компетенција за решавање проблема

Ученик је у стању да препозна, разуме и реши проблемске ситуације у којима решење није видљиво на први поглед, користи знања и вештине стечене из различитих предмета. Решавање проблема подразумева и спремност ученика да се ангажује и конструктивно и промишљено допринесе решавању проблема са којима се суочава заједница којој припада.

Компетенција за учење

Ослањајући се на претходна знања и искуства, ученик је у стању да организује учење, самостално или у групи, на ефикасан начин и у складу са сопственим потребама. Ученик је свестан начина на који учи и расположивих ресурса за учење (књиге, интернет, друге особе, итд.), мотивисан је да учи, може да управља процесом учења и превазилази тешкоће са којима се суочава током учења.

Компетенција за одговоран однос према здрављу

Ученик је свестан краткорочних и дугорочних, последица понашања које угрожава физичко и ментално здравље особе, има изграђене здраве животне стилове и спреман је да се понаша у складу са њима.

Компетенција за управљање сопственим понашањем и спремност за акцију

Ученик је у стању да идеје преточи у акцију. То подразумева критичко мишљење, креативност, иновативност и преузимање ризика, као и способност планирања и управљања активношћу, било самостално или у оквиру тима, ради остварења циљева. Ученик разуме контекст у коме се одвијају активности, уме да се усклади са њим и искористи могућности које се нуде.

Социјална компетенција (сарадња, тимски рад)

Ова врста компетенције омогућава ученицима да делотворно и конструктивно учествују у друштвеном животу, неугрожавајући интересе, потребе, осећања и права других, чиме се јача друштвена кохезија у Србији. Ученик уме да сарађује и тимски ради на решавању различитих проблема који су од значаја за њега и ширу друштвену заједницу.

Грађанска компетенција за живот у демократском друштву

Ученик разуме основне појмове, институције, процедуре, принципе и вредности демократског друштва, свестан је места и улоге грађанина за развој и одржање демократског друштва, разуме изазове са којима се суочава савремено друштво и држава и спреман је да активно и конструктивно учествује у различитим сферама друштвеног живота.

Културна компетенција

Ученик разуме значај који култура има за развој и просперитет појединца и друштва, у стању је да интерпретира културна дела и повеже их са својим личним искуством да би га осмислио и обогатио. Ученик разуме значај креативног изражавања идеја, мисли, осећања у различитим медијима (музика, извођачке уметности, књижевност и визуелне уметности).

Компетенција за одрживи развој

Ученик има свест да задовољење потреба садашњице не сме да угрози могућност будућих генерација да задовоље своје потребе, свестан је важности очувања животне околине, и спреман је да се ангажује на конструктиван и креативан начин у очувању животне околине кроз самосталну активност и укључивање у различите друштвене акције.

ПЛАН НАСТАВЕ И УЧЕЊА ЗА ГИМНАЗИЈУ ЗА УЧЕНИКЕ СА ПОСЕБНИМ СПОСОБНОСТИМА ЗА РАЧУНАРСТВО И ИНФОРМАТИКУ

		I РАЗРЕД				II РАЗРЕД				III РАЗРЕД				IV РАЗРЕД				УКУПНО		
		нед.		год.		нед.		год.		нед.		год.		нед.		год.		Т	В	Σ
		Т	В	Т	В	Т	В	Т	В	Т	В	Т	В	Т	В					
I ОБАВЕЗНИ ПРЕДМЕТИ		27	5	999	185	27	5	999	185	23	9	851	333	25	7	825	231	3674	934	4608
1.	Српски језик и књижевност	4		148		3		111		3		111		4		132		502		502
1.1.	_____језик и књижевност*	4		148		3		111		3		111		4		132		502		502
2.	Српски као нематерњи језик*	2		74		2		74		2		74		2		66		288		288
3.	Страни језик	2		74		2		74		2		74		2		66		288		288
		I РАЗРЕД				II РАЗРЕД				III РАЗРЕД				IV РАЗРЕД				УКУПНО		
		нед.		год.		нед.		год.		нед.		год.		нед.		год.				
		Т	В	Т	В	Т	В	Т	В	Т	В	Т	В	Т	В	Т	В	Т	В	Σ
4.	Историја	2		74		2		74										148		148
5.	Географија	2		74		2		74										148		148
6.	Музичка култура	1		37														37		37
7.	Физика	2		74		3		111		3		111		4		132		428		428
8.	Хемија	2		74		2		74										148		148
9.	Физичко васпитање	2		74		2		74		2		74		2		66		288		288
10.	Математика	5		185		5		185		5		185		5		165		720		720
11.	Дискретна математика					2		74										74		74
12.	Биологија									3		111		3		99		210		210
13.	Психологија									2		74						74		74
14.	Ликовна култура									1		37						37		37
15.	Социологија													2		66		66		66
16.	Филозофија													2		66		66		66
17.	Примена рачунара	1	2	37	74		2		74		2		74					37	222	259
18.	Програмирање	2	3	74	111	2	3	74	111		3		111					148	333	481
19.	Рачунарске системи	2		74														74		74
20.	Оперативни системи и рачунарске мреже					2		74										74		74
21.	Објектно оријентисано програмирање									1	3	37	111					37	111	148
22.	Базе података									1	1	37	37	1	2	33	66	70	103	173
23.	Програмске парадигме													3		99		99		99
24.	Веб програмирање													2		66		66		66
II: ИЗБОРНИ ПРОГРАМИ		1		37		1		37		1		37		1		33		144		144
1.	Грађанско васпитање / Верска настава	1		37		1		37		1		37		1		33		144		144
УКУПНО I+II		33		1221		33		1221		33		1221		33		1089		4752		4752

* За ученике који наставу слушају на матерњем језику националне мањине

Облици образовно-васпитног рада којима се остварују обавезни предмети, изборни програми и активности

ОБЛИК ОБРАЗОВНО-ВАСПИТНОГ РАДА	ПРВИ РАЗРЕД	ДРУГИ РАЗРЕД	ТРЕЋИ РАЗРЕД	ЧЕТВРТИ РАЗРЕД	УКУПНО
Час одељенског старешине	74 часа	74 часа	74 часа	66 часова	288 часова
Додатни рад *	до 30 часова	до 30 часова	до 30 часова	до 30 часова	до 120 часова
Допунски рад *	до 30 часова	до 30 часова	до 30 часова	до 30 часова	до 120 часова
Припремни рад *	до 30 часова	до 30 часова	до 30 часова	до 30 часова	до 120 часова

* Ако се укаже потреба за овим облицима рада

ОСТАЛИ ОБЛИЦИ ОБРАЗОВНО-ВАСПИТНОГ РАДА	I РАЗРЕД	II РАЗРЕД	III РАЗРЕД	IV РАЗРЕД
Екскурзија	до 3 дана	до 5 дана	до 5 наставних дана	до 5 наставних дана
Језик другог народа или националне мањине са елементима националне културе	2 часа недељно			
Други страни језик	2 часа недељно			
Слободне активности (хор, оркестар, секције, техничке, хуманитарне, спортско-рекреативне и друге ваннаставне активности)	30–60 часова годишње			
Друштвене активности – ученички парламент, ученичке задруге	15–30 часова годишње			

Остваривање плана и програма наставе и учења

1. Распоред радних недеља у току године

	I РАЗРЕД	II РАЗРЕД	III РАЗРЕД	IV РАЗРЕД
Разредно-часовна настава	37	37	37	33
Обавезне ваннаставне активности	2	2	2	2
Матурски испит				4
Укупно радних недеља	39	39	39	39

2. Подела одељења на групе ученика

Предмет	I разред	II разред	III разред	IV разред	Број ученика у групи
	Број часова вежби	Број часова вежби	Број часова вежби	Број часова вежби	
1. Примена рачунара	74	74	74		8–12
2. Програмирање	111	111	111		8–12
3. Објектно оријентисано програмирање			111		8–12
4. Базе података			37	66	8–12
5. Програмске парадигме				99	8–12
6. Веб програмирање				66	8–12

НАСТАВНИ ПРОГРАМ

Наставни план и програм остварује се у складу са:

1. Правилником о наставном плану и програму за гимназију („Службени гласник СРС – Просветни гласник”, број 5/90 и „Просветни гласник”, бр. 3/91, 3/92, 17/93, 2/94, 2/95, 8/95, 23/97, 2/02, 5/03, 10/03, 11/04, 18/04, 24/04, 3/05, 11/05, 2/06, 6/06, 12/06, 17/06, 1/08, 8/08, 1/09, 3/09, 10/09, 5/10, 7/11, 4/13, 14/13, 17/13, 18/13, 5/14, 4/15, 18/15, 11/16 и 13/16), у делу који се односи на природно-математички смер гимназије, са наставним планом и програмом за предмете:

- 1) српски језик и књижевност;
- 2) српски као нематерњи језик;
- 3) страни језик;
- 4) физика (први, други и трећи разред);
- 5) психологија;
- 6) филозофија;
- 7) грађанско васпитање;

и у делу који се односи на друштвено-језички смер гимназије, са наставним планом и програмом за предмет:

1) хемија;

2. Правилником о наставном плану и програму предмета верска настава („Просветни гласник”, бр. 6/03, 23/04, 9/05 и 11/16);

3. Правилником о наставном плану и програму за обдарене ученике у Математичкој гимназији („Службени гласник Републике Србије – Просветни гласник”, бр. 12/16 и 13/16), и то са планом и програмом предмета:

- 1) историја;
- 2) географија;
- 3) биологија;
- 4) социологија са правима грађана.

Уколико се наставни план и програм остварује на језику националне мањине, програм језика и књижевности националне мањине остварује се у складу са Правилником о наставном плану и програму за гимназију („Службени гласник СРС – Просветни гласник”, број 5/90 и „Просветни гласник”, бр. 3/91, 3/92, 17/93, 2/94, 2/95, 8/95, 23/97, 2/02, 5/03, 10/03, 11/04, 18/04, 24/04, 3/05, 11/05, 2/06, 6/06, 12/06, 17/06, 1/08, 8/08, 1/09, 3/09, 10/09, 5/10, 7/11, 4/13, 14/13, 17/13, 18/13, 5/14, 4/15, 18/15, 11/16 и 13/16) за природно-математички смер гимназије.

Годишњи број часова преузетих програма из Правилника о наставном плану и програму за гимназију и Правилника о наставном плану и програму за обдарене ученике у Математичкој гимназији треба прилагодити наставном плану за гимназију за ученике са посебним способностима за рачунарство и информатику.

МУЗИЧКА КУЛТУРА

Циљ и задаци

Циљ наставе предмета Музичка култура је развијање свести о значају и улози музичке уметности кроз развој цивилизације и друштва, неговање традиције, унапређивање естетских критеријума уз подстицање и неговање интересовања, навика и потреба за слушање вредних музичких остварења

Задаци наставе предмета Музичка култура су да ученици:

- стекну знања о музици различитих епоха;
- развију навику слушања музике, и способност за разумевање музике;
- развију креативност у свим музичким активностима (извођење, слушање, истраживање и стварање музике);
- упознају изражајна средстава музичке уметности и њихову корелацију са карактером музичких дела;
- препознају инструменталне саставе и инструменте по боји звука;
- разумеју корелације између избора извођачког састава и садржаја музике.
- развију способност за самостално истраживање у области музике;

И разред

(1 час недељно, 36 часова годишње)

САДРЖАЈИ ПРОГРАМА

Улога музике у животу појединца и друштвене заједнице.

Музика у првобитној друштвеној заједници и културама старог века

Корени музике и њене првобитне улоге. Музика у животу старих источних народа, Грчке и Рима.

Примери за слушање: Псалми, напитунице...

Музика средњег века

Стара хришћанска музика: грегоријански корал и византијско певање. Световна музика (трубадури, трувери...). Рани облици вишегласја. Почеци духовне и световне музике код нас.

Примери за слушање: грегоријански корал; византијско певање: Рамбо де Вакеира – Календа маја; Кир Стефан Србин – Ниња сили.

Музика ренесансе

Развој духовног и световног вокалног вишегласја: мотет, миса и мадригал. Највећи представници вокалне музике: Г. П. Палестрина, Орландо ди Ласо, Ј. П. Галус.

Примери за слушање:

Г. П. Палестрина – одломак из Мисе папе Марчела; Орландо ди Ласо – Мадона миа кара; Ј. П. Галус – Ево како умире праведник.

Музика барока и рококоа

Појава опере, њен развој и најистакнутији представници: Пери, К. Монтеверди, Ж. Б. Лили и Х. Персел. Комична опера – Г. Б. Перголези и Глукова реформа.

Развој инструменталне и вокално-инструменталне музике у бароку и њени главни облици: свита, соната, концерт, fuga, кантата, ораторијум, пасија. Најважнији представници: А. Корели, А. Вивалди, Ј. С. Бах, Ј. Хендл, Д. Скарлати, Ф. Купрен.

Примери за слушање:

К. Монтеверди – Аријаднина тужбалица (*Ласциате ми мори-ре*); Корели – *Ла фолиа*; А. Вивалди – Годишња доба (по избору); Бах – Бранденбуршки концерт (по избору); Токата и fuga де-мол, за оргуље; завршни хор из Пасије по Матеји; Хендл – арија (Ларго) из опере Ксеркс; Алилуја из ораторијума Месија, Д. Скарлати – Соната (по избору); Ф. Купрен – Жетеоци; Перголези – Арија Серпине из опере Служавка господарица; Глук – арија Орфеја из опере Орфеј и Еуридика.

Музика преткласике и класике

Развој сонате и симфоније. Најистакнутији представници бечке класике – Ј. Хајдн, В. А. Моцарт и Л. ван Бетовен.

Примери за слушање:

Ј. Хајдн – Симфонија са ударцем тимпана, II став; Лондонска симфонија, Де-дур број 104, I став; В. А. Моцарт – Симфонија ге-мол, I став; Мала ноћна музика; увертира и нека арија из опере Фигарова женидба; Л. ван Бетовен – Соната цис-мол. – Месечина; Клавирски концерт број 3, це-мол, III став; Ода радости из 9. симфоније; 5. симфонија, I став.

Музика 19. и 20. века

Романтизам у музици

Општа обележја романтизма у музици. Соло-песма, клавирска минијатура, програмска музика.

Примери за слушање:

Ф. Шуберт – Недовршена симфонија, I став; соло-песме Пастрмка и Вилењак; Ф. Менделсон – Песма без речи; Виолински концерт е-мол, I став; Р. Шуман – Лептири; Ф. Шопен – Полонеза Ас-дур; мазурка по избору; Соната бе-мол, II и III став; Х. Берлиоз – Фантастична симфонија, II став; Ф. Лист – симфонијска поема Прелиди; Мефисто-валцер; Ј. Брамс – Виолински концерт, III став; манарска игра по избору; 3. симфонија, Еф-дур, III став; Р. Штраус – Тил Ојленшпигл.

Опера, оперета и балет у 19. и 20. веку

Развој опере у Италији (Ђ. Росини, В. Белини, Ђ. Верди), Немачкој (К. М. Вебер, Р. Вагнер) и Француској (Ч. Гуно, Ж. Бизе, Л. Делиб), Г. Пучини и веристи. Оперета и балет (Ж. Офенбах, Ј. Штраус, А. Адолф...).

Примери за слушање:

Ђ. Росини – увертира и арија Фигара из опере Севилски берберин; В. Белини – арија Каста дива из опере Норма; Ђ. Верди – Хор Јевреја из опере Набуко; Квартет из IV чина опере Риголето; тријумфална сцена из опере Аида; Вебер – увертира за оперу Чаробни стрелац; Р. Вагнер – увертира и хор морнара из опере Холанђанин луталица; свадбени хор из Лоенгрин; Кас Валкира из музичке драме Валкира; Ч. Гуно – балетска музика из опере Фауст; Ж. Бизе – хабанера и арија тореадора из опере Кармен; Л. Делиб – одломак из балета Копелија; Ж. Офенбах – баркарола из

Хофманових прича; J. Штраус – увертира Слепи миш. Г. Пучини: Тоска (молитва Тоске из II чина, арија Каварадосија из III чина), Боми (завршни дует из I чина); P. Леонкавало: пролог из Пајаца; П. Маскањи: Кавалерија рустикана (Интермецо).

Националне школе

Развој националних школа код руса (Руска петорка, П. И. Чајковски); Чеха (Б. Б. Сметана, А. Дворжак); Срба (К. Станковић, Ј. Маринковић, С. Мокрањац), Шпанаца, Скандинаваца...

Примери за слушање:

М. И. Глинка – увертира за оперу Руслан и Лјудмила; А. Бородин – Половјецке игре из опере Кнез Игор; М. П. Мусоргски – смрт Бориса из Бориса Годунова; одломци из Слика са изложбе: Н. Римски – Корсаков – I став из свите Шехерезада; П. И. Чајковски – V симфонија, II став; б. симфонија, I и IV став; Клавирски концерт бе-мол I став; одломци из балета Лабудово језеро; арија Ленског из III чина опере Егвеније Оњегин, и сцена Татјаниног писма из II чина; увертира 181; Б. Сметана – Вишехрад из циклуса Моја домовина; увертира за оперу Продана невеста; А. Дворжак – Симфонија из Новог света, III став; Концерт за виолончело ха-мол, I став; Словенска игра (по избору).

К. Станковић: Варијације на песму Што се боре мисли моје, Српске народне песме (избор); Ј. Маринковић: Грм, Молитва, Чезња; С. Мокрањац: V и X руковет, Козар, Њест свјат (из Опела), Херувимска песма (из Литургије). Е. Григ – Клавирски концерт а-мол, I став; I свита музике за драму Пер Гинт.

Импесионизам

Упознавање основних особености импесионизма у музици, као и њихових главних представника (К. Дебиси и М. Равел) и њихових дела.

Избор дела за слушање:

К. Дебиси: Месечина, Арабеске; М. Равел: Болеро, Дафнис и Клое (одломци).

Музика XX века

Стилски правци: експресионизам, додекафонија, неокласицизам. Представници: И. Стравински, А. Шенберг, П. Хиндемит, С. Прокофјев, Д. Шостакович, Б. Бритн и Б. Барток.

Избор дела за слушање:

А. Шенберг: пет комада за клавир, Пјеро месечар; С. Прокофјев: Класична симфонија, Ромео и Јулија (одломци); Б. Бритн: Једноставна симфонија; Д. Шостакович: V симфонија, Лењинградска симфонија; И. Стравински: Посвећење пролећа (одломак), Петрушка (руска игра); Б. Барток: Концерт за оркестар (став), Гудачки квартети (избор); К. Орф: Кармина бурана.

Музичко стваралаштво и музички живот Србије од XX века

Почеци развоја модерне музике – П. Коњовић, М. Милојевић, С. Христић, В. Мокрањац, Деспих, К. Бабић, Љ. Марић, И. Жебељан.

Избор дела за слушање:

П. Коњовић: Триптихон из Коштана, Нане кажи тајку – из збирке Лирика; М. Милојевић: Четири комада за клавир (избор), Јесења елегија, Јапан – соло песме, Легенда о Јефимији за виолончело и клавир; С. Христић: Прва свита из Охридске легенде, Елегија, Поноћ, Вече на шкољу – соло-песме; Ј. Славенски: Вода звира, Балканофонија (одломци), Симфонија Оријента (одломци), Други гудачки квартет (Лирски); М. Тајчевић: Седам балканских игара (избор), Воспојте (из Четири духовна стиха); Љ. Марић: Песме простора (одломци); В. Мокрањац: IV симфонија (одломак), Етиде за клавир (избор) Одједи, Соната за виолину и клавир (IV став); Д. Деспих: Хумористичке етиде; К. Бабић: Хорске композиције (избор).

Жанрови популарне музике (цез, мјузикл, рок, поп и народна песма)

Представници: Џ. Гершвин, Л. Бернштајн, Ђ. К. Меноти...

Примењена музика: Филмска исцесна музика

Хор и оркестар

Школа је обавезна да организује часове хора 3 часа недељно. Ако у школи постоје услови, треба организовати школски оркестар. Часови хора и оркестра се изводе континуирано од почетка до краја школске године.

Часови хора и оркестра као редовна настава улазе у фонд часова наставника музичке културе.

Хор се организује у групи од 30 до 60 ученика од I до IV разреда. Хорске пробе се изводе одвојено по гласовима и заједно. Програм рада са хором треба да садржи пригодне композиције, као и дела озбиљније уметничке вредности, у зависности од могућности ансамбла.

Садржај рада:

- одабир и разврставање гласова;
- вежбе дисања, дикције и интонације;
- хорско распевавање и техничке вежбе;
- обрада композиције;
- увежбавање хорских деоница по гласовима и заједно;
- остваривање програма и реализација наступа према годишњем плану школе.

Оркестар је инструментални састав од најмање 10 извођача који свирају најмање три самосталне деонице. Могу се образовати оркестри гудачког састава, хармоника, мандолина, тамбурица, блокфлаута као и мешовити.

Садржај рада:

- избор инструмената и извођача у формирању оркестра;
- избор композиција према могућностима извођача и саставу оркестра;
- техничке и интонативне вежбе;
- расписивање деоница и увежбавање по групама (прстOMET, интонација, фразирање); спајање по групама;
- заједничко свирање целог оркестра, ритмичко-интонативно и стилско обликовање композиције.

Музичка секција

У оквиру музичке секције може се организовати рад са ученицима у оквиру камерне музике. Камерни састави се образују од 2 до 9 извођача, певача, инструменталиста или комбиновано. Садржај рада ових састава бира наставник у сарадњи са заинтересованим ученицима, а према извођачким способностима, склоностима и интересовању ученика.

Додатна настава

На часовима додатне наставе остварују се садржаји за које ученици показују посебно интересовање. Слушају се нова музичка дела (или понављају позната) из области које се обрађују на редовној настави или су моментално актуелна у уметничком животу. Поједини ученици могу по слободном избору, уз консултовање и помоћ наставника да припреме литерарне саставе о слушаним делима. У оквирима додатне наставе предмета *Музичка култура* организују се и концерти у школи, заједничке посете оперским и балетским представама, концертима и пројекцијама музичких филмова. Једна представа или концерт ван школе, рачунају се као 4 часа додатне наставе.

НАЧИН ОСТВАРИВАЊА ПРОГРАМА

Програм предмета Музичка култура је конципиран према историјско-стилским раздобљима које у излагању треба ограничити на најбитније елементе, а слушању музике дати примарно место. Како се предмет Музичка култура у Рачунарској гимназији изучава само једну годину, препорука је да се највише времена посвети слушању најзначајнијих представника музичких стилова, односно репрезентативних музичких примера. Наставничко усмено излагање треба да послужи као информација и инспирација за слушање музике које је централни део часа (20 одсто излагања,

60 одсто слушања – у фрагментима или у целини, у зависности од примера, или 20 одсто закључивања – разговор са ученицима или понављање слушања одређених делова композиције или целине). Директним, сугестивним и интересантним излагањем наставник треба да оспособи ученике да активно слушају музику, како би могли на одабраним примерима да развијају музичку меморију, естетски укус, препознају дела и инструменте, осећају музичку форму, значај и карактеристике стила и композиторовог стваралаштва. Пре слушања музике ученицима треба поставити јасне задатке. У програму је наведено много више музичких дела него што је могуће саслушати и усвојити. Због тога из наведених примера наставник прави сопствени избор који ће ученици саслушати и усвојити као примере за одређени правац, епоху итд. Код слушања музике настојати да се одабере пример који може да се слуша у целини (један цео став, краћу увертуру...), да би ученици могли да доживе дело као целину и схвате музичку форму. Епохе се представљају у корелацији са предметима сродних садржаја (књижевност, ликовна уметност, историја итд.). Ученике треба подстицати да посећују концертне дворане, оперу... као и да анализирају своје утиске и расположења. Поједине наставне теме могу се обрадити ученичким рефератима, у којима примарно место имају музички примери. Кроз ваннаставне активности и факултативну наставу могу се врло успешно осмислити и организовати разноврсни видови музичких активности који су значајан допринос културној и јавној делатности школе и личном усавршавању ученика. Оцењивањем се сагледава знање ученика, његово ангажовање, креативност и заинтересованост.

За реализацију садржаја програма нужно је обезбедити услове за рад који подразумевају опремљен кабинет (специјализовану учионицу) са клавиром (или другим основним инструментом) и савременим квалитетним аудиовизуелним средствима, компјутером и сталном интернет везом.

ФИЗИКА

Циљ и задаци

Циљ наставе предмета Физика јесте стицање функционалне писмености (природно-научне, математичке, техничке), систематско стицање знања о физичким појавама и процесима и њихово разумевање на основу физичких модела и теорија, оспособљавање ученика за примену знања и решавање проблема и задатака у новим и непознатим ситуацијама, активно стицање знања о физичким појавама кроз истраживачки приступ, стицање радних навика, одговорности и способности за самосталан рад и за тимски рад, формирање основе за даље образовање.

Задатак наставе предмета Физика јесте стварање разноврсних могућности да кроз различите садржаје и облике рада, применом савремених методичких и дидактичких поступака у настави, циљеви и задаци образовања у целини, као и циљеви наставе физике буду у пуној мери реализовани.

Остали задаци наставе предмета Физика су да ученици:

- развијају функционалну писменост (природно-научна, математичка, техничка);
- систематски стичу знања о физичким појавама и процесима;
- разумеју појаве, процесе и односе у природи на основу физичких модела и теорија;
- развијају начин мишљења и расуђивања у физици;
- развијају свест о значају експеримента у сазнавању, разумевању и проверавању физичких закона;
- буду оспособљени за примену физичких метода мерења у свим областима физике;
- стекну способност за уочавање, формулисање, анализирање и решавање проблема;
- развијају компетенције за извођење једноставних истраживања;
- развијају логичко и апстрактно мишљење и критички став у мишљењу;
- схвате значај физике за технику и природне науке;
- развијају способности за примену знања из физике;

- стичу знања о природним ресурсима, њиховој ограничености и одрживом коришћењу;
- развијају правилан однос према заштити, обнови и унапређењу животне средине;
- развијају мотивисаност за учење и заинтересованост за садржаје физике;
- развијају радне навике, одговорност и способност за примену стечених знања.

IV разред

(4 часа недељно, 124 часова годишње)

САДРЖАЈИ ПРОГРАМА

I Релативистичка физика

1. Основни постулати специјалне теорије релативности. Лоренцове трансформације координата. Релативистички закон сабирања брзина. (II)
2. Релативистички карактер времена и дужине. Гранични карактер брзине светлости. (II)
3. Инваријантност интервала. (P)
4. Релативистички импулс и енергија. Везе између релативистичког импулса, кинетичке енергије, енергије мировања и укупне енергије. Унутрашња енергија. Закон одржања масе и енергије. (II)
5. Појам о општој теорији релативности. (O)

II Квантна природа електромагнетног зрачења

1. Топлотно зрачење. Закони зрачења апсолутно црног тела. Планкова хипотеза. (II)
2. Фотоелектрични ефекат. Ајнштајнова једначина фотоефекта. (II)
3. Квантна природа светлости. (P)
Маса и импулс фотона. Притисак светлости. Комптонов ефекат. (II)
Корпускуларно-таласни дуализам светлости. (P)
Демонстрациони оглед:
– Фотоефекат (помоћу фотоћелије).

III Таласна својства честица и појам о квантној механици

1. Честично-таласни дуализам. Де Бројева хипотеза. Дифракција електрона. (II)
Електронски микроскоп. (P)
2. Хајзенбергове релације неодређености. (P)
3. Појам о Шрединговој једначини. Таласне функције и сопствене енергије. (P)
4. Кретање слободне честице. Честица у потенцијалној јами. Квантни линеарни хармонијски осцилатор. Пролаз кроз потенцијалну баријеру. (P)

IV Квантна теорија атома

1. Радерфордов модел атома. (O)
Дискретни спектар атома водоника. Борови постулати и Боров модел атома водониковог топа. (II)
Франк-Херцов оглед. (P)
2. Квантно-механичка теорија атома: главни, споредни и магнетни квантни број. (II).
Физички смисао „боровских орбита”. Спин електрона. Штерн-Герлахов оглед. (P)
3. Вишелектронски атоми и Паулијев принцип. Структура периодног система елемената. (II)
4. Закочно и карактеристично рендгенско зрачење. (II)

Лабораторијска вежба

- Калибрација спектроскопа и идентификација водониковог спектра.
- Одређивање Ридбергове константе (помоћу водоникове лампе и дифракционе решетке)

V Молекулска структура и спектри

1. Основне карактеристике хемијских веза (јонске и ковалентне). Молекулски спектри. (P)

VI Физика чврстог стања

1. Зонска теорија кристала. Енергијске зоне у чврстом телу. Зонски модели метала и диелектрика. (P)

2. Расподела слободних електрона по енергијама у металу. Квантна теорија проводљивости метала. Суперпроводљивост. (P)

3. Полупроводници. Сопствена и примесна проводљивост. Полупро-водници р и n-типа и полупроводнички p-n спој. (P)

Полупроводничке диоде, транзистори и фотоотпорници. (П)

Демонстрациони огледи:

– Диоде. Фотопроводници.

Лабораторијске вежбе

- Струјно-напонска карактеристика диоде.
- Струјно-напонске карактеристике транзистора.
- Одређивање Планкове константе (помоћу LED диоде).

VII Индуковано зрачење и ласери

1. Луминисценција. Квантни прелаз: спонтана емисија, апсорпција и стимулирана емисија зрачења. (P)

2. Основни принцип рада ласера. Врсте ласера. Карактеристике ласерског зрачења. (P)

3. Примене ласера. Холографија. (P)

Лабораторијска вежба

– Одређивање угаоне дивергенције ласерског снопа.

VIII Физика атомског језгра

1. Структура језгра. Карактеристике језгра. (P)

Дефект масе и енергија везе. (П)

Нуклеарне силе. (P)

Модели језгра. (O)

2. Природна радиоактивност. Алфа-, бета- и гама распад. (П)

3. Закон радиоактивног распада. Активност радиоактивног извора. (П)

Радиоактивни нивои и радиоактивна равнотежа. (P)

4. Интеракција радиоактивног зрачења са супстанцијом. Детекција зрачења. (П)

Дозиметрија и заштита од зрачења. (P)

5. Вештачка радиоактивност. Општа својства нуклеарних реакција (П).

Примери реакција (откриће протона и неутрона, интеракције неутрона са језгром, трансурански елементи). (P)

6. Акцелератори честица. (P)

7. Нуклеарна енергетика. Фисија. Нуклеарни реактори. Реакције фузије на звездама. Конфинирање плазме. (P)

Нуклеарне и термонуклеарне бомбе. (O)

Демонстрациони оглед:

– Детекција радиоактивног зрачења.

Лабораторијске вежбе

- Мерење активности.
- Детекција радиоактивног зрачења.
- Опадање интензитета гама зрачења са повећавањем удаљености од извора.

IX Физика елементарних честица

1. Класификација елементарних честица. Основне интеракције између честица. Честице и античестице. Кваркови. (П)

2. Космичко зрачење. (P)

НАЧИН ОСТВАРИВАЊА ПРОГРАМА

Избор метода логичког закључивања

Програм предвиђа коришћење разних метода логичког закључивања који су иначе присутни у физици као научној дисциплини (индуктивни, дедуктивни, закључивање по аналогiji итд.). Наставник сам треба да одабере најпогоднији приступ у обради сваке конкретне теме у складу са потребама и могућностима ученика, као и наставним средствима којима располаже.

На садржајима програма може се у потпуности илустровати суштина методологије истраживачког приступа у физици и другим природним наукама: посматрање појаве, уочавање битних својстава система на којима се појава одвија, занемаривање мање значајних својстава и параметара система, мерење у циљу проналажења међузависности одабраних величина, планирање нових експеримената ради прецизнијег утврђивања тражених односа, формулисање физичких закона. У неким случајевима методички је целесходно увођење дедуктивне методе у наставу (нпр. показати како из закона одржања следе неки мање општи физички закони и сл.).

Демонстрациони огледи

Демонстрациони огледи чине саставни део редовне наставе физике, али су све мање заступљени. Присутна је недовољна опремљеност школа наставним средствима, у неким није заступљена ни кабинетска настава, али има и оних у којима се наставна средства не користе.

Увођење једноставних експеримената за демонстрирање физичких појава има за циљ „враћање” огледа у наставу физике, развијање радозналости и интереса за физику и истраживачки приступ природним наукама.

Једноставне експерименте могу да изводе и сами ученици (самостално или по групама) на часу или да их осмисле, ураде, анализирају и обраде код куће, користећи многе предмете и материјале из свакодневног живота.

Наравно, наставници који имају могућности треба да у настави користе и сложеније експерименте.

У настави свакако треба користити и рачунаре (симулације експеримената и појава, лабораторијске вежбе и обрада резултата мерења, моделирање, самостални пројекти ученика у облику семинарских радова и сл.).

Начин презентовања програма

Програмски садржаји доследно су приказани у форми која задовољава основне методске захтеве наставе физике:

– *Поступност* (од простијег ка сложенијем) при упознавању нових појмова и формулисању закона.

– *Оцигледност* при излагању наставних садржаја (уз сваку тематску целину побројано је више демонстрационих огледа а треба користити и симулације).

– *Повезаност наставних садржаја* (хоризонтална и вертикална).

Програм предвиђа да се унутар сваке веће тематске целине, после поступног и аналитичног излагања појединачних наставних садржаја, кроз систематизацију и обнављање изложеног градива, изврши синтеза битних чињеница и закључака и да се кроз њихово обнављање омогући да их ученици у потпуности разумеју и трајно усвоје. Поред тога, сваку тематску целину требало би започети *обнављањем одговарајућег дела градива из претходног разреда или из основне школе*. Тиме се постиже и вертикално повезивање наставних садржаја. Веома је важно да се кроз рад води рачуна о овом захтеву програма, јер се тиме наглашава чињеница да су у физици све области међусобно повезане и омогућује се да ученик сагледа физику као кохерентну научну дисциплину у којој се почетак проучавања нове појаве наслања на резултате проучавања неких претходних.

Редослед проучавања појединих тема није потпуно обавезујући. Наставник може прерасподелити садржаје према својој процени.

Нивои образовно-васпитног рада

Овога пута у садржајима програма није дат оријентациони број часова предвиђених за обраду наставних тема, обнављање градива и лабораторијске вежбе. То би могао бити „увод” у наставни процес у коме ће наставник, на основу дефинисаних циљева и задатака предмета, исхода и стандарда знања, самостално планирати број часова обраде, утврђивања... У недостатку стандарда знања корисни ће бити нивои образовно-васпитних захтева, који дефинишу обим и дубину проучавања појединих елемената садржаја програма, а постојали су и до сада.

Први ниво: обавештеност (О)

Обавештеност као ниво образовно-васпитних захтева изискује да ученик може да се сети – репродукује оно што је учио: термине, специфичне чињенице, методе и поступке, опште појмове, принципе (законе) или теорије. Значи, од ученика се очекује да градиво које је учио само познаје: да може да га искаже, исприча, опише, наведе и сл., тј. да може да га репродукује у битно неизмењеном облику.

Други ниво: разумевање (Р)

Разумевање као ниво образовно-васпитних захтева изискује да ученик буде оспособљен да градиво које је учио реорганизује: да одређене чињенице, појмове и принципе (законе) објасни, анализира, доведе у нове везе, које нису биле непосредно дате у градиву.

Разумевање као образовно-васпитни ниво укључује у себе и претходни ниво – обавештеност. Уколико се овде градиво интерпретира, онда се то чини не у форми у којој је било претходно dato, већ у реорганизованом, тј. у битно измењеном облику.

Трећи ниво: примена (П)

Примена као ниво образовно-васпитних захтева изискује да ученик буде оспособљен да одређене генерализације, принципе (законе), теорије или опште методе примењује у решавању проблема и задатака.

Овде је реч о примени оног што се зна и разуме у решавању нових проблема (задатака), а не о његовом јединственом, репродуктивном коришћењу у појединим ситуацијама. Примена као највиши образовно-васпитни ниво укључује у себе оба претходна нивоа – обавештеност и разумевање.

Пошто су програми друштвено-језичког и општег смера гимназије у другом и четвртном разреду исти, наставницима друштвено-језичког смера је остављена слобода да, према потреби, смање ниво образовно-васпитних захтева у тематским целинама за које сматрају да је то потребно (нпр. са нивоа (П) на ниво (Р)).

Основни облици наставе и методска упутства за њихово извођење

Методичко остваривање садржаја програма у настави физике захтева да целокупни наставни процес буде прожет трима основним физичким идејама: структуром супстанције (на молекулском, атомском и субатомском нивоу), законима одржања (пре свега енергије) и физичким пољима као носиоцима узајамног деловања физичких објеката. Даљи захтев је да се физичке појаве и процеси тумаче у настави паралелним спровођењем, где год је то могуће, макроприлаза и микроприлаза у обради садржаја.

Физику је нужно представити ученицима као живу, недовршену науку, која се непрекидно интензивно развија и мења, а не као скуп завршених података, непроменљивих закона, теорија и модела. Зато је нужно истаћи проблеме које физика решава у садашњем времену.

Данас је физика експликативна, теоријска и фундаментална наука и њеним изучавањем, заједно са осталим природним наукама, стичу се основе научног погледа на свет. Идеја фундаменталности физике у природним наукама мора да доминира у настави физике.

Ширењу видика ученика допринеће објашњење појмова и категорија, као што су физичке величине, физички закони, однос експеримента и теорије, веза физике с осталим наукама, с примењеним наукама и с техником. Значајно је указати на везу физике и

филозофије. Потребно је навести и етичке проблеме који се јављају као последица развијања науке и технике.

Овако формулисани концепт наставе физике захтева појачано експериментално заснивање наставног процеса (демонстрациони огледи и лабораторијске вежбе, односно практични рад ученика).

Усвојени концепт наставе физике захтева стварање разноврсних могућности да кроз различите садржаје и облике рада, применом савремених методичких и дидактичких поступака у наставном процесу (пројектна, проблемска, активна настава и кооперативно учење) циљеви и задаци образовања као и циљеви наставе физике буду у пуној мери реализовани.

Стицање техничке културе кроз наставу физике састоји се у развијању вештина техничких примена знања, у решавању техничких задатака и у приказивању одређених примена физике у свакодневном животу.

После изучавања одговарајућих тематских целина, нужно је указати на заштиту човекове средине, која је загађена и угрожена одређеним физичко-техничким процесима и променама.

При обради физичких основа енергетике потребно је усмерити ученике на штедњу свих врста енергије, а посебно електричне енергије.

Циљеви и задаци наставе физике остварују се кроз следеће основне облике рада са ученицима:

1. излагање садржаја теме уз одговарајуће демонстрационе огледе;
2. решавање квалитативних и квантитативних задатака;
3. лабораторијске вежбе;
4. коришћење и других начина рада који доприносе бољем разумевању садржаја теме (домаћи задаци, семинарски радови, пројекти...);
5. систематско праћење рада сваког појединачног ученика.

Веома је важно да наставник при извођењу прва три облика наставе наглашава њихову обједињеност. У противном, ученик ће стећи утисак да постоје три различите физике: једна се слуша на предавањима, друга се ради кроз рачунске задатке, а трећа се користи у лабораторији.

Да би се циљеви и задаци наставе физике остварили у целини, неопходно је да ученици активно учествују у свим облицима наставног процеса. Имајући у виду да сваки од наведених облика наставе има своје специфичности у процесу остваривања, то су и методска упутства прилагођена овим специфичностима.

Методска упутства за предавања

Како су уз сваку тематску целину планирани демонстрациони огледи, ученици ће спонтано пратити ток посматране појаве, или непосредно учествовати у реализацији огледа, а на наставнику је да наведе ученика да својим речима, на основу сопственог расуђивања, опише појаву коју посматра или демонстрира. После тога наставник, користећи прецизни језик физике, дефинише нове појмове (величине) и речима формулише закон појаве. Када се прође кроз све етапе у излагању садржаја теме (оглед, учеников опис појаве, дефинисање појмова и формулисање закона), прелази се на презентовање закона у математичкој форми. Оваквим начином излагања садржаја теме наставник помаже ученику да потпуније разуме физичке појаве, трајније запамти усвојено градиво и у други план потисне формализовање усвојеног знања.

Када је могуће, треба користити проблемску наставу. Наставник поставља проблем ученицима и препушта да они самостално, у паровима или у тиму дођу до решења, по потреби усмерава ученике, подсећајући их питањима на нешто што су научили и сада треба да примене, упућује их на извођење експеримента који може довести до решења проблема и слично.

Неке од тема могу се обратити самосталним радом ученика кроз радионице. Такав начин рада је ученицима најинтересантнији, више су мотивисани, па лакше усвајају знање. Уз то се развија и њихово интересовање и смисао за истраживачки рад, као и способност тимског рада и сарадње. Овакав приступ обради наставне теме захтева добру припрему наставника: одабрати тему, припремити одговарајућа наставна средства и опрему, поделити ученике

у групе тако да сваки појединац у групи може дати одговарајући допринос, дати неопходна минимална упутства...

Неке теме треба да припреме и презентују сами ученици, појединачно или у паровима.

Методска упутства за решавање задатака

Решавање задатака је важна метода за увежбавање примене знања. Њоме се постиже: конкретизација теоријских знања; обнављање, продубљивање и утврђивање знања; кориговање ученичких знања и умећа; развијање логичког мишљења; подстицање ученика на иницијативу; стицање самопоуздања и самосталности у раду...

Оптимални ефекти решавања задатака у процесу учења физике остварују се добро осмишљеним комбиновањем квалитативних (задаци – питања), квантитативних (рачунских), графичких и експерименталних задатака.

Вежбање решавања рачунских задатака је важна компонента учења физике. Како оно за ученике често представља вид учења са најсложенијим захтевима, наставник је обавезан да им да одговарајуће инструкције, напомене и савете у вези са решавањем задатака. Напомене треба да се односе на типове задатака у датом теми, најчешће грешке при решавању таквих задатака, алгоритм решавања датог типа задатака...

При решавању квантитативних задатака, у задатку прво треба на прави начин сагледати физичке садржаје, па тек после тога прећи на математичко формулисање и израчунавање. Наиме, решавање задатака одвија се кроз три етапе: физичка анализа задатка, математичко израчунавање и дискусија резултата. У првој етапи уочавају се физичке појаве на које се односи задатак, а затим се набрајају и речима исказују закони по којима се појаве одвијају. У другој етапи се, на основу математичке форме закона, израчунава вредност тражене величине. У трећој етапи тражи се физичко тумачење добијеног резултата.

Потребно је пажљиво одабрати задатке који, ако је могуће, имају непосредну везу са реалним ситуацијама. Такође је важно да ученици правилно вреднују добијени резултат, као и његов правилан запис. Посебно треба обратити пажњу на поступност при избору задатака, од најједноставнијих ка онима који захтевају анализу и синтезу стечених знања.

Методска упутства за извођење лабораторијских вежби

Лабораторијске вежбе чине саставни део редовне наставе и организују се тако што се при изради вежби одељење дели на два дела а ученици вежбе раде у групама, 2–3 ученика.

За сваку вежбу ученици унапред треба да добију одговарајућа упутства.

Час експерименталних вежби састоји се из уводног дела, мерења и записивања резултата мерења и обраде добијених података.

У уводном делу часа наставник проверава да ли су ученици спремни за вежбу, упознаје их са мерним инструментима и осталим деловима апаратуре за вежбу, указује на мере предострожности којих се морају придржавати ради сопствене сигурности, при руковању апаратима, електричним изворима, разним уређајима и сл.

Док ученици врше мерења, наставник активно прати њихов рад, дискретно их надгледа и, кад затреба, објашњава и помаже.

При обради резултата мерења ученици се придржавају правила за табеларни приказ података, цртање графика, израчунавање заокружених вредности и грешака мерења (са тим правилима наставник треба да их упозна унапред или да их да уз писана упутства за вежбе).

Методска упутства за друге облике рада

Један од облика рада са ученицима су домаћи задаци који садрже квалитативне и квантитативне задатке, понекад и експерименталне. Такви домаћи задаци односе се на градиво које је обрађено непосредно на часу и на повезивање овог градива са претходним.

За домаћи задатак могу се давати и семинарски радови и мањи пројекти, које би ученици радили индивидуално или у групама.

Наставник је обавезан да прегледа домаће задатке и саопшти ученицима евентуалне грешке како би они имали информацију о успешности свог рада.

Праћење рада ученика

Наставник је дужан да континуирано прати рад сваког ученика кроз непрекидну контролу његових усвојених знања, стечених на основу свих облика наставе: демонстрационих огледа, предавања, решавања квантитативних и квалитативних задатака, лабораторијских вежби, семинарских радова и пројеката...

Потребно је континуирано проверавати и оцењивати знање ученика помоћу усменог испитивања, кратких (15-минутних) писмених провера, тестова на крају већих целина (рецимо, по једном у сваком класификационом периоду), контролних рачунских вежби (по једном у полугодишту), провером експерименталних вештина.

На почетку рада са ученицима, посебно ако је одељење променило структуру или је наставник преузео ново одељење, било би пожељно спровести дијагностички тест. Такав тест говори наставнику са каквим предзнањем и којим потенцијалима ученици улазе у нову школску годину. Такође, предажемо тестове систематизације градива на крају сваког полугодишта или на крају школске године. Припрема за овај тест, као и сам тест, требало би да осигурају трајно усвајање најосновнијих и најважнијих знања из претходно обрађених области.

Додатна и допунска настава

Додатни рад намењен је даровитим ученицима и треба да задовољи њихова интересовања за физику. Организује се са једним часом недељно. У оквиру ове наставе могу се продубљивати и проширивати садржаји из редовне наставе, радити нови садржаји, тежи задаци, сложенији експерименти од оних у редовној настави... Ученици се слободно опредељују при избору садржаја програма. Зато је нужно сачинити индивидуалне програме рада са ученицима на основу њихових претходних знања, интересовања и способности. Корисно је да наставник позове истакнуте стручњаке да у оквиру додатне наставе одрже популарна предавања као и да омогући ученицима посете институтима.

Допунска настава се такође организује са по једним часом недељно. Њу похађају ученици који у редовној настави нису били успешни. Циљ допунске наставе је да ученик, уз додатну помоћ наставника, стекне минимум основних знања из садржаја које предвиђа програм физике.

Слободне активности ученика, који су посебно заинтересовани за физику, могу се организовати кроз разне секције младих физичара.

МАТЕМАТИКА

Циљ и задаци

Циљ наставе предмета Математика јесте: да ученици усвоје елементарне математичке компетенције (знања, вештине и вредносне ставове) које су потребне за схватање појава и законитости у природи и друштву и које ће да оспособе ученике за примену усвојених математичких знања (у решавању разноврсних задатака из животне праксе) и за успешно настављање математичког образовања и за самообразовање; као и да допринесе развијању менталних способности, формирању научног погледа на свет и свестраном развитку личности ученика.

Задаци наставе предмета Математика су да ученици:

- развијају логичко и апстрактно мишљење;
- развијају способности јасног и прецизног изражавања и коришћења основног математичко-логичког језика;
- развијају способности одређивања и процене квантитативних величина и њиховог односа;

– разликују геометријске објекте и њихове узајамне односе и трансформације;
– разумеју функционалне зависности, њихово представљање и примену;
– развијају систематичност, уредност, прецизност, темељност, истрајност, критичност у раду, креативност; развијају радне навике и способности за самостални и групни рад; формирају систем вредности;
– стичу знања и вештине корисне за трансфер у друге предмете и развијају способности за правилно коришћење стручне литературе;
– формирају свест о универзалности и примени математичког начина мишљења;
– буду подстакнути за стручни развој и усавршавање у складу са индивидуалним способностима и потребама друштва;
– развијају способности потребне за решавање проблема и нових ситуација у процесу рада и свакодневном животу.

I разред

(5 часова недељно, 180 часова годишње)

САДРЖАЈИ ПРОГРАМА

Логика и скупови

Основне логичке и скуповне операције. Важнији закони закључивања.
Основни математички појмови, дефиниција, аксиома, теорема, доказ. Декартов производ; релације, функције.
Елементи комбинаторике (пребројавање коначних скупова: правило збира и правило производа).

Реални бројеви

Преглед бројева; операције, поље реалних бројева.
Приближне вредности реалних бројева (грешке, граница грешке, заокруживање бројева; основне операције са приближним вредностима.
Пропорционалност.
Размера и пропорција, пропорционалност величина (директна, обрнута, уопштење); примене (сразмерни рачун, рачун поделе и мешања).
Процентни рачун, каматни рачун.
Таблично и графичко приказивање стања, појава и процеса.

Увод у геометрију

Основни и изведени појмови и ставови геометрије. Основни објекти геометрије: тачка, права, раван.
Основни ставови о релацијама припадања, распореда и паралелности. Међусобни положаји тачака, правих и равни.
Дуж, многоугаона линија. Полуправа, полураван, полупростор. Угао, диједар. Многоугао. Оријентација.

Рационални алгебарски изрази

Полиноми и операције са њима; дељивост полинома, Безуов став. Растављање полинома на чиниоце. Операције са рационалним алгебарским изразима (алгебарски разломци).

Подударност

Основни ставови о подударности. Изометрије, подударност геометријских објеката. Подударност дужи, углова, троуглова.
Прав угао. Нормалност правих и равни. Угао између праве и равни. Вектори и операције са њима.
Директне и индиректне изометрије. Симетрије, ротације и транслације равни и простора. Односи страница и углова троугла.
Кружница и круг.
Значајне тачке троугла. Четвороугао. Примене.
Конструктивни задаци (троугао, четвороугао, многоугао, кружница).

Линеарне једначине и неједначине. Линеарна функција

Линеарне једначине са једном и више непознатих.
Еквивалентност и решавање линеарних једначина са једном непознатом.
Линеарна функција и њен график.
Систем линеарних једначина са две и три непознате (разне методе решавања).
Примена линеарних једначина и система линеарних једначина на решавање различитих проблема.
Линеарне неједначине са једном непознатом и њихово решавање.

Сличност

Мерење дужи и углова.
Пропорционалност дужи: Талесова теорема. Хомотетија.
Сличност. Питагорина теорема.
Потенција тачке.
Примене.

Тригонометрија правоуглог троугла

Тригонометријске функције оштрог угла; основне тригонометријске идентичности. Решавање правоуглог троугла.

Степеновање и кореновање

Степен чији је изложилац цео број, операције; децимални запис броја у стандардном облику.
Функција $y=x^n (n \in \mathbb{N})$ и њен график
Корен; степен чији је изложилац рационалан број. Основне операције са коренима.

II разред

(5 часова недељно, 180 часова годишње)

САДРЖАЈИ ПРОГРАМА

Низови и матрице

Основни појмови о низовима (дефиниција, задавање, операције). Рекурентне формуле и низови. Аритметички низ, геометријски низ; примене.
Појам матрице. Сабирање матрица, множење матрице скаларом. Множење матрица. Транспонована матрица. Детерминанте. Инверзна матрица. Ранг матрице.

Комплексни бројеви

Појам комплексног броја и операције са њима. Коњуговано комплексни бројеви. Модуо комплексног броја.

Квадратна једначина и квадратна функција

Квадратна једначина са једном непознатом и њено решавање, дискриминанта и природа решења квадратне једначине.
Вијетове формуле. Растављање квадратног тринома на линеарне чиниоце; примене. Неке једначине које се свode на квадратне.
Квадратна функција (нуле, знак, рашћење и опадање, екстремна вредност, график). Квадратне неједначине.
Системи једначина са две непознате који садрже квадратну једначину (квадратна и линеарна, две чисто квадратне, хомогена квадратна и линеарна) – са графичком интерпретацијом.
Ирационалне једначине и неједначине.

Експоненцијална и логаритамска функција

Експоненцијална функција и њено испитивање (својства, график). Једноставније експоненцијалне једначине и неједначине.
Појам инверзне функције. Појам логаритма, основна својства. Логаритамска функција и њен график.
Основна правила логаритмовања, антилогаритмовање. Декадни логаритми. Примена логаритама у решавању разних задатака (уз употребу рачунара).
Једноставније логаритамске једначине и неједначине.

Тригонометријске функције

Уопштење појма угла; мерење угла, радијан.

Тригонометријске функције ма ког угла; вредности тригонометријских функција ма ког угла, свођење на први квадрант, периодичност.

Графици основних тригонометријских функција; графици функција облика: $y = A \sin(ax+b)$ и $y = A \cos(ax+b)$.

Адиционе теореме. Трансформације тригонометријских израза (тригонометријских функција двоструких углова и полууглова, трансформације збира и разлике тригонометријских функција у производ и обрнуто).

Тригонометријске једначине и једноставније неједначине. Синусна и косинусна теорема; решавање троугла.

Примене тригонометрије (у метричкој геометрији, физици, пракси).

III разред

(5 часова недељно, 175 часова годишње)

САДРЖАЈИ ПРОГРАМА

Полиедри

Рогољ, триедар. Полиедар, Ојлерова теорема; правилен полиедар. Призма и пирамида; равни пресеци призме и пирамиде.

Површина полиедра; површина призме, пирамиде и зарубљене пирамиде.

Запремина полиедра: запремина квадра, Кавалеријев принцип. Запремина призме, пирамиде и зарубљене пирамиде.

Обртна тела

Цилиндрична и конусна површ, обртна површ.

Прав ваљак, права купа и зарубљена права купа. Површина и запремина правог кружног ваљка, праве кружне купе и зарубљене кружне купе.

Сфера и лопта; равни пресеци сфере и лопте. Површина лопте, сферне калоте и појаса. Запремина лопте.

Уписана и описана сфера полиедра, правог ваљка и купе.

Системи линеарних једначина

Системи линеарних једначина, Гаусов поступак. Крамерова теорема. Решавање система линеарних једначина помоћу матрица. Систем линеарних неједначина са две непознате и његова графичка интерпретација; појам линеарног програмирања.

Вектори

Правоугли координатни систем у простору, пројекције вектора; координате вектора. Скаларни, векторски и мешовити производ вектора;

Неке примене вектора.

Аналитичка геометрија у равни

Растојање две тачке. Подела дужи у датој размери. Површина троугла.

Права, разни облици једначине праве; угао између две праве; растојање тачке од праве.

Криве линије другог реда: кружница, елипса, хипербола, парабола (једначине; међусобни односи праве и кривих другог реда, услов додира, тангента; заједничка својства).

Математичка индукција. Низови

Математичка индукција и њене примене.

Гранична вредност низа, својства. Број e . Сума бесконачног геометријског низа.

Комплексни бројеви и полиноми

Појам и примери алгебарских структура (група, прстен, поље).

Поље комплексних бројева. Тригонометријски облик комплексног броја. Моаврова формула. Неке примене комплексних бројева.

Полиноми над пољем комплексних бројева. Вијетове формуле. Системи алгебарских једначина вишег реда.

IV разред

(5 часова недељно, 155 часова годишње)

САДРЖАЈИ ПРОГРАМА

Функције

Важнији појмови и чињенице о функцијама једне променљиве (дефинисаност, нуле, парност, монотоност, периодичност). Сложена функција (појам и једноставнији примери).

Преглед елементарних функција.

Гранична вредност и непрекидност функције (геометријски смисао); асимптоте.

Извод функције

Прираштај функције. Извод функције (проблем тангенте и брзине). Основне теореме о изводу, изводи елементарних функција. Лангранжова теорема. Лопиталова теорема.

Диференцијал и његова примена код апроксимације функција. Испитивање функција (уз примену извода); график функције.

Интеграл

Неодређени интеграл. Основна правила о интегралу; таблица основних интеграла; интеграл неких елементарних функција.

Метод замене, метод парцијалне интеграције.

Одређени интеграл; Њутн-Лајбницева формула (без доказа). Примене одређеног интеграла (ректификација, квадратура, кубатура).

Вероватноћа и статистика

Случајни догађаји. Вероватноћа. Условна вероватноћа и независност. Случајне величине. Биномна, Пуасонова и нормална расподела. Средња вредност и дисперзија. Популација, обележје и узорак. Прикупљање, сређивање и приказивање података. Појам оцене параметара. Оцене вероватноће, средње вредности и дисперзије. Интервалне оцене за вероватноћу и средњу вредност.

Елементи нумеричке математике

Апсолутна, релативна и процентуална грешка. Декадни запис приближног броја. Значајне и сигурне цифре. Заокруживање бројева. Грешке аритметичких операција.

Општи задатак интерполације. Линеарна и квадратна интерполација. Лагранжеова интерполациона формула.

Локализација и изоловање решења. Појам приближног решења. Метода половљења сегмента. Метода сечице. Метода тангенте.

НАЧИН ОСТВАРИВАЊА ПРОГРАМА

Основне карактеристике програма Математике су: усклађеност са програмом математике за основну школу, логичка повезаност садржаја и настојање, где год је то могуће, да садржаји претходе садржајима других предмета у којима се примењују. При избору садржаја програма узета је у обзир значајна образовна функција математике, која се огледа у стицању нових математичких знања и подизања нивоа математичког образовања ученика и њен допринос даљем оспособљавању ученика да логички мисле и стваралачки приступају решавању различитих проблема, као и васпитна функција наставе Математике, јер се код ученика развија правилно мишљење и доприноси изграђивању низа позитивних особина личности.

За реализацију циља и задатака наставе Математике изабрани су садржаји програма који су довољно приступачни свим ученицима, а могу и стимулативно деловати на ученике јер имају могућност да их усвоје и на нешто вишем нивоу (већи степен апстракције и генерализације, синтезе и примене, стваралачко решавање проблема). Строгост у интерпретацији садржаја треба да буде присутна у прихватљивој мери, уз ослањање на математичку

интуицију и њено даље развијање, тј. мотивација и интуитивно схватање проблема треба да претходе строгости и критичности, а излагање градива мора бити праћено добро одабраним примерима и тек након довољног броја урађених таквих примера треба приступити генерализацији појма, чињенице и сл.

У делу Објашњења садржаја програма је годишњи фонд часова за сваки разред подељен по темама. Укупан број часова који је наведен за сваку тему представља оријентациони број часова у оквиру којег треба реализовати одговарајуће садржаје. Приликом израде оперативних планова наставник распоређује укупан број часова предвиђен за поједине теме по типовима часова (обрада новог градива, утврђивање и увежбавање, понављање, проверавање и систематизација знања), водећи рачуна о циљу и задацима предмета.

При обради нових садржаја треба се ослањати на постојеће искуство и знање ученика, и настојати, где год је то могуће, да ученици самостално изводе закључке. Основна улога наставника је да буде организатор наставног процеса, да подстиче, организује и усмерава активност ученика. Ученике треба упућивати да, осим уџбеника, користе и друге изворе знања, како би усвојена знања била трајнија и шира, а ученици оспособљени за примену у решавању разноврсних задатака.

Строгост у интерпретацији садржаја треба да буде присутна у прихватљивој мери, уз ослањање на математичку интуицију и њено даље развијање, а излагање градива мора бити праћено добро одабраним примерима, који пре свега доприносе развијању способности примене стечених знања.

На часовима треба комбиновати различите методе и облике рада, што доприноси већој рационализацији наставног процеса, подстиче интелектуалну активност ученика и наставу чини интересантнијом и ефикаснијом. Избор метода и облика рада зависи од од наставних садржаја и образовно-васпитних задатака које треба реализовати на часу, али и од специфичности одређеног одељења и индивидуалних карактеристика ученика.

Праћење и вредновање постигнућа ученика треба да буде континуирано кроз праћење активности ученика на часу, домаће задатке, усмене провере, контролне вежбе и писмене задатке.

ОБЈАШЊЕЊА САДРЖАЈА ПРОГРАМА (Посебне напомене о обради програмских тема)

I разред

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Логика и скупови (15)

Реални бројеви (14)

Пропорционалност (8)

Увод у геометрију (8)

Рационални алгебарски изрази (16)

Подударност (36)

Линеарне једначине и неједначине. Линеарна функција (18)

Сличност (14)

Тригонометрија правоуглог троугла (9)

Степеновање и кореновање (30)

Обавезна су четири двочасовна школска писмена задатка са једночасовним исправкама (12).

Логика и скупови – Ову тему треба реализовати кроз понављања, продубљивања и допуњавања оног што су ученици учили у основној школи. Ови логичко-скуповни садржаји (исказ, формула, логичке и скуповне операције, основни математички појмови, логичко закључивање и доказивање тврдњи, релације и функције) су извесна основа за виши ниво дедукције и строгости у реализацији осталих садржаја програма математике на овом ступњу образовања и васпитања ученика. При томе, нагласак треба да буде на овладавању математичко-логичким језиком и разјашњавању суштине значајних математичких појмова и чињеница, без превеликих формализација.

Важан моменат у спречавању формализма и усмеравању пажње у настави математике на суштинска питања јесте правилно

схватање улоге и места логичко-скуповне (па и геометријске) терминологије и симболике. Симболика треба да се користи у оној мери у којој олакшава изражавање и записе (а не да их компликује), штеди време (а не да захтева додатна објашњења), помаже да се градиво што боље разјасни (а не да отежава његово схватање).

Елементе комбинаторике дати на једноставнијим примерима и задацима, као примену основних принципа пребројавања коначних скупова. Треба имати у виду да обрадом ових садржаја није завршена и изградња појединих појмова, јер ће се они дограђивати у наредним разредима.

Реални бројеви – У краћем прегледу бројева од природних до реалних, треба извршити систематизацију знања о бројевима, стеченог у основној школи, посебно истичући принцип перманенције својстава рачунских операција. При томе посебну пажњу обратити на својства рачунских операција, као основу за рационализацију рачунања и трансформације израза у оквиру других тема. У зависности од конкретне ситуације, ово се може дати и на нешто вишем нивоу. Потребну пажњу треба посветити обради приближних вредности. При томе ученик треба да схвати да рачунање са реалним бројевима најчешће значи рачунање са приближним вредностима.

Пропорционалност величина – Карактеристика ове теме је што у њој долази до изражаја повезивање и примена разних математичких знања. На бази проширивања и продубљивања раније стечених знања, основну пажњу овде треба посветити примени функција директне и обрнуте пропорционалности и пропорција у решавању разних практичних задатака, повезујући то и са табличним и графичким приказивањем одређених стања, процеса и појава.

Увод у геометрију – Ово је уводна тема у геометрију нарочито у погледу упознавања ученика са аксиоматским приступом изучавању геометрије (основни и изведени појмови и ставови, дефиниције важнијих геометријских фигура). Полазећи од посебно изабраних аксиома припадања, распореда и паралелности треба на неколико једноставнијих примера упознати ученике са суштином и начином доказивања теорема.

Рационални алгебарски изрази – Циљ ове теме је да ученици, користећи упозната својства операција са реалним бројевима, овладају идејама и поступцима вршења идентичних трансформација полинома и алгебарских разликама. При томе тежиште треба да буде на разноврсности идеја, сврси и суштини тих трансформација, а не на раду са компликованим изразима. Одређену пажњу треба посветити важнијим неједнакостима (доказивање и примена: неједнакост између средина и др.).

Подударност – Обрада садржаја из ове теме (подударност) треба да буде наставак оног што се о томе учило у основној школи. Упознати ученике са дефиницијом, основним појмовима и операцијама са векторима. Операције са векторима прво урадити на примерима слободних вектора а онда са векторима у правоуглом Декартовом координатном систему.

Треба истаћи основна својства сваке од изучаваних изометрија и њихово вршење, а нешто продубљеније обратити изометријске трансформације као пресликавање равни у саму себе, њихову класификацију и нарочито њихове примене (као метода) у доказним и конструктивним задацима у вези са троуглом, четвороуглом и кружницом. Коришћење изометријских трансформација не искључује дедукцију као метод доказивања (у Еуклидовом смислу). Трансформације се користе у оној мери у којој олакшавају изучавање одређених садржаја геометрије.

Линеарне једначине и неједначине. Линеарна функција – У оквиру ове теме треба извршити продубљивање и извесно проширивање знања ученика о линеарним једначинама и неједначинама, која су стекли у основној школи, истичући појам еквивалентности једначина и неједначина и примену у њиховом решавању. Треба узимати и примере једначина у којима је непозната у имениоцу разломка, као и оне које садрже један или два параметра. У сваком случају, треба избегавати једначине са сувише сложеним изразима. На неколико једноставнијих примера може се показати и решавање система линеарних једначина са више од две непознате. У овој теми тежиште треба да буде у примени једначина на

решавање разних проблема. Приликом обраде неједначина и система неједначина са једном непознатом треба се ограничити само на оне које не садрже параметре. Решења неједначина записивати на више начина, користећи при томе првенствено унију и пресек скупова.

Сличност – У оквиру ове теме, поред заснивања мерења дужи и углова, (доведећи у везу самерљивост дужи с карактером размере њихових дужина) и продубљенијег усвајања Талесове теореме (са применама), ученици треба да упознају хомотетију као једну трансформацију равни која није изометријска, а сличност као композицију хомотетије и изометрије (односно, хомотетију као трансформацију сличности), као и да уоче практичне примене сличности. Посебно треба да схвате суштину метода сличности у решавању рачунских и конструктивних задатака. Такође је значајна примена сличности у доказивању појединих теорема (Питагорине и др.). Може се обрадити и однос површина сличних многоуглова (у виду задатка). Одговарајућу пажњу треба посветити примени Питагорине теореме у рачунским и конструктивним задацима.

Тригонометрија правоуглог троугла – Ученици треба добро да схвате везе између странице и углова правоуглог троугла (дефиниције тригонометријских функција оштрог угла), њихове последице и примене. При решавању правоуглог троугла треба се ограничити на једноставније и разноврсне задатке.

Степеновање и кореновање – Овде треба посветити пуну пажњу усвајању појма степена и корена и савлађивању операција са њима (на карактеристичним, али не много сложеним задацима). Од посебног је значаја релација

$$\sqrt{a^2} = |a|,$$

а такође и децимални запис броја у тзв. стандардном облику $a \cdot 10^n$, где је $1 \leq a < 10$ и $(n \in \mathbb{N})$.

Функцију испитивати само у неколико случајева (за $n \leq 4$), са закључком о облику графика када је изложила n паран и када је непаран број.

II разред

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Низови и матрице (28)

Комплексни бројеви (6)

Квадратна једначина и квадратна функција (48)

Експоненцијална и логаритамска функција (32)

Тригонометријске функције (54)

Обавезна су четири двочасовна школска писмена задатка са једночасовним исправкама (12).

Низови и матрице – На подесним и једноставним примерима објаснити појам низа као пресликавања скупа N у скуп R уз графичку интерпретацију. Обрадити различите начине задавања низова. Као значајне примере низова подробније обрадити аритметички низ и геометријски низ (дефиниција – основно својство, општи члан, збир првих n чланова).

Након дефинисања матрице увести и појмове квадратне, дијагоналне, троугаоне, нула и јединичне матрице. У делу који се односи на детерминанте обрадити начине израчунавања и особине детерминанте. Упознати ученике и са појмовима регуларна и сингуларна матрица, минор, кофактор, елементарне трансформације матрица и еквивалентне матрице. Указати на значај ове теме за предмете из области рачунарства и информатике.

Комплексни бројеви – У оквиру ове теме треба обрадити само основне појмове и чињенице које ће бити неопходне при изучавању садржаја о квадратној једначини.

Квадратна једначина и квадратна функција – Садржаји ове теме значајни су са становишта систематског изграђивања алгебре и практичних примена. Треба решавати и једначине са непознатом у именуиоцу разломка, које се свде на квадратне једначине,

као и једноставније једначине са параметрима. Потребну пажњу треба посветити примени квадратних једначина и неједначина у решавању разноврсних а једноставнијих проблема. Неопходно је да ученици добро науче да скицирају и „читају” график квадратне функције. При испитивању квадратне функције у већој мери треба користити управо њен график (његову скицу), не инсистирајући много на одређеној „шеми испитивања функције” у којој цртање графика долази тек на крају. Квадратне неједначине треба решавати користећи знања о знаку квадратног тринорма, као и знања о решавању линеарних неједначина. Решавати само једноставније ирационалне једначине.

Експоненцијална и логаритамска функција – Приликом обраде ових функција, за уочавање њихових својстава користити првенствено графичке интерпретације. На једноставним примерима упознати одређивање логаритама (у циљу продубљивања појма логаритма). Логаритмовање обрадити у мери неопходној за практичне примене.

Тригонометријске функције – При дефинисању и уочавању својстава тригонометријских функција ма ког угла у тзв. свођењу на први квадрант треба користити тригонометријску кружницу, као и симетрију (осну и централну). Упоредо са одређивањем вредности тригонометријских функција, треба решавати и тригонометријске једначине облика: $\sin ax=b$, $\cos ax=b$, $\operatorname{tg} ax=b$. Ученици треба да схвате да се многи научни и технички проблеми моделују тригонометријским функцијама, па је зато неопходно настојати да упознају основна својства ових функција, а првенствено да умеју да скицирају и „читају” њихове графике. Посебну целину у тригонометријским садржајима представљају адиционе теореме и њихове последице. Оне су значајне не само за одређене идентичне трансформације у самој тригонометрији, већ и за примене у неким другим предметима. Упознавањем синусне и косинусне теореме ученици треба да схвате да се проширују могућности примене тригонометрије на решавање ма којег троугла, као и на решавање разних проблема из метричке геометрије, физике и посебно техничке праксе.

III разред

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Полиедри (25)

Обртна тела (20)

Системи линеарних једначина (20)

Вектори (18)

Аналитичка геометрија у равни (40)

Математичка индукција. Низови (20)

Комплексни бројеви и полиноми (20)

Обавезна су четири двочасовна школска писмена задатка са једночасовним исправкама (12).

Полиедри и обртна тела – При обради ових садржаја, ученици треба да примењују већ усвојене појмове и чињенице просторне геометрије у решавању задатака, а акценат треба да буде на задацима практичне природе (одређивање запремине модела неког геометријског тела, конкретне грађевине или предмета, ако нису непосредно дати неопходни подаци и сл.), односно на задацима у којима се види да се изучавања својства просторних фигура широко користе у пракси, астрономији, физици, хемији и др. Посебну пажњу треба посветити даљем развијању логичког мишљења и просторних представа ученика, чему у извесној мери може допринети позивање на очигледност, коришћење модела (па и приручних средстава) и правилно скицирање просторних фигура. Поред тога, треба повремено од ученика тражити да дају процену резултата рачунског задатка. Низом задатака може се илустровати и чињеница да је често рационалније и боље прво наћи решење задатка у „општем облику”, па онда замењивати дате податке. Мада у програму није експлицитно наведено, може се као задатак дати одређивање односа површина и односа запремина сличних полиедара и сличних обртних тела, а такође и одређивање полупречника уписане или описане сфере.

Системи линеарних једначина – У оквиру ове теме ученици треба да продубе и прошире знање о системима линеарних једначина, упознају системе линеарних неједначина са две непознате и упознају суштину проблема линеарног програмирања. Ови садржаји пружају могућност за повезивање раније стечених знања о једначинама, неједначинама, матрицама и неким геометријским појмовима.

Вектори – Основно у овој теми је да ученици упознају дефиницију и смисао скаларног, векторског и мешовитог производа вектора, као и координате вектора. Од посебног је значаја координатна интерпретација скаларног, векторског и мешовитог производа и њихова примена (одређивање угла између два вектора, израчунавање површине и запремине фигура, неке примене у физици и др.).

Аналитичка геометрија у равни – Основни циљ у реализацији ове теме јесте да ученици схвате суштину координатног метода и његову ефикасну примену. Ученике треба оспособити да, на основу својстава праве и кривих линија другог реда, формирају њихове једначине и испитују међусобне односе тих линија. Потребно је указати и на примену аналитичког апарата при решавању одређених задатака из геометрије.

Математичка индукција. Низови – Ученици треба да схвате значај и суштину метода математичке индукције као посебног и ефикасног метода у математици за доказивање појединих тврђења. Овај метод треба увести и увежбати помоћу што једноставнијих примера. Повезивање ових садржаја са трансформацијама полинома омогућава ефикасно решавање низа разноврсних задатака из елементарне теорије бројева – области која је врло прикладна за развијање математичког језика и расуђивања (докази разних теорема у вези са дељивошћу и својствима целих бројева, њихових квадрата, простим бројевима, конгруенцијама и сл.).

Појам граничне вредности бесконачног низа дати на што једноставнијим примерима и извести образац за збир чланова бесконачног геометријског низа, уз илустровање и неким примерима примене (периодични децимални разломци, једноставнији примери из геометрије).

Комплексни бројеви и полиноми – Остварити даље проширивање и продубљивање знања ученика о бројевима, посебно комплексним, полиномима и једначинама – на одговарајућим задацима и примерима примене.

IV разред

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Функције (29)

Извод функције (32)

Интеграл (27)

Вероватноћа и статистика(31)

Елементи нумеричке математике (24)

Обавезна су четири двочасовна школска писмена задатка са једночасовним исправкама (12).

Функције – Овде треба допунити и систематизовати ученичка знања о функцији и њеним основним својствима, а затим направити преглед елементарних функција. Упознавање граничне вредности и непрекидности функције треба да буде на основу интуитивног приступа тим појмовима. Није потребно дуже задржавање на техници одређивања граничне вредности разних функција, већ акценат треба да буде на неколико карактеристичних лимеса.

Извод функције – Ученике треба упознати са појмовима прираштаја независно променљиве и прираштаја функције и, полазећи од појма средње брзине и проблема тангенте на криву, формирати појам количника прираштаја функције и прираштаја независно променљиве, а затим дефинисати извод функције као граничну вредност тог количника када прираштај независно променљиве тежи нули. Указати на основне теореме о изводу и изводе неких елементарних функција. Уз појам диференцијала и његово геометријско значење треба указати и на његову примену код апроксимације функција. Потребну пажњу треба посветити испитивању функција и цртању њихових графика, користећи извод функције (не узимајући сувише компликоване примере).

Интеграл – Програм предвиђа да се прво обради неодређени интеграл, па је потребно указати на везу између извода и интеграла и дати појам примитивне функције. Интеграљење протумачити као операцију која је инверзна диференцирању. Поред таблице основних интеграла треба показати и неке методе интеграљења (метода замене и метода парцијалне интеграције). Полазећи од проблема површине и рада, доћи до појма одређеног интеграла као граничне вредности збира бесконачно много бесконачно малих величина. Указати на основна својства одређеног интеграла, а акценат треба да буде на применама одређеног интеграла.

Вероватноћа и статистика – На почетку ове теме потребно је обновити комбинаторику. После увођења појма случајног догађаја дати појам вероватноће (преко појма релативне фреквенције и класичном дефиницијом), као и основне теореме о вероватноћи. На подесним примерима треба увести појам случајне променљиве и указати на неке њене нумеричке карактеристике и расподеле. Ваља истаћи улогу случајног узорка и статистичког експеримента, а затим објаснити начин прикупљања података, њиховог приказивања и одређивања важнијих статистичких карактеристика.

Елементи нумеричке математике – Увести појам приближног броја и рачунања са приближним бројевима. Објаснити поделу грешке према пореклу и класичну поделу на апсолутну и релативну грешку. Указати посебно на везу релативне грешке и броја сигурних цифара у приближном броју.

Увести појам интерполације као посебног случаја апроксимација функција указујући на значај њене примене у пракси. Навести посебно да је интерполација од користи када се врше експерименти или нека мерења, а функција коју интерполирамо је тако, уместо аналитичким изразом, дата скупом података, што је веома честа појава у пракси. Извести формулу за Лагранжов интерполациони полином и доказати јединственост таквог полинома. Одредити грешку у интерполацији полиномом.

Делу који се односи на приближно решавање једначина треба посветити посебну пажњу јер на директан начин указује на значај примене нумеричке математике и њених принципа у решавању математичких проблема, у овом случају приближном налажењу корена једначине. Обработити локализацију и изоловање решења једначина, а затим нумеричке алгоритме за ефективно налажење корена једначине, тј. методе половљења сегмента, методе сечице и методе тангенте.

ДИСКРЕТНА МАТЕМАТИКА

Циљ наставе предмета Дискретна математика је овладавање концептима, знањима и техникама које припадају дискретној математици, оспособљавање да се научено примени у рачунарству и у решавању практичних проблема, развијање апстрактног и критичког мишљења, способности комуникације математичким писмом и формалним математичким језиком, повезивање стечених знања и вештина са новим сазнањима и применама у даљем школовању.

Задаци наставе предмета Дискретна математика су да ученици:

- развијају логичко и комбинаторно мишљење;
- развијају способности јасног и прецизног изражавања и коришћења основног математичко-логичког језика;
- повезују стечена знања и вештине са садржајима сродних наставних предмета;
- стекну знања и вештине корисне за трансфер у друге, посебно информатичке, предмете и развијају способности за правилно коришћење стручне литературе;
- развијају способности потребне за решавање проблема и нових ситуација у свакодневном животу.

II разред

(2 часа недељно, 72 часа годишње)

САДРЖАЈИ ПРОГРАМА

Логика и скупови

Искази, основне логичке операције. Исказне формуле, исказна валуација, таутологије (важни примери и методе доказивања).

ОБЈАШЊЕЊА САДРЖАЈА ПРОГРАМА (Посебне напомене о обради програмских тема)

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Исказна логика и скупови (20)

Елементарна теорија бројева (16)

Комбинаторика (14)

Графови (16)

Обавезна су два двочасовна школска писмена задатка са једночасовним исправкама.

Исказна логика и скупови – Ову тему треба реализовати кроз понављање, допуњавање и продубљивање онога што су ученици учили у I разреду (исказ, логичке операције, исказне формуле, скуповне операције, релације и функције). Као примере потпуних система везника довољно је навести \neg , \wedge , као и \vee . Нови логичко-скуповни садржаји (партиције скупа и релације еквиваленције, важне таутологије, методе доказивања, ДНФ и КНФ, Булове алгебре) су извесна основа за реализацију каснијих програмских садржаја, као и за примене у рачунарству.

Елементарна теорија бројева – Нагласити својства релације деливости природних бројева продубљујући знања о релацији поретка. Објаснити решавање линеарних диофантских једначина (са или без Еуклидовога алгоритма) по две непознате, са коефицијентима најпре <10 , а затим <100 . Нагласити главна својства конгруенција целих бројева продубљујући знања о релацији еквиваленције. На примерима показати како добијамо критеријуме деливости неким бројем, увести рачун остатака (истаћи да се својства операција у скупу целих бројева преносе и на модуларну аритметику), написати таблице операција, извести периодичност степена сваког остатка и применити у задацима. Посматрањем система две линеарне конгруенције по једној непознатој показати даље примене диофантских једначина и модуларне аритметике.

Комбинаторика – Треба истаћи да је циљ комбинаторних садржаја пребројавање коначних скупова који су настали, или су издвојени, од скупова са задатим бројем елемената. Као садржај из I разреда, треба поновити правило производа (број елемената Декартовог производа скупова) и правило збира (број елемената дисјунктне уније). На добро изабраним примерима и задацима, кроз допуњавање и продубљивање знања, реализовати формулу укључења и искључења (број елемента уније). Варијације са и без понављања (све функције и све инјекције из A у B), као и пермутације скупа (све бијекције) обрадити као сродне теме и увежбати на довољном броју задатака; увести факторијел. Комбинације (без понављања) увежбати на разноврсним задацима и повезати са биномном формулом.

Показати како се решавају једноставније диферендне једначине (линеарне, са константним коефицијентима, првог реда или хомогене другог реда) и увежбати њихову примену у комбинаторици на довољном броју разноврсних задатака.

Графови – Дефинисати граф, његове одреднице (чворове, гране; петље), описати представљање цртежом, увести типове графова: ако је граф задат конкретно, онда кажемо само граф, јер је потпуно одређен и све особине се виде; ако је граф произвољан, дат само као појам, онда је пожељно да се нагласи да ли је (прост) граф, или је (граф) мултиграф са петљама; слично се односи на оријентацију, или усмереност. Дефинисати степен чвора, регуларан граф степена r , комплетан граф (K_n), циклe (C_n), нацртати неке од ових примера, увежбати одговарајуће задатке. Дефинисати бипартитан граф, комплетан бипартитан граф ($K_{r,s}$), звезду ($K_{1,s}$). Дефинисати подграф, изоморфизам графова, комплемент графа и самокомплементарне графове. У делу теме који се односи на путеве, повезаност; шуме, стабла, шетње, Ојлерове и пленарне графове, проблем најкраћег пута, Дијкстра алгоритам, Минимални скелет и Краскалов алгоритам, акценат треба да буде на оспособљавању ученика да научено примене у задацима. О бојењу чворова и ивица довољно је дати само основне информације.

Потпун систем везника, примери. Дисјунктивна нормална форма (ДНФ) и конјунктивна нормална форма (КНФ). Скупови, једнакост и инклузија, скуповне операције. Партитивни скуп. Декартов производ. Доказивање скуповних релација помоћу таутологија. Релације: граф бинарне релације; релација поретка (инклузија); релација еквиваленције и класе еквиваленције (једнакост, логичка еквивалентност). Булове алгебре, аксиоме и примери (двочлана, Булова алгебра партитивног скупа, класе еквиваленције исказних формула са коначно много исказних слова).

Елементарна теорија бројева

Релација деливости, прости бројеви (основна теорема аритметике). Еуклидов алгоритам. Диофантске једначине. Конгруенције (једнакост остатака), својства, рачун остатака и примене. Систем линеарних конгруенција.

Комбинаторика

Предмет комбинаторике, правило збира и правило производа, формула укључења и искључења. Варијације и пермутације. Комбинације, биномна формула и биномни коефицијенти. Пермутације са понављањем. Рекурентне формуле и низови, једноставније диферендне једначине и њихова примена у комбинаторици.

Графови

Појам, одреднице (чворови, гране; петље) и типови графова (прост, мултиграф са петљама, оријентисан); подграф, изоморфизам. Степен чвора, регуларани и комплетни графови, цикли. Комплетни бипартитни графови. Путеве, повезаност; шуме, стабла. Шетње, Ојлерови графови. Планарни графови. Проблем најкраћег пута и Дијкстра алгоритам. Минимални скелет и Краскалов алгоритам. Бојење графова. Представљање графа матрицама: матрица суседства и матрица инциденције.

НАЧИН ОСТВАРИВАЊА ПРОГРАМА

Основне карактеристике програма Дискретне математике су: усклађеност са програмом математике за средњу школу, логичка повезаност садржаја и настојање, где год је то могуће, да садржаји претходе садржајима других предмета у којима се примењују. При избору садржаја програма узето је у обзир да Дискретна математика треба да представља мост (или везу) између математике и рачунарства, као и образовна функција предмета која се огледа у стицању нових математичких знања и подизања нивоа математичког образовања ученика и њен допринос даљем оспособљавању ученика да логички мисле и стваралачки приступају решавању различитих проблема.

При обради нових садржаја треба се ослањати на постојеће искуство и знање ученика, и настојати, где год је то могуће, да ученици самостално изводе закључке. Ученике треба упућивати да, осим уџбеника, користе и друге изворе знања, како би усвојена знања била трајнија и шира, а ученици оспособљени за примену у решавању разноврсних задатака.

Строгост у интерпретацији садржаја треба да буде присутна у прихватљивој мери, уз ослањање на математичку интуицију и њено даље развијање, а излагање градива мора бити праћено добро одабраним примерима, који пре свега доприносе развијању способности примене стечених знања у рачунарским предметима.

На часовима треба комбиновати различите методе и облике рада, што доприноси већој рационализацији наставног процеса, подстиче интелектуалну активност ученика и наставу чини интересантнијом и ефикаснијом. Избор метода и облика рада зависи од од наставних садржаја и образовно-васпитних задатака које треба реализовати на часу, али и од специфичности одређеног одељења и индивидуалних карактеристика ученика.

Праћење и вредновање постигнућа ученика треба да буде континуирано кроз праћење активности ученика на часу, домаће задатке, усмене провере, контролне вежбе и писмене задатке.

ЛИКОВНА КУЛТУРА

Циљ и задаци

Циљ наставе предмета Ликовна култура је унапређивање ликовне писмености и естетских критеријума, оспособљавање за успешну визуелну и вербалну комуникацију.

Задаци наставе предмета Ликовна култура су да ученици:

– унапреде знања о ликовним законитостима, техникама, материјалима и прибору и буду оспособљени за примену стеченог знања у настави и свакодневном животу;

– упознају национално и светско културно и уметничко наслеђе и развију одговоран однос према очувању културне и уметничке баштине;

– унапреде естетске критеријуме;

– развијају визуелно опажање, моторику, стваралачко и критичко мишљење, радозналост, имагинацију и иновативност кроз разноврсне ликовне и проблемске задатке и буду оспособљени да стечена искуства примењују у настави других предмета, свакодневном животу, даљем школовању и будућем занимању;

– стекну увид у значај повезаности ликовног стваралаштва са осталим уметностима, наукама и процесима рада;

– буду оспособљени за индивидуални и тимски рад;

– буду оспособљени да примене стечена знања и умења у уређењу, унапређењу и очувању животне и радне средине.

III разред

(1 час недељно, 35 часова годишње)

САДРЖАЈИ ПРОГРАМА

I Целина

а) Општи појмови

Шта је уметност. Уметност у простору, уметност у времену. Уметност и природа. Уметност и друштво.

б) Самостално ликовно изражавање

Линија као елемент уметничког израза. Боја као средство сликарског обликовања. Тродимензионално обликовање.

в) Уметничко наслеђе

Појава уметничког стварања. Праисторијски споменици на територији Србије.

Уметност старог Истока.

Развој уметности у Грчкој. Најзначајнија остварења и водећи уметници.

Уметност старог Рима. Антички споменици у Србији. Појава хришћанске уметности. Византијска уметност. Исламска уметност.

II Целина

а) Општи појмови

Перспектива: инверзна, линеарна, ваздушна.

б. Уметничко наслеђе

Средњовековна уметност западне Европе.

Српска средњовековна уметност: Рашка школа, Српска средњовековна уметност од 1300–1371. год. и Моравска школа (избор споменика по архитектури и живопису).

Рана и висока ренесанса у Италији. Водећи представници и њихова дела.

в) Самостално ликовно изражавање. Простор. Перспектива.

III Целина

а) Општи појмови

Светло и сенка као елементи сликарског и скулпторског обликовања.

б. Самостално ликовно изражавање. Светлински односи. Боја.

в) Уметничко наслеђе

Барокна уметност – опште одлике и водећи уметници барока (Бернини, Каравађо, Веласкез, Рубенс, Рембрант). Уметност 19. века (неокласицизам, романтизам и реализам) у Европи и Србији и водећи представници (Давид, Енгр, Жерико, Делакроа, Рид,

Тарнер, Курбе; Катарина Ивановић, Ђура Јакшић, Новак Радонић, Паја Јовановић, Урош Предић и Ђорђе Крстић).

IV Целина

а) Општи појмови

Нови облици изражавања у уметности.

б) Уметничко наслеђе

Правци у уметности крајем 19. века: импресионизам и постимпресионизам. Уметност прве половине 20. века: опште одлике, најпознатији правци и појаве. Тенденције у уметности друге половине 20. века.

в) Самостално ликовно изражавање.

Композиција (Компоновање објеката у одређеном простору с обзиром на намену). Фотографија. Филм – покретна слика.

НАЧИН ОСТВАРИВАЊА ПРОГРАМА

Садржаји програма у оквиру једне целине равноправно чине: теорија обликовања, самостално ликовно изражавање, уметничко наслеђе.

Теорија обликовања се реализује кроз теоретски и практичан рад и у функцији је унапређивања ликовне писмености (ликовни елементи и принципи компоновања), овладавања традиционалним и савременим техникама и материјалима и оспособљавања ученика да стечена знања и умења примењују у настави, свакодневном животу, даљем школовању и будућем занимању. Обрада теоретских садржаја укључује и рад на ликовним задацима, посредно и непосредно проучавање репрезентативних уметничких дела и перцепцију природе и окружења, а увежбавање се реализује кроз проблемске задатке.

Самостално ликовно изражавање подразумева стваралачко изражавање традиционалним и савременим медијима у складу са индивидуалним способностима и интересовањима ученика. Ученицима је потребно омогућити самостални избор медија, мотива, техника, материјала и формата.

Ученике треба упознати са најзначајнијим уметничким делима светске уметности, а већу пажњу посветити националном културном и уметничком наслеђу. Садржаје програма је неопходно реализовати савременим наставним методама и средствима. У оквиру сваке програмске целине, ученике треба оспособљавати за поређење и успостављање веза између различитих садржаја (нпр. повезивање садржаја предмета са свакодневним искуством, садржајима других предмета и др.); самопроцену и процену ликовних и уметничких дела и ефикасну визуелну и вербалну комуникацију.

Креативност подразумева подршку ученицима да откривају, експериментишу, успостављају нове односе и долазе до нових решења у различитим активностима и оспособљавање ученика да стечено искуство примене у другим наставним предметима, свакодневном животу и будућим занимањима.

За подстицање креативности неопходно је имати у виду значај мотивационих садржаја, домен ученичких доживљаја и корелацију.

Развој стваралачког мишљења треба подстицати у свим наставним активностима и садржајима у којима је то могуће.

Садржаји се реализују кроз следеће облике наставе:

– индивидуални;

– фронтални;

– рад у паровима;

– групни или тимски;

– домаћи рад;

– екскурзије, посете локалитетима, музејима, галеријама, установама за образовање уметника, уметничким атељеима; сусрете са уметницима и активно учешће у културним и уметничким манифестацијама и активностима.

Корелација са другим предметима:

– Музичка култура;

– Српски језик и књижевност;

– Историја;

– Географија;

- Биологија;
- Физика;
- Хемија;
- Математика;

Вредновање и оцењивање:

У настави ликовне културе оцењује се активност ученика, процес рада и постигнути напредак, а вреднује и самовреднује резултат рада.

ПРИМЕНА РАЧУНАРА

Циљ и задаци

Циљ наставног предмета Примена рачунара је стицање знања, овладавање вештинама и формирање вредносних ставова који доприносе развоју информатичке писмености неопходне за даље школовање, живот и рад у савременом друштву, као и оспособљавање ученика да ефикасно и рационално користе рачунаре на начин који не угрожава њихово физичко и ментално здравље.

Задаци наставе Примена рачунарасу да ученици:

- развију свест о неопходности коришћења рачунара у свакодневном животу и раду и значају информатике за функционисање и развој друштва;
- стекну знања потребна за подешавање параметара оперативног система на нивоу корисничког интерфејса, коришћење могућности оперативних система и система датотека конкретних оперативних система;
- овладају коришћењем програма за обраду текста и табеларних података и креирање докумената у коме су интегрисани текст, слика и табела;
- упознају начине израде презентација и оспособе се за израду једноставнијих презентација;
- разумеју принципе функционисања интернета, локалних мрежа и оспособе се за коришћење мрежних ресурса, интернет сервиса и система за електронско учење;
- упознају принципе представљања и обраде цртежа и слика на рачунару и овладају техникама коришћења једног од графичких програма за обраду цртежа и слика;
- унапреде способности за брзо, ефикасно и рационално проналажење информација коришћењем рачунара, као и њихово критичко анализирање, складиштење и преношење;
- унапреде стратегије и технике самосталног учења користећи могућности рачунара и развију спремност за учење током целог живота;
- примене стечена знања и вештине у савладавању програма других наставних предмета;
- израде правилне ставове према коришћењу рачунара, без злоупотребе и претеривања које угрожава њихов физичко и ментално здравље;
- упознају савремена ергономска решења која олакшавају употребу рачунара и израде спремност за праћење нових решења у области информатичке технологије.

САДРЖАЈИ ПРОГРАМА

I разред

(1+2 часа недељно, 108 часова годишње)

1. Основи информатике

- Основни појмови информатике и рачунарства (информације, подаци, информационо-комуникационе технологије, предмет и области информатике и рачунарства).
- Представљање информација у дигиталним уређајима (дигитализација, представљање слика, звука, текста, бројевни системи, запис бројева, јединице за мерење количине информација).
- Развој информатичких технологија (прикупљања, складиштења, обраде, приказивања и преноса података).
- Утицај савремене рачунарске технологије на живот људи (значај и примена рачунара, карактеристике информационог друштва, утицај коришћења рачунара на здравље, безбедност и приватност).

2. Архитектура рачунарског система

- Структура и принцип рада рачунара.
- Процесор.
- Меморијска хијерархија.
- Улазно-излазни уређаји.
- Магистрале.
- Савремена хардверска технологија.
- Утицај компоненти на перформансе рачунара.

3. Програмска подршка рачунара

- Апликативни софтвер.
- Системски софтвер.
- Оперативни систем.
- Дистрибуција софтвера (верзије и модификације програма, влаштителство над софтвером, заштита права на интелектуалну својину).

4. Основе рада у оперативном систему са графичким интерфејсом

- Основни елементи ГКИ и интеракција са њима (радна површина, прозори, менији, дугмад, акције мишем, пречице на тастатури, ...).
- Основна подешавања оперативног система (подешавање датума и времена, радне површине, регионална подешавања, подешавања језика и тастатуре, коришћење и подешавање корисничких налога).
- Инсталирање и уклањање програма (апликативних програма, драјвера).
- Рад са документима и системом датотека.
- Средства и методе заштите рачунара и информација.

5. Обрада текста уз помоћ рачунара

- Приступити уносу и обради текста (текст-едитори, језици за обележавање, текст-процесори).
- Радно окружење текст-процесора и његово подешавање.
- Унос текста и његово једноставно уређивање (ефикасно кретање кроз текст, копирање, премештање, претрага, замена текста).
- Форматирање и обликовање текста (странице, пасуса, карактера).
- Посебни елементи у тексту (листе, табеле, слике, математичке формуле, ...).
- Коришћење и израда стилова, генерисање садржаја.
- Алатке интегрисане у текст-процесоре (провера граматике и правописа, редиговање текста, библиографске референце, индекс појмова, циркуларна писама, ...).
- Коришћење готових шаблона и израда сопствених шаблона.
- Штампанаје докумената.

6. Слајд-презентације

- Презентације и њихова примена (правила добре презентације, етапе у изради презентација).
- Радно окружење програма за израду слајд-презентација и његово подешавање (погледи на презентацију).
- Креирање слајдова (уметање и форматирање текста, графика, слика, звучних и видео-записа, ...).
- Складно форматирање слајдова (мастер слајд).
- Анимације (анимације објеката на слајдовима, анимације прелаза између слајдова, аутоматски прелазак између слајдова и снимање нарације).
- Интерактивне презентације (хипервезе, акциона дугмад).
- Штампанаје презентације.

7. Рад са табелама

- Основни појмови (прикупљање података, њихово табеларно и графичко приказивање на разне начине, као и читање и тумачење таквих приказа).
- Основни појмови о програмима за рад са табелама (структура документа, формати датотека).

- Подешавање радног окружења (палете алатки, пречице, лењир, поглед, зум) додавање, брисање, премештање и преименовање радних листова.

- Типови података.

- Уношење података у табелу (појединачни садржаји ћелија и аутоматске попуне).

- Подешавање димензија, премештање, фиксирање и сакривање редова и колона.

- Уношење формула са основним аритметичким операцијама, уз коришћење референце на ћелије.

- Копирање формула, релативно и апсолутно референцирање ћелија.

- Функције за: сумирање, средњу вредност, минимум, максимум, пребројавање, заокруживање.

- Логичке функције.

- Форматирање ћелија (број децималних места, датум, валута, проценат, поравнање, прелом, оријентација, спајање ћелија, фронт, боја садржаја и позадине, стил и боја рама ћелије).

- Сортирање и филтрирање.

- Намена различитих типова графикона, приказивање података из табеле помоћу графикона.

- Подешавање изгледа странице документа за штампање (оријентација папира, величина, маргине, прелом, уређивање заглавља и подножја, аутоматско нумерисање страна).

- Прегледање документа пре штампања, аутоматско штампање насловног реда, штампање опсега ћелија, целог радног листа, целог документа, графикона и одређивање броја копија, штампање документа.

8. Интернет и електронска комуникација

- Појам рачунарске мреже.

- Рачунари-сервери и рачунари-клијенти.

- Глобална мрежа (Интернет).

- Интернет-провајдери и њихове мреже.

- Технологије приступа Интернету.

- Сервиси Интернета: World Wide Web, FTP, електронска пошта, веб-форуми. Веб-читачи.

- Претраживачи, претраживање и коришћење информација са Интернета.

- Друштвене мреже и њихово коришћење.

- Електронска трговина, електронско пословање и банкарство.

- Електронски подржано учење.

II разред

(2 часа недељно, 72 часова годишње)

1. Рачунарска графика

Увод у рачунарску графику

- Начини представљања слика у рачунару – векторска и растерска графика.

- Карактеристике рачунарске графике – резолуција и број боја.

- Класификација програма за рад са рачунарском графиком.

- Формати датотека.

- Улазне и излазне графичке јединице.

Пример програма за креирање и обраду растерске графике

- Увоз слике са камере и скенера.

- Основне корекције слике.

- Промена резолуције слике и формата датотеке.

- Оптимизација за веб.

- Организовање фото-албума.

- Штампање растерске графике.

- Аллати за цртање.

- Палете боја.

- Ефекти, маске, исечање, брисање, копирање делова слике, подешавање осветљености, контраста, итд.

- Ретуширање и фото-монтажа

- Додавање текста.

- Израда ГИФ-анимације.

Пример програма за креирање векторске графике

- Подешавање радног окружења.

- Цртање основних графичких елемената – објеката.

- Трансформације објеката.

- Комбиновање објеката.

- Додавање текста.

- Комбинација растерске и векторске графике.

- Слојеви.

- Глобални преглед цртежа.

- Штампање векторске графике.

2. Мултимедија

- Начини представљања звука у рачунару.

- Основни формати записа звука (ВАВ, МПЗ, МИДИ).

- Програми за репродукцију звучних записа.

- Пример програма за снимање звука.

- Начини представљања видео-записа у рачунару.

- Основни формати видео-записа.

- Програми за репродукцију видео записа.

- Манипулација са видео записом у одабраном програму

- Увоз видео записа са уређаја.

- Пример програма за монтажу видео записа (комбинација слике, видео и звука).

- Конверзија видео формата.

- Постављање видео записа на веб.

3. Презентације на интернет

Рад са готовим веб дизајн решењима (CMS)

- Шта је CMS (Content Management System).

- Особине CMS-а.

- Најчешће коришћени CMS портали.

Презентација одабране теме коришћењем CMS портала.

4. Рачунарство и друштво

- Индустрија софтвера.

- Квалитет, тестирање и одржавање софтвера.

- Етичка питања.

- Лиценце.

- Питања безбедности.

- Утицај на друштво.

- Паметни градови.

III разред

(2 часа недељно, 70 часова годишње)

1. Примена рачунара у математици

- Програмска окружења намењена за примену у математици.

- Типови података, константе и променљиве, аритметички и логички оператори, наредбе и функције.

- Алгоритми за рад са низовима, листама и матрицама.

- Основни елементи нумеричке математике, анализа грешке, интерполација и апроксимација функција, нумеричко решавање једначина и методе Монте Карло.

- Дводимензионални и тродимензионални график функције.

- Комбинаторни и графовски алгоритми.

2. Примена рачунара у разним областима

- Вештачка интелигенција и машинско учење.

- Анализа података.

- Криптографија.

- Биомедицинска информатика.

- Критични инфраструктурни системи.

3. Рачунарска графика – напреднији курс

- Прављење основне анимације помоћу ефеката и готових поставки.
- Анимирање текста.
- Рад са слојевима облика.
- Анимација мултимедијалне презентације.
- Анимација слојева.
- Рад са маскама.
- Дисторзија објеката.
- Уклањање и промена боја.
- Израда 3D објеката.
- Употреба 3D алата.
- Представљање и извожење готовог формата.

НАЧИН ОСТВАРИВАЊА ПРОГРАМА

Настава се изводи у двочасу, са половином одељења у рачунарском кабинету.

На почетку наставе урадити проверу нивоа знања и вештина ученика, која треба да послужи као оријентир за организацију и евентуалну индивидуализацију наставе.

Активност треба да укључује практичан рад, примену ИКТ, повезивање и примену садржаја различитих наставних предмета, тема и области са којима се сусрећу изван школе. Активности осмислити тако да повећавају мотивацију за учење и подстичу формирање ставова, уверења и система вредности у вези са развојем језичке и информатичке писмености, здравим стиловима живота, развојем креативности, способности вредновања и самовредновања.

При реализацији програма дати предност пројектној, проблемској и активној настави, кооперативном учењу, изградњи знања и развоју критичког мишљења. Подстицати тимски рад и сарадњу нарочито у областима где наставник процени да су присутне велике разлике у предзнању код појединих ученика. Уколико услови дозвољавају дати ученицима подршку хибридном моделом наставе (комбинацијом традиционалне наставе и електронски подржаног учења), поготово у случајевима када је због разлика у предзнању потребна већа индивидуализација наставе.

I разред

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

- Основи информатике (12)
- Архитектура рачунарског система (4)
- Програмска подршка рачунара (14)
- Основе рада у оперативном систему са графичким интерфејсом (14)
- Обрада текста уз помоћ рачунара (18)
- Слајд-презентације (16)
- Рад са табелама (18)
- Интернет и електронска комуникација (12).

При реализацији тематске целине **Основи информатике** ученици би требало да се упознају са предметом изучавања информатике и рачунарства, са основним појмовима којима се ове области баве (појам информација, података, знања) и са основним областима информатике и рачунарства.

Потребно је затим увести појам дигитализације (дискретизације), објаснити како се у дигиталним уређајима све информације представљају (кодирају) помоћу бројева и продискутовати предности дигиталног у односу на аналогни запис. Ученици би требало да стекну представу о томе како се кодирају текстуалне, графичке и звучне информације. Увести појам бројевних основа (пре свега бинарне, декадне, хексадекадне и окталне) и приказати како је могуће број записати у некој од наведених бројевних основа и прочитати број записан у некој основни (уз помоћ дигитрона, али и без њега). Ученици треба да усвоје појмове бит, бајт, и редове величине за мерење количине информација.

Развој информационих технологија сагледати у контексту значаја развоја ових технологија за развој и ширење писмености

и развој људског друштва уопште. Подстаћи ученике да повезују развој ИКТ-а са темама из историје, математике, физике и осталим областима људске делатности. Из овог угла сагледати значај информатике, области примене рачунара (и њихов развој) и карактеристике информационог друштва. Не инсистирати на прецизном познавању чињеница (не анализирати детаљно перформансе одређених рачунара, не инсистирати на тачном познавању година увођења одређених технологија), већ ову причу учинити што занимљивијом и пријемчивијом ученицима.

При реализацији тематске целине **Архитектура рачунарског система** потребно је да ученици стекну основна информативна знања о структури (хардверу) и принципу рада рачунара без упуштања у детаље техничке реализације, електронске схеме и слично. Ученици би требало да знају да наведу и практично препознају из којих се компоненти састоји стони или преносни рачунар. При том, прва, површна класификација разликује кућиште, монитор, тастатуру, миш, штампач, док друга, из стручне перспективе много важнија, разликује процесор, меморије, улазно-излазне уређаје и магистрале које их повезују. Ученици би требало да умеју да објасне улогу процесора у функционисању рачунарског система (да познају појам радног такта, да разумеју улогу регистара, аритметичко-логичке и контролне јединице, да буду упознати са појмом процесорских инструкција и свођењу комплексних операција на низ елементарних инструкција), да објасне врсте и улогу различитих меморија у рачунарима и да разликују унутрашње меморије (кеш, RAM) од спољашњих, складишних меморија (хард-дискова, флеш-меморија, SSD уређаја, оптичких дискова). Инсистирати на хијерархијској организацији меморија и објаснити разлику у брзини, капацитету и цени различитих облика меморија. Објаснити основне врсте улазно-излазних уређаја и начине комуникације са њима. Описати и различите врсте магистрала и њихову улогу у остваривању комуникације између различитих компонената унутар рачунара.

Са ученицима заједно продискутовати карактеристике у том тренутку актуелне хардверске технологије (на пример, анализирати детаље хардверских конфигурација које се описују у огласима за продају рачунара). Ученици могу анализирати конфигурације школских рачунара (уз помоћ података доступних из оперативног система) и за домаћи им је могуће задати да анализирају конфигурације својих кућних рачунара. Ученицима је могуће приказати и поступак расклапања и склапања рачунара и указати им на једноставне кварове које могу сами препознати и отклонити.

Ученицима је могуће приказати и архитектуру и хардверске компоненте савремених мобилних уређаја (таблета, паметних телефона).

При реализацији тематске целине **Програмска подршка рачунара** потребно је да ученици стекну знања о значају програмске подршке (софтвера) за функционисање рачунара и утицају на могућности рачунара. Са ученицима продискутовати софтвер који они свакодневно користе и на основу тога формирати класификацију апликативног софтвера (на пример, софтвер за приступ услугама интернета, канцеларијски софтвер, софтвер за креирање и обраду мултимедијалних садржаја, рачунарске игре, ...). Увести затим појам системског софтвера и ученицима разјаснити појам и улоге оперативног система (ОС) из мало дубље перспективе него што је то само коришћење интерфејса ОС за основно управљање радом за рачунаром (не упуштају се у напредне детаље рада ОС, попут детаља распоређивања процесора, организације виртуелне меморије и слично). Описати и приказати и услужни софтвер (на пример, антивирусне програме, заштитне зидове и слично).

Ученицима скренути пажњу на појам власништва над софтвером, софтверских лиценци и заштите ауторских права. Описати разлику између власничког и слободног софтвера и софтвера отвореног кода. Описати и различите облике дистрибуције софтвера (пробне верзије, делимичне верзије). Ученицима (и на личном примеру) развијати правну и етичку свест о ауторским правима над софтвером, али и над подацима који се дистрибуирају путем мреже. Посебну пажњу посветити потреби коришћења лиценцираних програма, заштити програма и података, вирусима и заштити од њих.

При реализацији тематске целине **Основе рада у оперативном систему саграфичким интерфејсом** ученик треба да стекне знања, вештине и навике битне за успешно коришћење основних могућности оперативног система. Може се претпоставити да ученици већ умеју покрећу и користе апликативне програме (на пример, програме који су у саставу оперативног система за приказ мултимедијалних садржаја, уређење текста, цртање и једноставна нумеричка израчунавања, али и друге програме инсталиране на рачунар). Кроз неколико примера приказати поступак инсталације и уклањања апликативних програма, али и управљачких програма (драјвера) за одређене уређаје. Ученик би требало да уме да подеси основне параметре оперативног система (изглед окружења, датум и време, регионална и језичка подешавања, укључивање и искључивање приказа скривених датотека, честих екстензија датотека, подешавање подразумеваног програма за рад са одређеним типом датотека и слично).

Детаљно приказати рад са системом датотека (фајлова, докумената). Ученик треба да разликује намену датотека и директоријума (фасцикли, фолдера, каталога) и да познаје намену типова датотека који се најчешће користе (.txt, .jpeg, .avi, .docx, ...), да уме да из програма, коришћењем стандардних дијалога, учита и сними своје документе у одабраном формату на жељене локације, да пребаци документе са једног на други уређај или партицију диска, да хијерархијски организује своје документе коришћењем директоријума, да разликује логички и физички поглед на систем датотека (на пример, да познају положај фасцикле Desktop или Documents у вишекорисничком окружењу, да умеју да користе пречице и библиотеке), да примењује технике архивирања и компресије података, да уме да изврши основне операције са системом датотека из командне линије оперативног система (промени текући директоријум, прегледа његов садржај, копира, обрише или премести одређене документе и слично).

Скренути пажњу на честе концепте који се јављају током рада у окружењима са ГКИ и на нивоу оперативног система и на нивоу различитих апликативних програма (на пример, приказати концепт селекције и начине селектовања мишем и тастатуром, концепт клиборда и примену на копирање и пребацивање података). Ученик треба да зна да прати и да одреагује на најчешће поруке оперативног система (при брисању датотека и каталога; при затварању програма, а да није претходно сачуван документ, итд.). Пажњу посветити и питањима заштите (подешавања антивирусног програма и заштитног зида).

Неки елементи ове тематске целине се могу прожимати са другим тематским целинама. На пример, програм калкулатор (који се налази у оквиру оперативног система) се може користити када се уче бројевни системи, структура и перформансе конкретног рачунара се могу сагледати коришћењем података о уређајима добијених од оперативног система итд.

При реализацији тематске целине **Обрада текста на рачунару** потребно је да ученик стекне знања, вештине и навике неопходне за успешно коришћење програма за обраду текста. Ученицима приказати неколико различитих приступа обради текста (WYSIWYG приступ и језике за обележавање какви су LaTeX или HTML). Објаснити разлику између чистих текстуелних докумената креираних у текст-едиторима (.txt документи, обележени текстови, изворни кодови програма) и форматираних текстуелних докумената креираних у текст-процесорима. Нагласити да су основни кораци у раду са текстом (уношење текста, кретање кроз текст, копирање, исецање и премештање делова текста, претрага и замена) заједнички за широку класу програма који раде са текстом (текст-едиторе, текст-процесоре, разне апликативне програме, уобичајене контроле за унос текста). Инсистирати да ученици умеју вешто и ефикасно врше основне операције са текстом, коришћењем само тастатуре (да се крећу кроз текст карактер по карактер, реч по реч, пасус по пасус, да користе тастере Home и End, да селекују текст помоћу тастера Shift и кретања кроз текст, користе пречице за копирање, исецање и лепљење и слично). Скренути пажњу ученицима на вештину слепог куцања и мотивисати их да у самосталном раду савладају ту вештину.

Обучити ученике за рад у једном конкретном процесору текста. Ученик треба да уме да подеси радно окружење текст процесора, унесе текст (у ћириличком и латиничком писму), сачува унети текст, отвори постојећи текстуални документ, затвори активни документ, премешта садржај између више отворених докумената. Ученик треба да уме да изврши основно форматирање текста (да подеси димензије странице, маргине, да подеси својства пасуса, фонтове, својства карактера и слично). Ученик треба да уме да текст организује коришћењем листа (нумерисаних и нумерисаних), да у текст уметне и форматира табеле, да организује текст у секције и сложи га у више колона, да уметне у текст и исправно позиционира специјалне симболе, датум и време, слике, дијаграме, формуле, итд.

Пре преласка на рад са дужим и комплекснијим документима, потребно је објаснити разлику између логичке структуре докумената и њиховог визуелног и стилског обликовања и форматирања и увести стилове као основну технику логичког структурирања докумената. Инсистирати на томе да у свим дужим документима морају бити коришћени стилови (постојећи и кориснички дефинисани). У сложеније документе ученик треба да уме да уметне аутоматску нумерацију страница, да подеси подножја и заглавља страница, да аутоматски генерише садржај, индекс појмова, списак библиографских референци и слично. Ученике треба упознати са логичком структуром типичних докумената (молби, обавештења, итд.), школских реферата, семинарских и матурских радова и у свим вежбањима потребно је користити документе какви се срећу у реалном животу и инсистирати на њиховој униформности и прегледности, а не на усиљеним естетским подешавањима (избегавати документе који немају смислен садржај и који служе само да прикажу што више различитих могућности текст-процесора). За вежбу се може од ученика тражити да неформатирани дужи текст форматирају на основу датог узора (на пример, на основу датог документа у PDF формату).

Ученицима приказати алат за исправљање граматичких и правописних грешака, за коментарисање и обележавања измена у тексту и слично. Ученик треба да уме да прегледа текстуални документ пре штампе, подешава параметре за штампу и штампа.

На крају, ученицима је могуће приказати и рад у неколико различитих програма за обраду текста (нарочито, ако је неки од њих „у облаку“) и нагласити сличност са текст-процесором коришћеним у претходном току наставе.

При реализацији тематске целине **Слајд-презентације** потребно је да ученици схвате предности коришћења слајд-презентација у различитим ситуацијама, препознају ситуације у којима се може користити слајд-презентација, планирају и израђују адекватне презентације. При томе је потребно да знају основне етапе при развоју слајд-презентације, основне принципе доброг дизајна презентације (број информација по слајду, естетика, анимација у служби садржаја).

Ученике треба обучити коришћењу бар једног програма за креирање слајд презентација. Ученик треба да уме да подеси радно окружење, бира одговарајући поглед на презентацију, креира слајдове, поставља на њих текст и нетекстуалне објекте (слике, табеле, графиконе) доследно их форматира (користи мастер слајд). Ученик треба да уме да креира и интерактивне презентације које садрже линкове и акциону дугмад, да подешава анимације објеката на слајдовима и анимације преласка између слајдова, али те анимације треба да буду једино у функцији садржаја (избегавати анимације „по сваку цену“ које оптерећују презентацију).

Примери презентација треба да буду смислени, из реалног живота (најбоље је да се користе слајд-презентације у којима се обрађују теме из наставе, како информатике и рачунарства, тако и других предмета). Ученици неке презентације могу да креирају и у склопу домаћих задатака, а на часу је могуће анализирати презентације направљене код куће.

На крају, ученицима је могуће приказати још неколико програма за креирање слајд-презентација (нарочито, оне „у облаку“) и подвући сличности са програмом који је коришћен током наставе.

При реализацији тематске целине **Рад са табелама** објаснити основне појмове о програмима за рад са табелама (табела, врста, колона, ћелија, ...) и указати на њихову општост у програмима овог типа.

При уношењу података у табелу, објаснити разлику између различитих типова података (нумерички формати, датум и време), као и грешке које могу из тога да настану. Приликом манипулација са подацима (означавања ћелија, кретање кроз табелу, премештање, копирање, ...), указати на општост ових команди и упоредити их са сличним командама у програмима за обраду текста. Код трансформација табеле указати на различите могућности додавања или одузимања редова, или колона у табели. Објаснити појам опсега.

Код форматирања приказа податка у ћелији, приказати на примерима могућност различитог тумачења истог нумеричког податка (број, датум, време). Такође, нагласити важност доброг приказа података (висине и ширине ћелија, фонта, поравнања) и истицања појединих података или група података раздвајањем различитим типовима линија и бојењем или сенчењем.

Указати на повезаност података у табели и могућност добијања изведених података применом формула. Објаснити појам адресе и различите могућности референцирања ћелија. Указати на различите могућности додељивања имена подацима или групама података и предности коришћења имена.

Приказати функције уграђене у програм и обратити пажњу на најосновније функције, посебно за сумирање и сортирање.

Указати на различите могућности аутоматског уношења података у серији.

Посебну пажњу посветити различитим могућностима графичког представљања података. Указати на промене података дефинисаних у табели формулама, и графикону у случају измене појединих података у табели. Указати на могућност накнадних промена у графикону, како у тексту, тако и у размери и бојама (поздине слова, скале, боја, промена величине, ...).

Указати на важност претходног прегледа података и графикона пре штампања, као и на основне опције при штампању.

Све појмове уводити кроз демонстрацију на примерима. Од самог почетка давати ученицима најпре једноставне, а затим све сложеније примере кроз које ће сами практично испробати оно што је демонстрирао наставник.

Препоручљиво је да се сви нови појмови уведу у првих 10 часова тако што ће ученици радити задатке које је припремио наставник (текстуалним описом задатка или задатом коначном табелом, одштампаном, без увида у формуле) а затим ученицима дати конкретне мале пројекте различите природе: да направе електронски образац (на пример предрачун или нешто слично), прикупљање и обраду података који се односе на успех ученика из појединих предмета, неку појаву или процес из других наставних и ваннаставних области рада и интересовања ученика.

При реализацији тематске целине **Интернет и електронска комуникација** потребно је да ученици схвате предности умрежавања, разумеју у чему је разлика између рачунара-сервера и рачунара-клијента, који посао обављају Интернет-провајдери, карактеристике основних технологија приступа Интернету, адресирање на Интернету, принципе функционисања Интернета, чему служе Интернет протоколи, како да повеже рачунар са Интернетом. Потребно је да ученици науче да: визуализују веб-страницу коришћењем веб-читача, крећу се веб-пространством коришћењем адреса и хиперлинкова, проналазе и преузимају садржаје са веба, користе Интернет мапе; отворе и подесе налог електронске поште (вебмејл), шаљу и преузимају поруке и датотеке коришћењем електронске поште и сервиса за синхрону комуникацију, упознају се са правилима електронске комуникације (*netiquette*); упознају се са начином функционисања, правилима понашања, предностима и опасностима социјалних мрежа; упознају се са сервисима за дељење датотека на Интернету и појмом веб-апликације (*Cloud computing*); отворе налог и користе виртуелни телефон, (нпр. Скајп и сл.); разумеју појмове „електронска трговина” и „електронско банкарство”, науче како функционише електронски подржано учење. При свему овоме неопходно је перманентно радити на развијању

свести о важности поштовања правних и етичких норми при коришћењу Интернета, критичком прихватању информација са веба, поштовању ауторских права при коришћењу информација са веба, поштовању права приватности.

Препоруке за реализацију:

– при реализацији ове теме инсистирати на правопису и употреби одговарајућег језика тастатуре;

– полазна тачка при упознавању локалних мрежа треба да буде конкретна школска мрежа на којој се могу илустровати њене саставне компоненте, топологија, ресурси, клијент-сервер, итд.;

– локалне мреже, након упознавања, треба ставити у контекст Интернета (мреже свих мрежа) и кооперативног коришћења расположивих информационих ресурса;

– вежбе крстарења (енгл. *surf*) и претраживања требало би да су у функцији овог, али и других предмета, како би се код ученика развијала навика коришћења Интернета за прикупљање информација за потребе сопственог образовања;

– преузимање датотека са веба вежбати на датотекама разних типова (текст, слика, видео-клип);

– ученицима треба објаснити како раде претраживачки системи и о чему треба водити рачуна да би се остварила ефикаснија претрага;

– при реализацији ове теме инсистирати на могућностима злоупотребе Интернета а поготово на безбедности корисника, поузданости информација и спречавању злоупотребе деце;

– при претраживању Интернета ученике усмерити ка тражењу образовних веб сајтова и коришћењу система за електронско учење;

– при обради електронског пословања демонстрирати различите врсте веб образаца који се користе за поручивање и плаћање робе путем Интернета, поручивање докумената...;

– показати рад са текстом, рад са табелама и складиштење података у неком од сервиса „у облаку” (*cloud computing*), доступних путем веба;

– при реализацији садржаја везаних за електронску пошту објаснити ученицима „пут” електронског писма.

II разред

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Рачунарска графика (28)

Мултимедија (18)

Презентације на интернету (18)

Рачунарство и друштво (8).

При реализацији теме **Рачунарска графика** објаснити разлику између векторског и растерског представљања слике, предности и недостатке једног и другог начина. Објаснити основне типове формата слика и указати на разлике међу њима. Указати на постојање библиотеке готових цртежа и слика.

При увођењу појмова растерске и векторске графике, нека ученици на својим рачунарима паралелно отворе прозоре програма за цртање који је у саставу оперативног система и нпр. текст-процесор, рећи им да у оба нацртају елипсу и максимално зумирају, нацртају затим обојени квадрат преко дела елипсе и покушају да га „преместе”, при свему томе захтевати од њих да изводе закључке у вези са карактеристикама једне и друге врсте графике. Направити паралелу између ове две врсте графике у односу на цртеже воденим бојама и колаже од папира. Код наставне јединице која се односи на формате датотека илустровати конкретним примерима, урађеним од једне фотографије, зумирати слике, поредити величине датотека.

Припремити за часове дигитални фото-апарат или мобилни телефон са камером и на часу правити фотографије. На претходном часу дати ученицима задатак да донесу фотографије које ће на часу скенирати. Ученици могу на својим фотографијама да увежбавају технике основних корекција и обраде фотографије: уклањање „дрвених очију”, ретуширање, фото-монтажу, промену резолуције и формата слике, а затим направе фото-албум свих радова.

Посебну пажњу посветити пројектовању цртежа (подели на ниво, уочавању симетрије, објеката који се добијају померањем, ротацијом, трансформацијом или модификацијом других објеката итд.), као и припреми за цртање (избор величине и оријентације папира, постављање јединица мере, размере, помоћних линија и мреже, привлачења, углова, итд.).

Код цртања основних графичких елемената (дуж, изломљена линија, правоугаоник, квадрат, круг, елипса) објаснити принцип коришћења алатки и указати на сличности са командама у различитим програмима. Слично је и са радом са графичким елементима и њиховим означавањем, брисањем, копирањем, груписањем и разлагањем, премештањем, ротирањем, симетричним пресликавањем и осталим манипулацијама. Указати на важност поделе по нивоима и основне особине нивоа (видљивост, могућност штампања, закључавање).

Код трансформација објеката обратити пажњу на тачно одређивање величине, промену величине (по једној или обе димензије), промену атрибута линија и њихово евентуално везивање за ниво. Посебно указати на разлику отворене и затворене линије и могућност попуњавања (бојом, узорком, итд.).

Указати на важност промене величине приказа слике на екрану (увећавање и умањивање цртежа), и на разлоге и начине освежавања цртежа. Код коришћења текста указати на различите врсте текста у овим програмима, објаснити њихову намену и приказати ефекте који се тиме постижу.

Код штампања указати на различите могућности штампања цртежа и детаљно објаснити само најосновније. За увежбавање дати ученицима конкретан задатак да нацртају грб школе, свог града или спортског друштва, насловну страну школског часописа, рекламни пано и сл.

Обраду теме **Мултимедија** засновати на искуствима ученика, резимирати њихова знања, запажања и искуства у раду са звуком и видеом. Направити упоредни преглед неколико програма за репродукцију звука. При упознавању са основним форматима записа звука, направити паралелу између растерске и векторске графике са једне стране и снимљеног и синтетичког звука са друге стране. Дати ученицима прилику да сниме сопствени глас и репродукују га. Повезати са темом израде презентација у првом разреду и могућношћу снимања нарације уз слајдове. Направити упоредни преглед неколико програма за репродукцију видео-записа.

Рад са видео-записима засновати на видео радовима ученика направљених на часу или припремљених унапред (у виду домаћих задатака). Потребно је да ученици савладају основне технике монтаже видео материјала, звука, ефеката и натписа.

При реализацији тематске целине **Презентације на интернету** потребно је упознати ученике готовим веб решењима који се бесплатно могу наћи на интернету, преузети и користити у личне и комерцијалне сврхе, а обједињени су под називом CMS; основним одликама и предностима CMS портала. Нагласити главне особине CMS портала: лакоћа и једноставност уређивања где није потребно велико познавање веб-технологија, изглед портала се мења коришћењем тема које се врло често ажурирају тако да администратор има велику могућност избора. Упознати ученике са данас најпопуларнијим CMS решењима: WordPress, Joomla, Weebly и Google Sites.

Препоруке за реализацију:

– при реализацији ове тематске целине подстаћи ученике да примене стечена знања из рада са програмима за обраду текста и програмима за обраду слика и да тако припреме садржај за креирање и администрирање веб-портала;

– подстаћи ученике да одабране теме презентују у облику веб-портала (блага...), при бирању садржаја инсистирати на критичком приступу информацијама и неговању естетике.

У оквиру теме **Рачунарство и друштво** скренути пажњу ученицима на друштвене промене које је изазвала информациона револуција и на позитивне промене које су ИКТ донеле у све сегменте људског живота.

Указати на лепо понашање на интернету (netiquette), правилно писање и изражавање и правила лепог понашања у комуникацији,

као и на етичка и правна питања приступа садржајима (ауторска права, лиценце). Објаснити појам софтверских лиценци и информативно их упознати са најчешће коришћеним типовима лиценци. Објаснити разлику између слободног софтвера и софтвера отвореног кода и бесплатног софтвера.

Развити код ученика свест и о опасностима и неопходним мерама заштите здравља од претеране и неправилне употребе рачунара као и о питањима безбедности и приватности приликом употребе уређаја, нарочито у данас неизбежном мрежном окружењу.

Посебну пажњу обратити на проблематику поштовања правних и етичких норми при коришћењу Интернета, критичком прихватању информација са веба, на проблематику ауторских права и етичких норми при коришћењу туђих звучних и видео записа, као и на поштовање права на приватност особа које су биле актери снимљених материјала и тражење њихових дозвола за објављивање. Продискутовати важна питања безбедности и приватности на интернету у контексту употребе друштвених мрежа.

Такође, у оквиру ове тематске целине потребно је са ученицима отворити дискусију о tzv. паметним градовима, тачније о софтверским системима који на паметан начин управљају инфраструктуром у градовима. Објаснити улогу, значај и друштвени утицај ових софтверских система у садашњости и будућности.

III разред

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Примена рачунара у математици (34)

Примена рачунара у разним областима (12)

Рачунарска графика – напреднији курс (24).

У оквиру теме **Примена рачунара у математичким израчунавањима** применом одговарајућих програмских окружења (*Matlab, Mathematica, Octave, Scilab* и сл.) извршити основна математичка израчунавања, моделовање и симулације, анализу и обраду података, графичко приказивање резултата и развој алгоритама. Упознати ученике са типовима података, константама и променљивима, аритметичким и логичким операторима, наредбама и функцијама, алгоритмима за рад са низовима, листама и матрицама. Објаснити основне елементе нумеричке математике, анализу грешке, интерполацију и апроксимацију функција, нумеричко решавање једначина и методе Монте Карло, као и графички приказ континуалне и дискретне функције помоћу дводимензионалног и тродимензионалног графика. Размотрити основне комбинаторне и графовске проблеме, као и могућности за њихово решавање.

У оквиру тематске целине **Примена рачунара у разним областима** упознати ученике са одабраним темама из вештачке интелигенције и машинског учења (неуронске мреже, фази логика, генетски алгоритам и сл.), анализе података (eng. data mining), криптографије (алгоритми у криптографији), биомедицинске информатике (обрада података у биологији и медицини) и критичних инфраструктурних система (паметно управљање инфраструктуром). Ова тематска целина треба да буде прегледног катактера, а њен главни циљ је да код ученика створи свест о веома широкој употреби рачунарства у свету у коме живимо.

Вештачка интелигенција и машинско учење

Објаснити појам вештачке интелигенције и навести успешне примере савремених система вештачке интелигенције. Упознати ученике са појмом неуронска мрежа, моделима неуронских мрежа, могућностима примене. Дати преглед неуронских мрежа према начину повезивања, у односу на правац преноса и обраду података и у зависности од начина учења (са надгледаним и ненадгледаним учењем). Објаснити начин рада неуронске мреже, слојеве мреже и поступак учења. Упознати ученике са филозофским и етичким питањима везаним за постојање вештачке интелигенције.

Објаснити појам меког рачунарства, чему служи фази логика и принципе фази логике. Навести предности њеног коришћења у неуронским мрежама, као и области где се може користити фази логика.

Упознати ученике са променама које је донела примена машинског учења (решавање класификационих проблема, анализа података у реалном времену и предвиђање будућих трендова). Навести основне технике машинског учења (стабла одлучивања, вештачке неуронске мреже).

Data mining

Објаснити основну сврху анализе података као проналажење скривених односа међу подацима, повећање њихове употребљивости и трансформацију у корисно знање. Објаснити укратко које све области *Data Mining* обухвата, као и кораке које подразумева (сакупљање података, филтрирање података и трансформација, креирање и избор модела, процена квалитета модела, креирање извештаја, оцењивање модела, имплементација *data mining* модела у апликацију, управљање моделом).

Криптографија и заштита података

Дати дефиницију криптографије, поделу на симетричне и асиметричне криптографске системе и делове савремених криптографских система. Објаснити примену теорије бројева у криптографији и најпознатије криптографске системе са јавним кључем.

Биомедицинска информатика

Информативно ученике упознати са типовима модела информационог процеса у медицини и са структуром апликација које се користе у медицини у зависности од удела човека у корацима који се подразумевају. Дати преглед основних функција здравственог информационог система.

Паметни градови

Упознати ученике информативно са идејом паметних градова и циљем очувања животне средине кроз истовремени технолошки и друштвени напредак грађана једног таквог града. Од ученика тражити да у групама, тимски дефинишу области на којима би таква један напредак требало очекивати и да разраде кораке интеграције и формирања концепта паметног града.

При реализацији теме **Рачунарска графика – напреднији курс** потребно је све примере урадити у одговарајућем професионалном програму (*Adobe After Effects*) за графику или у некој од бесплатних верзија ових програма.

Све спољне датотеке увозити као композиције (у више слојева) да би се слојеви мењали засебно. При томе урадити и пример увозења слојевите датотеке у облику једне слике. Курс започети једноставним пројектом у виду једне композиције која се састоји од једног или више слојева, распоређених у панелу *Composition* и у панелу *Timeline*. Испробати на слојевима ефекте замућености (*Radial Blur*), сенке (*Drop Shadow*), ефекат изоштравања ивица објекта (*Color Emboss*), корекцију боје (*Exposure*), својство *Transform*. Објаснити прекомпоновање слојева при пребацивању у другу композицију и својство за додавање провидности слоју (*Opacity*), као и постављање слојева у статус родитељског и подређеног слоја.

Објаснити једноставну анимацију текста, урадити позиционирање текста и применити ефекте из готове поставке анимације на текст. При анимирању текста урадити анимацију размака између знакова и подешавање провидности знакова. Употребом аниматора текста анимирати у времену својства појединачних знакова у текстуалном слоју.

Урадити примере са цртањем различитих облика: правоугаоник (*Rectangle*), заобљени правоугаоник (*Rounded Rectangle*), елипса (*Ellipse*), многоугао (*Polygon*) и звезда (*Star*). Урадити примену градијентне попуне са облицима. Помоћу алатке *Polygon* направити и друге облике који не спадају у основне, применити ефекте увијања (*Twist*), понављања облика (*Repeater*) и ротирања облика (*Rotation*).

Показати рад са маскама са изабраним конкретним слојем. Представити случајеве када са алатком за цртање маска црта као произвољна путања и пример када се маска одређује алатком за облике. Урадити пример додавања слоја 3D светла.

Представити алатке за додавање природног кретања у растерске слике и векторску графику (алатке *Puppet*). Користити алате за дефинисање тачке деформисаности, за означавање делова слике које су у првом плану код преклапања области и алате које служе за „замрзавање” неких делова слике. У овом делу идеалан пример за вежбу је силуэта човека који се оклизне, падне или креће по одређеној подлози.

Објаснити поступак за уклањање, ублажавање боја у видео материјалима дефинисањем провидности одређеној вредности боје, као и коришћењем алатки за уклањање појаве „преливања светлости”. Урадити дотеривање баланса боја, замену позадине и уклањање одређеног опсега боја.

Додавањем треће осе одабраном слоју показати својства карактеристична за 3D слојеве. Објаснити улогу слоја који служи за гледање 3D слоја, камере. Урадити примере 3D модела књиге и 3D текста, ротацију објекта, додавање одраза облицима. Посебно објаснити додавање светла, урадити примере са четири врсте додатог светла паралелно, снап, акцентујуће и амбијентално. Анимирајући светло да се појављује у различито време. На слојевима испробати ефекте транзиције, замућености покрета и временско подешавање слојева.

Код сваке теме објаснити додавање звучних и видео датотека на почетку, одсецање радног подручја, завршну фазу обраде композиције и примену ефекта за стилизовање изгледа видеа (*Cartoon*) и ефекта *Brainstorm*. Показати извожење композиције у жељени формат, извожење у друге графичке програме и у формат за веб.

ПРОГРАМИРАЊЕ

Циљ и задаци

Циљ наставног предмета Програмирање је развој алгоритамског приступа решавању проблема код ученика, овладавање техникама програмирања и стицања знања о савременим програмским језицима.

Задаци наставе предмета Програмирање су да ученици:

- стекну основна знања о унутрашњој организацији рачунара и начину извршавања програма;
- овладају свим основним, али и неким напреднијим концептима програмирања;
- упознају се са различитим приступима решавању проблема програмирањем;
- унапреде своје способности за брзо, ефикасно и рационално проналажење, складиштење и преношење информација коришћењем рачунара, као и да стекну свест о потреби за критичким приступом и потреби за пажљивим анализирањем информација;
- јачају способност за прецизно и концизно дефинисање проблема;
- овладају широким дијапазоном основних рачунарских алгоритама;
- разумеју и примењују принципе креирања модуларних и добро структурираних програма;
- савладају технику креирања апликација са графичким корисничким интерфејсом и основне принципе њиховог функционисања (програми вођени догађајима);
- упознају се са теоријом израчунљивости, појмом сложености алгоритама и напредним алгоритмима који решавају тешке проблеме;
- овладају правилним креирањем, анализом и трансформацијом променљивих декларисаних као сложени типови података (низови, структуре, листе, стабла).

I разред

(1+2 часа недељно, 108 часова годишње + 30 часова наставе у блоку)

САДРЖАЈИ ПРОГРАМА

1. Појам и примери алгоритама

- Појам и примери алгоритама.
- Начини описа алгоритама.
- Блоквско програмирање.

2. Основни концепти програмских jezika и окружења за развој програма

- Алатке и интегрисана окружења за развој програма.
- Основни елементи синтаксе и семантике процедуралних jezika (променљиве, константе, типови, изрази, наредбе, потпрограми).
- Интерфејс програма (КЛИ и ГКИ).

3. Основни алгоритми линијске и разгранате структуре

- Имплементација формула.
- Целобројна аритметика.
- Једноставно гранање и логички изрази.
- Сложено (угнежђено, хијерархијско) гранање.

4. Основни алгоритми цикличке структуре

- Линеарна обрада серија података (учитавање, испис, статистике (број, збир, производ, минимум, максимум, ...), филтрирање, пресликавање, претрага).
- Угнежђене петље.

5. Детаљни преглед основних типова података (променљивих, константи, оператора и израза)

- Бројевни типови (скуп вредности, константе, оператори, библиотечке функције).
- Конверзије типова (имплицитна, експлицитна).
- Карактерски тип (константе, оператори, библиотечке функције).

6. Низови, ниске и основни алгоритми за рад са њима

- Једнодимензиони низови (алокација меморије, индексни приступ елементима, пренос између потпрограма).
- Ниске карактера (stringови).
- Попуњавање и анализа садржаја низова.
- Трансформације низова (уметање и избацивање елемената, филтрирање, сортирање).
- Низови као репрезентација полинома и великих бројева.
- Низови као репрезентација вектора.

II разред

(1+2 часа недељно, 108 часова годишње + 30 часова наставе у блоку)

САДРЖАЈИ ПРОГРАМА

1. Вишедимензиони низови, матрице и основни алгоритми за рад са њима

- Вишедимензиони низови (матрице).
- Анализа садржаја матрица.
- Трансформације матрица.
- Матрице као математички објекти.

2. Кориснички дефинисани типови

- Набројиви типови, интервални, скуповни типови.
- Структурни типови, унијски типови.
- Низови и матрице структура.

3. Улаз и излаз програма

- Датотеке (текстуалне, бинарне).
- Аргументи командне линије.

4. Анализа алгоритама

- Анализа коректности.
- Анализа просторне и временске сложености.

5. Опште технике конструкције алгоритама

- Рекрузија.
- Претрага са повратком.

- Алгоритми грубе силе.
- Грамзиви алгоритми.
- Динамичко програмирање.
- Подели па владај.

6. Динамичке структуре података и апстрактни типови података

- Показивачи и динамичка алокација меморије.
- Структуре података (статички и динамички низ, листа, дрво, хеш-табела, граф).
- Апстрактни типови података (стек, ред, ред са приоритетом, скуп, речник, дисјунктни скупови).

III разред

(2 часа недељно, 70 часова годишње + 30 часова наставе у блоку)

1. Графови и алгоритми за рад са графовима (20 часова)

- Појам и репрезентација графова.
- Обилазак графа (у дубину и у ширину).
- Тополошко сортирање.
- Најкраћи путеви у графу (Дајкстрин и Флојд-Варшалов алгоритам).
- Минимално разапињуће дрво.

2. Алгоритми текста (10 часова)

- Проналажење подниске (грубом силом, Кнут-Морис-Пратовим или Бојер-Муровим алгоритмом).
- Регулари изрази и примена.
- Контекстно-слободне граматике и примена. Рекурзивни спус.

3. Геометријски алгоритми (10 часова)

- Површина конвексног и простог многоугла.
- Провера припадности тачке конвексном многоуглу.
- Конвексни омотач скупа тачака.

4. Алгоритми теорије бројева (10 часова)

- Проширен Еуклидов алгоритам и примене.
- Тестови прималности.
- Растављање броја на просте чиниоце и примене.

5. Алгоритми над битовима (4 часа)

6. Преглед одабраних структура података и алгоритама (16 часова).

НАЧИН ОСТВАРИВАЊА ПРОГРАМА

I разред

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Појам и примери алгоритама (20 часова)

Основни концепти програмских jezika и окружења за развој програма (10 часова)

Основни алгоритми линијске и разгранате структуре (20 часова)

Основни алгоритми цикличке структуре (30 часова)

Детаљни преглед основних типова података (променљивих, константи, оператора и израза) (8 часова)

Низови, ниске и основни алгоритми за рад са њима (20 часова)

Предвиђена су три двочасовна писмена задатка са једночасовним исправкама (9 часова).

У оквиру наставне теме **Појам и примери алгоритама** упознати ученике са неформалним појмом процедуралног алгоритма (низа корака чијим се доследним спровођењем решава неки проблем). Истакнути значај алгоритмизације и алгоритамошког начина размишљања у савременом свету. Ученицима приказати разне

начине за opisivanje algoritama (псеудокод, дијаграме тока, MIT Scratch/Blockly дијаграме, програмски језик). Повући паралеле, исте једноставне алгоритме описати на разне начине и скренути пажњу на различит ниво детаљности који разни формализми захтевају (објаснити који су то детаљи који разликују псеудокод од кода у правом програмском језику). Појам алгоритма приближити ученицима кроз што већи број примера из свакодневног живота, пре свега алгоритма познатим ученицима из математике, физике и хемије. Инсистирати да ученици прво корак-по-корак спроводе унапред задате алгоритме и тек након тога од њих тражити да постепено сами почну да описују у почетку веома једноставне алгоритме. Већ приликом првих алгоритма истакнути основне концепте алгоритма: променљиве, доделу, наредбе, секвенцијално извршавање наредби, гранање и петље.

У оквиру ове теме ученике упознати и са неким окружењем за учење програмирања заснованом на визуелном (блоковском) програмском језику (Scratch, code.org, Alice, Greenfoot, Karel, Logo, ...). У оквиру таквог окружења са ученицима реализовати одређен број елементарних алгоритама, постепено повећавајући њихову комплексност, али задржавајући се на алгоритмима који се у датом окружењу могу једноставно имплементирати (на пример, Scratch не подржава могућност дефинисање класичних функција и требало би избегавати алгоритме који су такви да се исти блокови копирају између разних ликова, што је веома лош стил програмирања и може имати лош утицај на касније изучавање програмирања).

Увести и појам алгоритамске декомпозиције као поделе крупнијег корака на ситније и једноставније поткорацие.

У оквиру теме **Основни концепти програмских језика и окружења за развој програма** ученике кроз примере израде једноставнијих програма упознати са програмским окружењем и основним концептима програмског језика који ће се у наставку изучавати.

Упознати ученике са процесом креирања конзолних апликација (апликација са командно-линијским интерфејсом, КЛИ), читавањем појединачних бројева са улаза и исписом текста и бројева на излаз.

Увести појам типова променљивих и њихове декларације, задржавајући се само на два основна типа (целобројном типу, попут `int tj`. Integer и реалном типу `double tj`. Real). Увести појам израза и основне аритметичке операције (множење, сабирање и одузимање и реално дељење, док се целобројно дељење због својих специфичности може одложити за касније). Увести појам наредбе и кроз веома једноставне програме демонстрирати основне наредбе (доделу, гранање, основне петље). Појам петље, као најтежи од поменутих увести на веома једноставним примерима (на пример, исписивање одређеног текста више пута или исписивање свих бројева од 1 до n) и његову разраду оставити за касније.

Увести појам потпрограма (функције, процедуре) и описати њихово коришћење као основне технике за алгоритамску декомпозицију, добијање разумљивијих програма и избегавање понављања програмског кода. Дати примере функција са параметрима, повратном вредношћу и вишеструким повратним вредностима (реализованих преко излазних параметара).

У случају да се наставник одлучи за креирање апликација са графичким корисничким интерфејсом (ГКИ) од самог почетка, потребно је ученицима описати фазу дизајна интерфејса и фазу програмирања апликације, увести потребне појмове објектно оријентисаног програмирања (у најмањој мери, без приче о напредним концептима ООП какви су наслеђивање и полиморфизам, а који заправо нису потребни да би се користили готови објектни потребни за креирање интерфејса), увести појам догађаја и реакције на догађаје и навести и описати најчешће коришћене контроле. Све време инсистирати на јасној сепарацији основне функционалности програма и функционалности интерфејса. Централне теме наставе програмирања треба да буду концепти који су заједнички за све императивне програмске језике и стога би требало избегавати инсистирање на специфичностима библиотеке језика намењене креирању ГКИ (нема потребе користити сувише напредне контроле, њихова специфична својства, нити специфичне догађаје).

Тему креирања апликација са ГКИ могуће је обрадити касније током првог разреда, па чак и у другом или у трећем разреду. Уколико се осети потреба за тим, наставници који на почетку описују и креирање апликација са ГКИ могу посветити и мало више часова на рачун ове уводне теме, док наставници коју ту тему остављају за касније могу уводну тему мало скратити.

Што се тиче окружења, ученицима приказати поступак уноса/уређивања, превођења и покретања програма. При том описати и превођење (појам и намену) указујући ученицима кроз примере на најчешће грешке које се при том пријављују. Укратко описати и појам и намену повезивања, начин употребе и грешке које се најчешће пријављују у тој фази (недефинисани симбол или вишеструки симболи). Описати и појам пројекта и изградња програма (из окружења, али и ван њега). Приказати ученицима и извршавање програма корак по корак (са и без уласка у потпрограма), извршавање до зауставне тачке и дебаговање (праћење вредности променљивих) – из окружења (ако оно то допушта) или из посебног дебагера.

Још једна могућност је да се у оквиру ове главе користе библиотеке специјално припремљене од стране наставника које омогућавају једноставнији и интересантнији увод у основне концепте програмског језика. Примери таквих библиотека су библиотека за цртање наредбама за померање објекта који црта по екрану (тзв. корњача графика), библиотека за шетање робота по лавиринту (нпр. робот Karel) и слично. Овај приступ сличан је оном који се користи на сајту `code.org`, међутим уместо блокова програмирање се врши у класичном реалном текстуалном програмском језику. Иако је у принципу могуће и коришћење специјализованих текстуалних језика за учење програмирања, њих не би требало користити предуго и ученике би што пре требало изложити синтакси и семантици језика који ће се користити током наставка курса.

Тему **Основни алгоритми линијске и разгранате структуре** започети имплементацијом програма који решавају задатке применом формула из математике физике и хемије. То су програми засновани на формулама за рачунање геометријских мера (обима, површина, запремина), формулама за рачунање параметара кретања (равномерног и равномерно убрзаног), формулама заснованих на пропорцијама и слично. Поред примене готових формула урадити и одређени број примера у којима се захтева извођење формуле којом се од датих података израчунава непознати податак (на пример, задатке у којима се решење добија постављањем и решавањем неке једноставне једначине или мањег система једначина). При решавању ових задатака увести и користити елементарне математичке библиотечке реалне функције и константе (на пример, кореновање, степеновање, апсолутна вредност, константа π и слично).

Где год је то пригодно увести и користити помоћне функције (на пример, функцију за израчунавање растојања између две тачке која се више пута користи у оквиру функције за израчунавање површине троугла чије су координате теме на познату основу Хероновог обрасца). Ако је потребно на теми функција се задржати и мало дужи и детаљније продискутовати све релевантне аспекте (улазне, излазне и улазно-излазне параметре, досег tj . однос између локалних и глобалних променљивих и мане употребе глобалних променљивих, однос између функција и метода и слично).

Посебну пажњу посветити теми целобројног дељења (одређивања количника и остатка) и применама (на пример, свођење разломака на мешовити број). Анализирати заокруживање количника наниже (на пример, одредити највећи број парова који се могу формирати од датог броја ученика) и навише (на пример, одредити најмањих број возњи лифтом потребних да превезе дати број људи ако у лифт стаје 4 човека). Посебно приказати технике заокруживања на целобројном, а посебно на реалном типу. Приказати технике модулларне аритметике tj . сабирања, множења и одузимања бројева по датом модулу (модулларни инверз ни дељење по модулу, као напредније операције не обрађивати сада). Приказати алгоритме за рад са цифрама у декадном запису бројева (троцифрених, четвороцифрених) – издвајање цифре на датој позицији, издвајање свих цифара почевши од цифре јединица, замена цифре на датој позицији, размена цифара, формирање броја на основу

датих цифара (класичан полином), формирање броја на основу цифара слева (Хорнерова шема), формирање броја на основу цифара с десна, сабирање бројева датих цифарама, одузимање бројева датих цифарама и слично. Уопштити на позициони запис бројева у произвољној бројевној основи (на пример, октални запис). Посебно обрадити бројевну основу 60 (запис времена и запис углова), као и мешовите бројевне основе (нпр. 24, 60, 60, 100 – дани, сати, минути, секунди, милесекунди). Приказати алгоритме за рад са временом и угловима (нпр. разлика између тренутка завршетка и почетка, сабирање два угла по модулу пуног круга и слично) и то помоћу технике конверзије у најмању јединицу и назад (нпр. конверзије угла задатог у степенима, минутима и секундама у угао задат само у секундама и назад), али и директно, применом алгоритма за рад над бројевима задатим својим цифрама у позиционом запису (нпр. сабирање углова сабирањем секунди, минута и степени уз вршење преноса са претходних позиција).

Имајући у виду да су најрудиментарнији програми са гранањем уведени у склопу прегледа програмског језика, као и да се гранање у елементарном облику могло користити и током изучавања алгоритма доминантно линијске структуре, детаљнији преглед алгоритма заснованих на гранању започети програмима у којима се резултат одређује на основу више услова, које је најчешће потребно повезати одређеним логичким операторима (на пример, испитати да ли унети бројеви могу представљати странице троугла, да ли је унета година преступна, да ли су два унета броја истог знака, да ли две тачке припадају истом квадранту и слично).

Објаснити сложено (угнежђено) гранање и његове најчешће облике. Објаснити проблем придруживања `else`. Посебно истакнути конструкцију `else-if` и њену употребу. Приказати примере хијерархијског гранања (на пример, одређивање квадранта или осе којем припада дата тачка, дискусија броја решења линеарне или квадратне једначине на основу коефицијената, стабло одлучивања за одређивање непознате животиње на основу неколико датих карактеристика и слично). Приказати гранање на основу дискретне вредности (на пример, име месеца на основу редног броја) и реализацију помоћу различитих наредби и облика гранања. Приказати гранање на основу припадности интервалама реалне праве (на пример, одредити агрегатно стање воде на основу дате температуре, оцену на испиту на основу датог броја поена, школски успех на основу просечне оцене и слично). Приказати лексикографско поређење n -торки вредности (на пример, упоредити два времена или датума, упоредити такмичаре на основу броја поена, а затим, у случају нерешеног резултата, на основу времена потребног да заврше задатке).

Посебну пажњу обратити на поређење две вредности и на уређивање две вредности по величини (са посебним нагласком на размену вредности променљивих). Приказати функције за одређивање минимума и максимума два броја (библиотеке, ако постоје и ручно имплементирани). Приказати примене ових функција (на пример, одређивање пресека и уније два интервала реалне праве, површине пресека два правоугаоника чије су странице паралелне координатним осама, максимума три броја у облику $\max(a, b, c)$ и слично).

У склопу изучавања гранања приказати ученицима и условни израз (ако то програмски језик подржава) и адекватне начине његове употребе.

У теми **Основни алгоритми цикличке структуре** потребно је ученицима увести појам итерације тј. поступака који се понављају одређени број пута (фиксиран број пута или све док је неки услов испуњен). Да би ученици лакше усвојили овај концепт најбоље је у почетку приказати алгоритме обраде малих серија елемената фиксираних дужине (три, четири или пет елемената). На почетку приказати алгоритме одређивања статистика таквих серија бројева: збира, производа, просека, максимума и минимума. Поред очигледног начина одређивања збира елемената формирањем сложеног израза приказати и поступно израчунавање збира (иницијализацијом на нулу или на први члан серије и додавањем једног по једног елемента серије). Исти принцип применити на израчунавање производа и искористити као увод у представљање алгоритма

одређивања минимума и максимума мале серије бројева (иницијализација резултата на вредност првог члана, и затим итеративно ажурирање резултата одређивањем минимума тј. максимума дотадашњег резултата и текућег члана серије). Максимум и минимум серије реализовати и коришћењем функције за одређивање максимума и минимума две вредности, али и без тога, коришћењем наредбе гранања. Дискутовати предности итеративног приступа у односу на одређивање минимума/максимума три или четири броја угнежђеним, хијерархијским гранањем. Уколико то језик подржава, приказати и библиотечку функцију за одређивање поменутих статистика малих серија елемената. Примери малих серија могу бити бројеви који се читавају са улаза, али и цифре троцифрених и четвороцифрених бројева (алгоритам њиховог одређивања обрађен је раније). Приказати и да се исти алгоритми могу спроводити и на серијама које нису чисто нумеричке већ се могу или неким пресликавањем свести на нумеричке или поредити у односу на неку релацију поретка (на пример, одредити маратонца који је постигао најбољи резултат превођењем времена у секунде или лексикографским поређењем времена). Приказати и начине одређивања позиције максималног/минималног елемента. Након доброг утврђивања поступка одређивања статистика малих серија, прећи на веће серије и серије чија величина није унапред фиксирана. Демонстрирати начине генерисања таквих серија (учитавање n бројева са стандардног улаза, учитавање бројева све док се не унесе нула, серије узастопних природних бројева, серије елемената аритметичког и геометријског низа, попут равномерног размакнутих тачака датог интервала реалне праве и слично). Посебно истакнути одређивања серије цифара у декадном запису природног броја (целобројним дељењем са 10 све док се број не сведе на нулу). Након тога приказати како се алгоритми одређивања статистика имплементирају помоћу петљи.

Обрадити алгоритам пресликавања серија (на пример, испитати таблицу квадрата и корена првих n природних бројева, табелирати реалну функцију једне променљиве на неком интервалу) и статистике пресликане серије (на пример, интензитет вектора као суму квадрата његових компонената, хармонијску средину серије бројева и слично).

Обрадити алгоритам филтрирања серије тј. одређивања свих елемената серије који задовољавају неки услов (на пример, одредити све непарне позитивне бројеве учитане са улаза). Елементе филтриране серије или исписивати (на пример, исписати све делиоце броја) или комбиновати филтрирањем са пресликавањем и одређивањем статистика (на пример, пронаћи збир квадрата свих непарних цифара у декадном запису датог природног броја или пребројати све троцифрене бројеве чији је збир цифара дељење са k).

Посебно приказати алгоритам линеарне претраге којим се проверава да ли у серији елемената постоји елемент који задовољава дато својство, односно, дуално, да ли сви елементи задовољавају дато својство. Дискутовати и варијанте у којима се тражи најмањи или највећи елемент који задовољава дато својство или се тражи његова позиција у серији. Посебну пажњу скренути ученицима на могућност прекида петље након проналажења траженог елемента и начине имплементације тог прекида (наредбом прекида петље, ојачањем услова логичком променљивом и слично). Приказати и класичне алгоритме који су засновани на претрази (нпр. провера да ли је дата серија елемената сортирана, провера да ли је број прост која комбинује претрагу постојања делиоца са математичком теоремом која сужава скуп делилаца које треба проверити захваљујући чијењеници да се делиоци увек јављају у пару и слично).

Након обраде линеарних алгоритама увести концепт угнежђене петље. На једноставним примерима разјаснити везу између спољашње и унутрашње петље (на пример, сви двоцифрени бројеви се могу исписати тако што спољна петља броји десетице, а унутрашња јединице и корак спољашње петље извршава се тек када се цела унутрашња петља изврши). Приказати класичне примере генерисања дводимензионих објеката (на пример, таблице множења, цртежа геометријских облика креираних од ASCII карактера и слично). Ако се користи графичко окружење добар полигон за вежбање угнежђених петљи је цртање уз помоћ корњача графике.

У делу **Детаљни преглед типова података** извршити преглед специфичних детаља програмског језика који су из методичких разлога изостављени из ранијих тема. Увести све елементарне скаларне типове које изабрани програмски језик подржава (бројевне типове, карактерски тип, набројиви тип, скуповни, интервални тип и слично). Дискутовати скуп вредности које се могу репрезентовати сваким од ових типова и детаљно се осврнути и на њихову интерну репрезентацију (направити корелацију са наставом из предмета Рачунарски системи). Детаљно дискутовати запис константи различитих типова. Описати операторе примењиве на податке ових типова и дискутовати њихова својства (н-арност, асоцијативност, приоритет). Посебну пажњу скренути на појам прекорачења и на ситуације у којима долази до прекорачења. У склопу излагања реалног типа података изложити специфичности записа у покретном зарезу и кроз примере демонстрирати неочекиване резултате и грешке до којих може доћи услед непажљивог руковања са подацима записаним у покретном зарезу (на пример, истакнути проблем поређења једнакости две вредности у покретном зарезу, истакнути губитак тачности до којег долази након примене неких операција и слично).

Увести појам имплицитне и експлицитне конверзије типова и прецизно навести правила конверзије типова у језику који се изучава. Навести примере грешака до којих може доћи ако се не обрати пажња на конверзије које се имплицитно врше. Навести најчешће ситуације у којима је потребно прибећи експлицитној конверзији типова.

У теми **Низови и алгоритми за рад над низовима** увести низ као тип података. Кроз неколико илустративних примера увести потребу за меморисањем свих података истовремено и навести задатке који се не могу решити без низова, коришћењем раније приказаних техника (на пример, исписати све учитане у обратном редоследу, одредити број изнад просечних елемената серије). Приказати и употребу низа као имплементације коначног пресликавања (на пример, низ бројача приликом бројања појављивања свих карактера у тексту).

Описати механизам декларације низова и детаљно продискутовати меморију коју низови заузимају. Фокусирати се само на статички алоциране низове (у језицима у којима су сви низови динамички алоцирани, разматрати само случај низова у којима је димензија константна и позната током компилације). Дискутовати индексни приступ елементима и проблеме до којих долази када се приступи индексу ван дозвољеног опсега низа. Дискутовати и везу низова са потпрограмима (пренос низова у потпрограме и њихово враћање из потпрограма, ако је то подржано језиком).

Приказати неке примере попуњавања и анализе садржаја низова (израчунавање статистика елемената низа, филтрирање низова тј. издвајање елемената који задовољавају дато својство, пресликавања елемената низова и слично). Повући паралеле са алгоритмима обраде серија података који су изучавани у оквиру теме Основни алгоритми цикличке структуре. Појам низа могуће је у свом основном облику увести и раније и користити га за демонстрацију алгоритама цикличке структуре (на пример, методички гледано, много је једноставније разумети сабирање елемената низа него сабирање цифара броја тј. одређивање збира серије бројева је једноставније ако је серија задата кроз елементе низа, него ако је она задата као серија цифара у декадном запису броја).

Приказати алгоритме претраге низа – осврнути се на раније уведено линеарну претрагу, а детаљно увести и продискутовати бинарну претрагу специфичну за сортиране низове.

У наставку се фокусирати на алгоритме модификације који су специфични за низове. Описати уметање елемента на дату позицију и уклањање елемента са дате позиције (уз обавезно очување редоследа осталих елемената, али и без тога). Описати и приказати и уметање и уклањање подниза узастопних елемената.

Приказати и елементарне алгоритме сортирања низова – сортирање селекцијом најмањег елемента (selection sort), сортирање уметањем (insertion sort) и мехурасто сортирање (bubble sort).

Продискутовати специфичности низова карактера (ниски, стрингова). Приказати операторе и библиотеке функције за рад са нискама.

Посебну пажњу посветити и коришћењу низова као репрезентације математичких објеката (вектора, полинома, великих бројева) и приказати алгоритме за рад са тако дефинисаним објектима (на пример, одређивање збира, разлике, производа, количника и остатка при дељењу полинома, израчунавање вредности полинома у датој тачки и слично). Ако се покаже да је ученицима потребно више времена да савладају елементарније теме, ова област се може померити и изложити касније (у другом разреду).

II разред

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Вишедимензиони низови, матрице и основни алгоритми за рад са њима (12 часова)

Кориснички дефинисани типови (6 часова)

Улаз и излаз програма (8 часова)

Анализа алгоритама (6 часова)

Опште технике конструкције алгоритама (38 часова)

Динамичке структуре података и апстрактни типови података (38 часова)

Предвиђена су три двочасовна писмена задатка са једночасовним исправкама (9 часова).

У оквиру теме **Вишедимензиони низови, матрице и основни алгоритми за рад са њима** увести појам матрице (дводимензионог низа), а онда и могућност грађења низова виших димензија. Објаснити алокацију меморије и дискутовати начин смештања елемената у меморији. Фокусирати се само на статички алоциране матрице (у језицима у којима су сви низови динамички алоцирани, разматрати само случај у којима је димензија матрица константна и позната током компилације). Илустровати и однос вишедимензионих низова и потпрограма (пренос у потпрограме и враћање као резултата рада функција).

Описати начине итерације кроз елементе матрице или њеног одређеног дела и нагласити улогу угнежених петљи у томе. Приказати итерацију кроз горњи и доњи троугао матрице, кроз њене произвољне правоугаоне области, кроз околне елементе датог елемента, кроз елементе дате врсте, елементе дате колоне, кроз елементе на главној и на споредној дијагонали и слично. На тако добијене серије бројева применити основне алгоритме за обраду серија (сабирање, тражење минимума, максимума, филтрирање, пресликавање, претрагу и слично). На пример, одредити норму матрице као корен из збира квадрата свих њених елемената, одредити број јединица које се налазе у околини датог поља неке 0-1 матрице (број бомби око поља у игри Minesweeper), проверити да ли је матрица горње троугаона (садржи све нуле испод главне дијагонале) и слично. Приказати статистике по врстама и колонама (на пример, ако су по врстама дате оцене ученика из разних предмета израчунати просечне оцене свих ученика, просечне оцене из свих предмета и пронаћи ученика са највишим и предмет са најнижим просеком).

Приказати и алгоритме трансформације садржаја матрица. Приказати поступак рефлексije елемената у односу на хоризонталну, вертикалну или дијагоналну осу, транспонување, размену две врсте, размену две колоне, сортирање врста или колона по одређеном критеријуму (на пример, сортирање оцена ученика по просеку).

На крају приказати и улогу матрица као математичких објеката. Приказати извођење операција над матрицама (сабирање, одузимање и множење матрица). Истакнути улогу матрица у запису и решавању система линеарних једначина и приказати имплементацију поступка решавања система (на пример, Гаусовог) датих матрицом и низом коефицијената. Приказати неки поступак израчунавања детерминанте матрице (прво за матрице димензија 2 и 3, а затим и општи случај). Заинтересованим ученицима приказати и улогу матрица као линеарних пресликавања вектора и улогу у рачунарској графици (на пример, матрице ротације у равни и у простору, матрице перспективних и пројективних пресликавања над хомогеним координатама).

У склопу теме **Кориснички дефинисани типови** приказати начине дефинисања нових типова на начине које подржава одабрани програмски језик. Приказати рад са набројивим типовима и илустровати њихову употребу (на пример, представљање успеха ученика набројивим типом уз функцију за одређивање успеха на основу низа оцена и функцију конверзије успеха у ниску ради исписа). Ако језик то подржава, приказати дефинисање и употребу интервалних и скуповних типова. Ако наставник процени за сходно, набројиви, интервални и скаларни типови се могу увести и у првом разреду, у склопу разматрања основних скалараних типова.

Посебну пажњу посветити структурним типовима у језицима који их подржавају (структурама у језицима C, C++ или C#, слоговима у језику Pascal). У језицима који не подржавају директно структуре, приказати најближи начин да се оне замене (на пример, коришћење класа са свим јавним подацима, без метода у језику Java и слично). Истаћи разлику између дефиниције структурног типа и дефиниције објекта тог типа. Објаснити меморијску реализацију структурних типова и њихову алокацију (нарочито ако се уместо структура морају користити динамички алоцирани објекти). Дискутовати однос структура и потпрограма (пренос у потпрограм и враћање као резултат рада потпрограма). Приказати примере програма у којима се структуре користе (структуре за репрезентацију разломка, комплексног броја, тачке у равни или простору, података о ученику и слично).

У оквиру теме **Улаз и излаз програма** описати рад са датотекама. Описати начин приступа датотеци из програма (отварање датотеке, затварање датотеке), учитавање података (појединачних карактера, линија, бројева) из датотека и упис података у датотеке. Скренути пажњу на разлику између текстуалног и бинарног приступа датотекама. Текстуални приступ датотекама приказати детаљно, а бинарни само информативно. Стандардни улаз, стандардни излаз и стандардни излаз за грешке приказати као посебне врсте датотека. У склопу ове теме детаљно дискутовати и форматирање текста (испис у пољу фиксне ширине, испис са одређеном прецизношћу, бројем децимала и слично).

Приказати и прихватање аргумената командне линије програма и нагласити најчешће сценарије њихове употребе (на пример, задавање опција програму). Приликом обраде ове теме нагласак ставити на конзолне апликације са КЛИ, али ученицима нагласити и да аргументи командне линије нису специфичне само за њих, већ се аргументи командне линије могу користити и у апликацијама са ГКИ. Приказати и како се аргументи командне линије програма могу подесити и користити у ако се програм не покреће директно из командне линије већ у оквиру интегрисаног развојног окружења.

У оквиру теме **Анализа алгоритама** приказати елементарне начине анализе коректности и анализе сложености алгоритама.

У склопу анализе коректности приказати значај темељног и што исцрпнијег тестирања програма и приказати различите технике тестирања. У склопу обраде ове теме могуће је увести и неки радни оквир за тестирање (нпр. JUnit у програмском језику Java).

Поред тестирања као технике динамичке верификације програма, ученике упознати и са методама статичке верификације и формалне анализе коректности програма. Дефинисати појам заустављања програма, парцијалне коректности и тоталне коректности (као комбинације претходна два појма). Увести појам предуслова и постуслова програма и његових разних делова (пре свега потпрограма). На пример, скренути пажњу да иако математичке функције које раде над природним бројевима (на пример, функција за испитивање да ли је број прост) примају параметре неког целобројног типа, предуслов њихове коректности је да је да прослеђени аргументи буду природни бројеви, некад строго већи од нуле, а да функције са рад са угловима у степенима, минутима и секундама као предуслов захтевају, а као постуслов осигуравају да се број минута и секунди креће између 0 и 59.

Неформално увести и појам инваријанте, као логичког услова који важи током извршавања одређеног дела програма и осигурава његову коректност. Илустровати овај појам кроз већи број примера (на пример, инваријанта петље која врши степеновање броја x узастопним множењем n пута је то да се након извршених

k множења у променљивој која чува резултат налази k -ти степен броја x , док је вредност k мања или једнака вредности n , док је инваријанта спољне петље алгорита сортирања селекцијом то да се након њених k извршавања на првих k места у низу налази најмањих k елемената низа и да су сви они међусобно сортирани, као и да се мултискуп елемената у низу није променио). Приказати како се доказивањем да је инваријанта испуњена пре уласка у петљу и да се одржава извршавањем тела петље, на основу принципа математичке индукције осигурава да ће инваријанта бити испуњена и након изласка из петље, што заједно са условом изласка из петље гарантује коректност.

Увести појам просторне и временске сложености програма. Објаснити неопходност одређивања перформанси програма у реалним условима. Појам временске сложености прво илустровати пуштањем примера различитих имплементација истог програма и мерењем времена потребних да се програм изврши (на пример, израчунавање последње три цифре збира првих милијарду природних бројева коришћењем петље или применом Гаусове формуле, претрага низа линеарном и бинарном претрагом и слично). Објаснити како се очекивано време извршавања може проценити на основу броја операција које програм треба да изврши за улаз одређене димензије. На једноставнијим примерима приказати како се тај број операција може грубо проценити. Увести сложеност најгорег случаја и просечну сложеност алгорита.

Приказати ученицима табелу која под претпоставком да се једна операција извршава за једну наносекунду приказује време потребно да се изврши програм чији је улаз различите димензије n (на пример, за $n=10^2$, па до 10^9), ако број операција од димензије улаза зависи у виду функције $\log_2 n$, n , $n \cdot \log_2 n$, n^2 , n^3 , 2^n и $n!$. Да би се стекао бољи осећај комбинаторне експлозије дугачка времена изразити у минутима, сатима, данима, месецима, годинама и слично. Коришћењем ове табеле објаснити како укупно време израчунавања програма практично зависи само од доминантног сабирка у функцији која описује зависност броја операција од димензије проблема (на пример, у функцији $n^2 + 3n + 5$, за иоле веће димензије n практично све време одлази на n^2 операција). Илустровати и како облик функције неупоредиво више утиче на време извршавања за велике улазе, него константни фактор који се јавља уз водећи сабирка (упоредити, на пример, n^2 и $10n$). Овакву анализу употребити као основу за (неформално) увођење O нотације. Навести примере алгоритама које су ученици раније сретали, а који имају сложеност $O(\log n)$, $O(n)$, $O(n^2)$, $O(2^n)$. Дискутовати шта се дешава са временом извршавања двоструким увећањем димензије улаза.

Пошто ова тема представља основу за конструкцију ефикасних алгоритама, њено детаљније разрађивање је предвиђено и током обраде наредних тема (на пример, алгоритме сложености $O(n \log n)$ ученицима има смисла приказати током обраде технике подели и владај.

Обраду теме **Опште технике конструкције алгоритама** започети прегледом рекурзије тј. индукције. Као мотивацију за увођење технике рекурзије, на пример, приказати веома елегантно рекурзивно решење проблема Хановских кула (пре тога осигурати да су сви ученици имали довољно времена да разумеју правила игре и да покушају самостално да конструишу неки алгоритама решавања).

Систематски преглед **рекурзије** започети приказом примитивно рекурзивних функција над природним бројевима (функције у којима постоји правило изласка из рекурзије када је вредност параметра 0 и у којима постоји правило рекурзивног корака у којем се вредност функције за параметар који је следбеник неког броја израчунава на основу вредности рекурзивног позива у којем је параметар тај број). Приказати имплементацију степеновања свођењем на множење, множење свођењем на сабирање, сабирање свођењем на следбеника, израчунавање факторијела и слично. Приказати и примитивно рекурзивне функције за обраду низова (функције које као базу користе празан или једночлан низ, а које у склопу рекурзивног корака низ разлажу на префикс низа испред последњег елемента и последњи елемент, или, дуално, на први елемент и суфикс низа иза њега). Приказати функције за израчунавање збира елемената низа, максимума/минимума, линеарну претрагу низа, филтрирање,

пресликавање и слично. Такође, могуће је приказати и рекурзивне имплементације елементарних алгоритама сортирања који су раније ученицима приказани у итеративном облику.

Након примитивно рекурзивних функција показати и сложене облике рекурзије. Приказати функције које за вредност параметра n користе рекурзивне позиве за произвољне вредности мање од n , укључујући и могућност постојања већег броја рекурзивних позива. Приказати имплементацију ефикасног алгоритама степеновања, анализом парности експонента и свођењем вредности n на вредност $n/2$ (уместо на $n-1$) у случају парног експонента. Приказати рекурзивну дефиницију Еуклидовога алгоритама. На домену рада са нивозима, приказати рекурзивну имплементацију обртања низа, провере да ли је низ палиндром, бинарне претраге низа и слично.

Погодни примери за илустрацију рекурзије у случају да се програмирају апликације са ГКИ јесу фрактали (нпр. тепих Сијерпинског, L-системи и слично).

Приказати примену рекурзије на израчунавање елемената рекурентно задатих низова укључујући и Фибоначијев низ и рекурзију Фибоначијевог типа и дискутовати проблеме који настају због преклапајућих потпроблема, односно вршења истих рекурзивних позива више пута. Навестити да се ти проблеми решавају динамичким програмирањем (мемоизацијом, односно динамичким програмирањем навише) и да ће томе бити посвећена посебна наставна тема.

Ученицима приказати механизам реализације рекурзије коришћењем системског стека (при чему је важно осигурати да ученици разумеју да се реализација позива функција врши преко стека и ван контекста рекурзије). Пажљиво анализирати предности и мане рекурзивних у односу на итеративна решења. Дискутовати временску и просторну сложеност различитих рекурентних решења и скренути пажњу ученицима на проблем веће меморијске сложености до које може доћи због интензивног нагомиланања стек оквира и до проблема прерачења стека. Са друге стране истакнути језгровитост и разумљивост рекурзивних дефиниција.

Дискутовати могућност систематског уклањања рекурзије. Скренути пажњу на појам репне рекурзије и објаснити начине њене елиминације, тј. превођења репно-рекурзивних у итеративне дефиниције.

Током разраде ове теме од ученика захтевати и да пишу своје рекурзивне функције, али и да корак-по-корак приказују како се извршавају задате рекурзивне дефиниције.

Природна примена рекурзије лежи у алгоритама **систематске енумерације и претраге са повратком** (енгл. backtracking). Приказати примену систематске енумерације на проблем генерисања свих подскупова датог скупа, свих варијација са понављањем, исписа истинитосне таблице дате исказне формуле и слично. Претрагу са повратком, на пример, илустровати на проблему 8 дама, решавању магичних квадрата, латинских квадрата и судоку загонетке, проналаску обиласка шаховске табле скакачем и слично.

Дефинисати алгоритме засноване на **грубој сили** (енгл. brute force) као алгоритме који у претрази за решењем проверавају све могућности и нагласити како су систематска исцрпна енумерација и претрага са повратком типични примери алгоритама грубе силе. Навести и једноставније примере алгоритама грубе силе са којима су се ученици и раније сретали (на пример, линеарна претрага низа, наивна претрага подниске и слично). Објаснити у којим ситуацијама има смисла користити приступ заснован на грубој сили: када је једноставност имплементације важнија од брзине, када доказ коректности алгоритама треба да буде што једноставнији (на пример, када се рачунар користи за доказ математичке теореме или када алгоритама служи само за проверу тачности или квалитета другог алгоритама или хеуристике) и слично.

Дефинисати класу **грамзивих алгоритама** као алгоритама за решавање оптимizacionих проблема који у сваком кораку бирају локално оптимално решење у циљу да ће се на тај начин на крају доћи и до глобално оптималног решења. Кроз примере скренути пажњу на мноштво ситуација у којима грамзива стратегија не доводи до оптималног решења проблема (на чак у неким случајевима може да врати и најлошије решење). Ипак, истакнути да се чак и

упркос проналажењу субоптималних решења, у неким ситуацијама грамзивии алгоритми могу користити као хеуристике које се једноставно могу имплементирати и које дају релативно квалитетна решења. Објаснити како примена грамзивих алгоритама захтева да проблем има својство оптималне подструктуре (оптимално решење проблема се конструише од оптималних решења потпроблема). Навести примере проблема који имају то својство (на пример, ако најкраћи пут од Ниша до Новог Сада пролази кроз Крагујевац и Београд, онда најкраћи пут од Крагујевца до Новог Сада мора такође проћи кроз Београд) и проблема које то својство немају (ако најјефтинија аутобуска веза од Ниша до Суботице подразумева само преседање у Београду, док најјефтинија веза од Београда до Суботице може да укључи преседање у Новом Саду, јер цена карте са преседањем не мора да буде увек збир цена појединачних возњи). Истакнути и како својство оптималне подструктуре не гарантује могућност примене грамзивог приступа, већ је у многим случајевима потребно користити исцрпнију претрагу (обично засновану на динамичком програмирању). Навести примере грамзивих решења који доводе до оптимума. На пример, описати проблем проналажења најширег подскупа непреклапајућих активности за дати скуп активности којима се зна почетак и крај, описати проблем упаривања две шаховске екипе тако да екипа домаћина може да оствари највећи број победа, описати Хафманово кодирање и конструкцију Хафмановог стабла, варијанту проблема ранца у којој могу да се узимају делови производа, проналажење највеће површине три круга уписана у троугла и слично. Теми грамзивих алгоритама се враћати и касније (на пример, код изучавања Дајкстриног, Примовог и Краскаловог алгоритама истакнути њихову грамзиву природу).

Приликом излагања технике **динамичког програмирања** скренути пажњу на проблем преклапајућих потпроблема. Класичан пример за то је рекурзивна дефиниција Фибоначијевог функције. Дефинисати динамичко програмирање као технику у којој се користи помоћна структура података (најчешће низ или матрица) за меморисање вредности решења одређених потпроблема. Увести технику мемоизације као динамичко програмирање одозго наниже, и технику класичног динамичког програмирања одоздо навише. Указати значај динамичког програмирања у решавању оптимizacionих проблема и истакнути важност постојања оптималне подструктуре проблема у том случају. Повући паралелу са грамзивим приступом. Поред Фибоначијевог низа, приказати, на пример, и решења проблема проналажења најмањег броја новчаница потребних да се плати дати износ, проналажења максималног збира поља у матрици који се може постићи обиласком из горњег левог до доњег десног угла у којем се у сваком кораку прелази на суседно поље доле или десно од тренутног, проналажења најдужега заједничког подниса два низа, проналажења подниса узастопних елемената чији је збир максималан, проналажења најдужега неоппадајућег подниса датог низа, израчунавања Левенштајновог (edit) растојања две ниске, проблем ранца, проналажења заграђивања (асоцијативности) при множењу серије матрица тако да је укупан број операција у израчунавању производа минималан и слично. Кроз неколико примера приказати како се чувањем додатних информација омогућава реконструкција решења након проналажења вредности оптималног решења. Дискутовати меморијску и просторну сложеност свих конструисаних алгоритама.

Ученицима показати технику **подели па владај** и истакнути предност ефикасног свођења проблема на решавање два потпроблема двоструко мање димензије неко једног потпроблема димензије за један мање. Исказати мастер-теорему и ученицима дати њено неформално образложење (потпуно формалан доказ може бити показан најзаинтересованијим ученицима). Приказати најзначајније примере алгоритама заснованих на принципу подели па владај, који су значајни како за приказ ове технике, тако и сами за себе. Приказати у овом светлу алгоритама бинарне претраге низа (чак и ако је раније уведен сам за себе) и извести да ње његова временска сложеност логаритамска. Приказати алгоритама сортирања обједињавањем (MergeSort) и алгоритама брзог сортирања (QuickSort). Извести њихову сложеност најгорег случаја и просечну сложеност.

Инсистирати на томе да осим времена потребног за решавање мањих потпроблема, време потребно за припрему рекурзивних позива односно за обраду резултата након рекурзивног позива сме да буде највише линеарно (да би се остварило укупно време $O(n \log n)$). У том светлу приказати неколико различитих алгоритама партиционисања приликом припреме за брзо сортирање. Дискутовати и начине избора пивотирајућег елемента и последице те одлуке на сложеност најгорег случаја брзог сортирања. Приказати модификације идеја брзог сортирања на решавање других проблема (на пример, проналажење медијане низа, проналажење k -тог по величини елемента у низу). Приказати и примене технике поделе па владај и на решавање проблема из других домена. Погодни проблеми могу бити проналажење најближег пара тачака међу датим тачкама у равни, одређивање контуре коју покрива скуп правоугаоника, множење два велика броја Каратсубиним алгоритмом, Штра-сенов алгоритам за множење матрица, брза Фуријеова трансформација (FFT) и примена на множење полинома, паралелно бројање битова и слично.

Увод у тему **Динамичке структуре и апстрактни типови података** започети детаљнијим прегледом организације меморије, адреса и показивача. Увести и објаснити сегменте меморије (стек, хип, сегмент података, сегмент кода). У језицима у којима је то примерено (на пример, C и C++) увести везу између показивача и низова и показивачку аритметику. У језицима у којима показивачи нису експлицитно видљиви (на пример, Java, C#) објаснити да су референтни типови података у програму представљени показивачима (на пример, `int[] arr`; у језику C# или `Random rnd`; у језику Java су заправо само адресе објеката који се алоцирају динамички и који се налазе на хипу). Дискутовати веома детаљно пренос података у функцију (пренос преко показивача, пренос преко референце, пренос референтних типова и слично) и враћање података из функције. Све време дијаграмима илустровати садржај меморије. Детаљно објаснити појам динамичке алокације меморије и разлику између динамички и статички алоцираних података. Чак и када језик поседује сакупљач отпадака скренути пажњу на опасност неконтролисаних алокације меморије (најчешћи пример је непотребна стална алокација једног те истог низа или објекта унутар петље) и указати на то да постоје језици у којима је програмер дужан да експлицитно ослободи сву меморију коју је динамички алоцирао.

Преглед динамичких структура података започети **динамичким низовима и матрицама** чија се величина по потреби мења коришћењем реалокације (чиме се остварује функционалност колекција какве су `vector` у језику C++ или `ArrayList` у језику Java).

Увести затим појам **уланчане тј. повезане листе**. Дискутовати предности и мане овако организоване колекције у односу на статичке и динамичке низове. Дискутовати распоред елемената у меморији и поред стандардног начина где се уз податке чувају меморијске адресе (показивачи) поменути и могућност имплементације помоћу низа у којем се након сваког податка чува индекс наредног података у склопу тог низа. Дискутовати варијанте једнострукто повезане листе, двоструко повезане листе и кружне листе. Увести операције уметања елемента на почетак листе, уметања елемента на крај листе, итерацију кроз елементе листе, брисање елемента са почетка или краја листе, и слично. Разматрати и рекурзивне и итеративне имплементације. Разматрати варијанте у којима је познат само почетак и у којима су познати и почетак и крај листе. Упоредити сложеност свих ових операција и упоредити их са сложеносћу истих операција код динамичких низова. Функције за рад са листама реализовати и итеративно и рекурзивно. Скренути ученицима пажњу да је у функцијама које могу да промене почетак листе, тај почетак неопходно или пренети по референци или обезбедити да функција врати нови почетак, а да се након позива вредност почетка ажурира. Дискутовати сотиране листе и уметање елемента на своје место у сортираној листи. Упоредити ово са раније изложеним алгоритмом сортирања уметањем. Увести и обједињавање две сортиране листе у трећу и упоредити са раније изложеним алгоритмом сортирања обједињавањем.

Након тога прећи на разматрање **бинарних дрвета**. Дефинисати основне појмове (чвор, лист, корен итд.). Описати разне

употребе дрволикх (хијерархијских) структура у рачунарству. Дискутовати дрволику репрезентацију израза (синтакстичка дрвета), дрволику репрезентацију директоријума (фасцикли) у оквиру система датотека и слично. Дефинисати разне начине обиласка бинарног дрвета, по ширини и по дубини (префиксно тј. `preorder`, инфиксно тј. `inorder` и постфиксно тј. `postorder`). Илустровати обилазак и на примеру дрвета израза и указати на значај префиксног и постфиксног записа израза.

Приказати репрезентацију бинарног дрвета у програму (као скупа чворова који садрже податке и два показивача). Поменути могуће уопштење на n -арна дрвета (репрезентација са низом показивача у чвору, бинаризација помоћу показивача на брата и сина). Приказати поступак формирања дрвета (алокација и повезивање чворова). Скренути пажњу на неопходност преноса корена по референци или експлицитног враћања показивача на нови корен у функцијама које могу да промене дрво тако да му се мења корен. Имплементирати функције обиласка (рекурзивно и итеративно). Имплементирати функције које одређују статистике над подацима забележеним у чворовима дрвета (на пример, број чворова дрвета, број листова дрвета, збир свих елемената дрвета, највећи и најмањи елемент у дрвету и слично).

– Дефинисати **уређено бинарно дрво** (бинарно дрво претраге, BST) и објаснити његове карактеристике. Реализовати операције тражења елемента са датом вредношћу и повући паралеле са алгоритмом бинарне претраге. Реализовати операције додавања елемента у дрво и брисања елемента из дрвета. Приказати итеративне и рекурзивне имплементације различитих операција. Дискутовати сложеност најгорег случаја за све операције и истакнути да сложеност зависи од тога да ли је дрво балансирано. Поменути да постоје самобалансирајућа дрвета (Адисон-Вељски-Ландис AVL, црвено-црна дрвета, RBT), али да њихово изучавање превазилази опсег овог предмета (заинтересоване ученике је могуће упознати са овим структурама у оквиру додатне наставе). Дискутовати могућност имплементације сортирања коришћењем бинарног дрвета (TreeSort) и упоредити га са алгоритмом сортирања уметањем (InsertionSort). Нагласити како се иста идеја алгорита (у овом случају је то уметање елемената у претходно сортирану колекцију) може имплементирати над различитим структурама (у овом случају, низа, листе и дрвета) података и како се у зависности од одабране структуре добија различита сложеност.

Дефинисати **хип** (енгл. *Heap*) тј. гомилу. Приказати имплементацију хипа помоћу низа. Приказати операције додавања елемента у хип, формирања хипа (дискутовати варијанте формирања хипа одоздо навише и одозго наниже), проналажења најмањег тј. највећег елемента као елемента на врху (у зависности од тога како је хип уређен) и његово уклањање из хипа. Дискутовати сложеност свих ових операција. Илустровати алгоритам сортирања помоћу хипа (HeapSort) и дискутовати његову везу са сортирањем избором (*SelectionSort*).

Дефинисати **хеширање** и **хеш-табеле**. Дискутовати особине које хеш-функција треба да задовољи и навести примере неких хеш функција. Дефинисати колизије и објаснити различите могућности разрешавања колизија (уланчавање и отворено адресирање). Приказати имплементацију разрешавања колизија коришћењем уланчавања помоћу једнострукто уланчаних листа.

Дефинисати појам **апстрактног типа података** као математичког модела података где се у први план ставља скуп вредности које тип подржава, операције које се над подацима тог типа могу извести и њихово понашање. Скренути пажњу на то да се апстрактни тип података често може имплементирати коришћењем различитих конкретних структура података.

Дефинисати апстрактни тип података **стек** и операције које га карактеришу. Објаснити шта значи да стек функционише по LIFO принципу. Низом примера илустровати употребу стека (на пример, провера упарености заграда, системски стек, елиминација рекурзије). Приказати различите начине имплементације стека (помоћу статичког низа, динамичког низа, уланчане листе). Дискутовати сложеност операција за различите имплементације. Ако програмски језик подржава библиотечку имплементацију стека, приказати њену употребу.

Дефинисати тип података **ред** и операције које га карактеришу (разматрати ред са једним и ред са два краја). Објаснити шта значи да ред функционише по FIFO принципу. Низом примера илустровати употребу реда. Приказати различите начине имплементације реда. Ако програмски језик подржава библиотечку имплементацију реда, приказати њену употребу.

Дискутовати сложеност операција за различите имплементације. Дефинисати апстрактни тип **ред са приоритетом**. Низом примера илустровати његову употребу. Приказати различите начине имплементације, дискутовати сложеност операција и приказати предности имплементације помоћу хипа. Ако програмски језик подржава библиотечку имплементацију реда са приоритетом, приказати њену употребу.

Дефинисати апстрактни тип података **скуп** и приказати основне операције (уметање елемента, провера да ли елемент припада скупу) и изведене операције над скуповима (унија, пресек, разлика, подскуп итд.). Приказати различите начине имплементације, дискутовати сложеност и приказати предност имплементације помоћу хеш-табеле или уређеног бинарног дрвета.

Дефинисати апстрактни тип података **речник** (назива се још и мапа, магацин, каталог, асоцијативни низ) и основне операције које он подржава (додела вредности датом кључу и претраживање вредности на основу датог кључа). Приказати различите начине имплементације, дискутовати сложеност и приказати предност имплементације помоћу хеш-табеле или уређеног бинарног дрвета. Дискутовати да се мултискупови могу једноставно имплементирати помоћу речника. Низом примера илустровати употребу овог типа података (на пример, пребројати појављивања свих речи које се појављују у текстуалној датотеци). Ако програмски језик подржава библиотечку имплементацију речника, приказати њену употребу.

Дефинисати апстрактни тип података за представљање колекције **дисјунктних скупова** (енгл. disjoint-set data structure). Дефинисати основне операције: проналажење уније два скупа (енгл. union) и проналажење скупа коме припада дати елемент (енгл. find). Дискутовати разне могућности имплементације, а фокусирају се на имплементацију са обратном усмереним дрветима. Дискутовати сложеност операција и могућност скраћивања путања приликом операције *find*.

Иако план предвиђа излагање конкретних структура података (листа, дрвета и сл.) пре апстрактних типова података (стека, речника и сл.) могуће је излагање организовати и обратно тј. прво показати како се користе апстрактни типови података (нарочито у језицима који поседују њихове библиотечке имплементације), а тек након тога приказати детаље њихове имплементације.

III разред

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Графови и алгоритми за рад са графовима (20 часова)

Алгоритми текста (10 часова)

Геометријски алгоритми (10 часова)

Алгоритми теорије бројева (10 часова)

Алгоритми над битовима (4 часа)

Преглед одабраних структура података и алгоритама (16 часова)

Предвиђена су три двочасовна писмена задатка са једночасовним исправкама (9 часова).

Преглед теме **Графови и алгоритми за рад са графовима** започети дефиницијом графа (усмерен, неусмерен) и основних појмова у вези са графовима (чворови, гране, улазне гране, излазне гране, степен чвора, путеви, циклуси и слично). Описати различите начине репрезентација графова у програму. Инсистирати на репрезентацији помоћу матрице повезаности и помоћу листа повезаности за сваки чвор. Дискутовати сложеност различитих операција над овим репрезентацијама (провера да ли постоји грана између два чвора, проналазак свих суседа датог чвора). Дефинисати појам тежинског графа и приказати његове могуће репрезентације.

Приказати имплементацију алгоритама **обиласка тј. претраге графа**. Имплементирати претрагу графа у дубину (DFS) и претрагу графа у ширину (BFS). Упоредити са раније описаном претрагом дрвета. Приказати уобичајену рекурзивну имплементацију графа у дубину и повезати са ранијом причом о алгоритмима систематске претраге и претраге са повратком. Приказати итеративне имплементације обе врсте претраге уз коришћење помоћних структура података (стека у случају претраге у дубину и реда у случају претраге у ширину). Дефинисати дрво претраге у дубину и његове могуће примене. Приказати примене обиласка графова (испис свих чворова, одређивање компонената повезаности и слично). При одабиру примера графове употребити тако да илуструју проблеме из реалне праксе (чворови могу, на пример, представљати градове, а гране путеве, чворови могу представљати неке послове, а гране зависности између њих и слично).

Описати алгоритме **тополошког сортирања** (Канов алгоритам, алгоритам заснован на обиласку у дубину), приказати њихову имплементацију и дискутовати сложеност. Приказати могуће реалне примене овог алгоритма (на пример, провера да ли је граф цикличан, одређивање неког редоследа којим се могу положити испити ако су дате условне зависности међу њима и слично).

Описати **Дајкстрин алгоритам** проналажења најкраћег пута од датог чвора до осталих чворова у тежинском графу. Дискутовати да проналажење најкраћег пута од датог полазишта до неког конкретного одређеног места у најгорем случају захтева и одређивањем најкраћих путева од полазишта до свих осталих чворова у графу. Навести како је Дајкстрин алгоритам грамзив и доказати оправданост узимања најближег чвора у сваком појединачном кораку. Приказати имплементацију у којој се текуће раздаљине чувају у низу и имплементацију у којој се чувају у реду са приоритетом. Дискутовати сложеност обе имплементације и истакнути предност друге имплементације. Упоредити Дајкстрин алгоритам са алгоритмом претраге у ширину. Поред одређивања дужине најкраћег пута, описати и како се може реконструисати сам пут.

Описати **Флојд-Варшалов алгоритам** за проналажење најкраћих путева између свих чворова. Описати како се током сваке наредне итерације алгоритма повећава скуп чворова преко којих могу да прођу путеви између чворова. Алгоритам приказати као алгоритам динамичког програмирања. Дискутовати сложеност. Упоредити са (мање ефикасним, али можда очигледнијим) алгоритмима у којима се индуктивна конструкција врши по скупу чворова графа, по скупу грана и по допуштеној дужини путева. Поред одређивање дужина путева, приказати како је могуће реконструисати саме путеве. Приказати примене овог алгоритма (на пример, проналажење времена најбрже комуникације између два чвора у мрежи, проналажење транзитивног затворења релације).

Дефинисати разаципићу дрво датог графа и описати проблем проналажења **минималног разаципићу дрвета**. Приказати **Примов** и **Краскалов** алгоритам. Нагласити да су оба алгоритма грамзива и доказати оправданост бирања локално најбољег решења у сваком кораку. Приказати разне могућности за имплементацију Примовог алгоритма (коришћењем низа или коришћењем реда са приоритетом) и дискутовати сложеност. Упоредити имплементацију Примовог и Дајкстриног алгоритма. Приказати имплементацију Краскаловог алгоритма у којој се користи структура података за представљање дисјунктних скупова и дискутовати сложеност. Приказати примере у којима се примењују ови алгоритми (нпр. разаципићање што мање каблова тако да сви чворови постану повезани, генерисање лавиринта и слично).

Ако наставник осети потребу и ако време то дозволи, може ученицима додатно приказати и неке друге графовске алгоритме.

У склопу обраде теме **Алгоритми текста** на почетку анализирати алгоритме за **претрагу дате подниске** унутар дате ниске. Подсетити ученике на стандардни алгоритам који претрагу врши грубом силом и дискутовати његову сложеност. Дискутовати мање тог алгоритма и истакнути потенцијална места за оптимизацију. Приказати неки од алгоритама претраге линеарне сложености (Кнут-Морис-Пратов алгоритам или Бојер-Муров алгоритам). Увести појам **регуларних израза** и приказати како се регуларни

изрази користе за опис језика (на пример, описати адресе електронске поште или реалне константе). Ако програмски језик то подржава, приказати како се у њему врши претрага на основу регуларних израза. Увести појам **контекстно-слободних граматика** и објаснити њихову употребу за опис синтаксе програмских језика (у облику, на пример, проширене Бекусове нотације). Приказати технику **рекурзивног спуста** и њену примену на рашчлањавање аритметичких израза или исказних формула.

Ако наставник осети потребу и ако време то дозволи, може ученицима додатно приказати и неке друге алгоритме за рад са текстом.

У склопу обраде теме **Геометријски алгоритми** подсетити ученике на основне појмове **аналитичке геометрије** (векторе, координате, скаларни, векторски и мешовити производ). Показати примене скаларног производа на испитивање нормалности, векторског производа на испитивање колинеарности и мешовитог производа на испитивање копланарности тачака у простору. Увести појам оријентације тачака у равни и приказати примене векторског производа на испитивање оријентације тачака. Увести релацију „са исте стране праве” и приказати како се она може проверити (на пример, применом векторског производа).

Подсетити ученике на начин **израчунавање површине** троугла задатих координатама темена (Хероновим обрасцом, векторским производом). Демонстрирати како се површина конвексног полигона може добити једноставним разлагањем на троуглове (чија је страница једна фиксирана ивица полигона). Увести **формулу пертлања** (енгл. shoelace formula) за израчунавање површине произвољног простог полигона (полигона без самопресека). Приказати алгоритам провере да ли тачка припада унутрашњости полигона.

Приказати ученицима алгоритам за **крепање простог полинома** на основу датог скупа тачака.

Приказати ученицима алгоритме за **израчунавање конвексног омотача** полигона. Поменути алгоритам грубе силе (за сваку дуж проверити да ли је део конвексног омотача) и дискутовати његову сложеност. Приказати алгоритам умотавања поклона (**Царвисов марш**) и **Грахамов алгоритам**. Дискутовати њихову сложеност. Поменути могућност изградње алгоритма техником подела па владај.

Ако наставник осети потребу и ако време то дозволи, може ученицима додатно приказати и неке друге геометријске алгоритме.

У оквиру теме **Алгоритми теорије бројева** са ученицима обрадити **Еуклидов алгоритам** за одређивање НЗД (ако је овај алгоритам раније рађен као пример, подсетити ученике на њега). Приказати како се израчунавање НЗС своди на израчунавање НЗД. Увести проширен Еуклидов алгоритам и приказати његове примене: решавање диофантских једначина, одређивање модуларног инверза и дељење по модулу.

Обрадити тему **деливости**, доказати да се делиоци броја увек јављају у паровима и употребити је у алгоритмима (одређивање делилаца, испитивање да ли је број прост). Описати **Ератостеново сито** за проналажење свих простих бројева мањих од датог броја. Приказати алгоритам за **растављање броја на просте чиниоце** и његове примене (одређивање броја и збира делилаца на основу мултипликативности тих функција). Увести Ојлерову функцију (број бројева узајамно простих са датим бројем), доказати њену мултипликативност и показати како се она може ефикасно израчунати када је број растављен на просте чиниоце. Приказати примене у криптографији.

Ако наставник осети потребу и ако време то дозволи, може ученицима додатно приказати и неке друге алгоритме теорије бројева.

У оквиру теме преглед **Алгоритама над битовима** покрити приказ бинарног садржаја променљивих, поступке издвајања одређеног скупа битова из броја, упис нула битова на одређене позиције у броју, упис један битова на одређене позиције у броју, инвертовање битова на одређеним позицијама у броју, уклањање последњег један бита у броју и слично. Приказати и одређивање степена двојке шифтовањем и ефикасно испитивање да ли је број степен двојке. Приказати разне алгоритме бројања један битова у

запису броја, укључујући и паралелно бројање битова (по принципу подела па владај) и дискутовати њихову сложеност. Приказати рефлексију (обртање редоследа) битова.

Тема **Преглед одабраних структура података и алгоритама** предвиђена је за утврђивање и проширивање градива. Реализацију ове теме или неког њеног дела могуће је остварити и на почетку године, ако се покаже да је потребно више времена за обнављање неких тема предвиђеним планом и програмом другог разреда. Такође, ако се покаже потреба за тим, број часова за детаљно утврђивање основнијих алгоритама је могуће проширити па и по цену да се неки напреднији алгоритми (на пример, неки графовски алгоритми, неки алгоритми теорије бројева или неки геометријски алгоритми) не обраде или се обраде само у склопу додатне наставе.

РАЧУНАРСКИ СИСТЕМИ

Циљ и задаци

Циљ наставног предмета Рачунарски системи је стицање основних знања о архитектури и организацији рачунара и карактеристикама рачунарских система.

Задаци наставе предмета Рачунарски системи су да ученици:

- упознају компоненте рачунарског система, њихове функције и начин њиховог повезивања;
- буду оспособљени да ураде једноставне аритметичке операције са целим бројевима у бинарном и хексадецималном бројном систему и конверзије између система;
- познају основне хардверске компоненте рачунара, принципе њиховог функционисања и улогу у ширем рачунарском систему;
- разумеју улогу машинског језика и разлику између асемблерског и машинског језика.

I разред

(2 часа недељно, 72 часа годишње)

1. Увод у рачунарске системе

- Појам и врсте рачунарских система.
- Хардвер и софтвер.
- Улога рачунарских система у савременом друштву.
- Историјат развоја рачунарских система.

2. Дигитални запис података

- Однос дигиталног и аналогног записа.
- Бројевни системи: декадни, бинарни, хексадекадни, октални.
- Запис неозначених бројева и операције над њима.
- Запис означених бројева и операције над њима.
- Запис разломљених бројева и операције над њима.
- Запис текста.
- Запис слике, звука и видеа.

3. Логичке основе обраде података

- Булова алгебра.
- Комбинаторна логичка кола.
- Секвенцијална логичка кола.

4. Основи архитектуре и организације рачунара

- Архитектура и организација рачунарског система.
- Процесор.
- Меморијска хијерархија.
- Улазно-излазни уређаји.
- Магистрале.

5. Асемблерско програмирање

- Асемблерски и машински језици.
- Скуп инструкција.
- Начини адресирања.
- Машински стек.
- Примери програма.

НАЧИН ОСТВАРИВАЊА ПРОГРАМА

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Увод у рачунарске системе (12)

Дигитални запис података (16)

Логичке основе обраде података (14)

Основи архитектуре и организације рачунара (14)

Асемблерско програмирање (14).

У оквиру теме **Увод у рачунарске системе** пожељно је ученицима створити грубу слику о структури и начину функционисања рачунарских система. Излагање може да буде само прегледно, веома неформално и током овог уводног прегледа нема потребе улазити у сувишне детаље. Дефинисати појам технолошке конвергенције и истакнути да се данас користе различити облици рачунарских система (стони и преносни рачунари, таблети, паметни телефони, уграђени (embedded) рачунарске системи и слично), који сви функционишу по веома сличним принципима. Истаћи да су данашњи рачунарске системи по правилу дигитални и укратко и неформално илустровати како се различити подаци (текст, слике, звук) записују дигитално, помоћу бројева. Дефинисати основне компоненте хардвера рачунара (процесор, главну меморију, магистрале и улазно-излазне уређаје) и илустровати како они учествују у раду рачунарских система (на примеру извршавања неколико асемблерских инструкција). Осврнути се на софтвер рачунара и његову класификацију, са посебним нагласком на системски софтвер, оперативне системе и њихов однос са хардвером рачунара. У оквиру ове теме обрадити и улогу рачунарских система у савременом друштву. Описати ситуације у којима рачунари и технологија олакшавају личне и професионалне животне ситуације. Осврнути се и на утицај прекомерне употребе рачунара на здравље људи и утицај рачунарског отпада на животну средину. Ученицима дати хронолошки преглед развоја рачунарских система и програмабилних рачунарских система кроз историју. Не insistирати на техничким карактеристикама појединих система нити на прецизним датумима њиховог изума, већ истаћи значај и нове концепте које су ти уређаји први пут увели. Ипак, insistирати на томе да ученици умеју да истакну значај сваког уређаја и да га сместе у неки временски период и историјски контекст. Последње две теме (улога и историјат рачунарских система) погодне су и за самосталну обраду и могу се обрађивати у облику семинарских радова.

У оквиру теме **Дигитални запис података** прецизно увести механизме записа различитих типова података (бројева, текста, слика, звука, видеа) у облику (бинарно записаних) бројева. Дефинисати појам дигитализације и однос између дигиталног (дискретног) и аналогног (континуалног) записа. Истакнути препреке дигиталном запису (сложеност потребне технологије), али и предности које дигитални запис доноси једном када се реализује (трајност, прављење идентичних копија, једноставност обраде, пренос и слично). Аналогну технологију илустровати на примеру аналогне фотографије и аналогног записа звука на магнетне траке и грамофонске плоче (имати у виду да се ученици вероватно нису уживо сретли са том технологијом). Insistirати на томе да дигитализација представља запис података у облику бројева (не обавезно бинарних), а да се ти бројеви најчешће записују бинарно, зато што је уређаје и меморије са два различита стања једноставније реализовати него уређаје са више дискретних стања.

Позициони запис декадних бројева са којим се ученици сусрећу од раних разреда основне школе уопштити на произвољну бројевну основу. Фокусирати се на основе 10 (декадни бројеви), 2 (бинарни бројеви), 16 (хексадекадни бројеви) и 8 (октални бројеви). Ученицима демонстрирати конверзије записа природних бројева уз помоћ калкулатора. Увести различите алгоритме преводјења бројевних основа (из дате основе у основу 10 и из основе 10 у дату основу). Insistirати на вези између бинарног и хексадекадног и бинарног и окталног система и хексадекадне и окталне бројеве користити као начин скраћивања записа бинарних бројева.

Увести појам записа неозначених бројева са фиксираним бројем битова. Дискутовати интервал бројева који се могу записати за

различит број битова (интервал од 0 до 2^n-1). Једноставности ради разматрати углавном осомбитне бројеве, али обезбедити да ученици буду свесни да се бројеви у рачунарима данас обично записују помоћу 32 бита. Дефинисати алгоритам сабирања неозначених бројева, уз увођење појма прекорачења приликом сабирања. Увести и алгоритам множења неозначених бројева. Током излагања пожељно је вући паралеле са програмским језиком који ученици изучавају (на пример, дискутовати типове unsigned char и unsigned int у језику C или са типовима byte и uint у језику C#).

Увести запис означених бројева у облику означене апсолутне вредности. Дискутовати проблеме извођења аритметичких операција над тако записаним бројевима и истакнути проблем двоструког записа нуле. Након тога, као решење проблема сабирања и одузимања увести запис означених бројева у облику потпуног (другог) комплемента. Дискутовати интервал бројева који се могу записати за различит број битова (интервал од -2^{n-1} до $2^{n-1}-1$). Увести Бутов алгоритам за множење бројева записаних у потпуном комплементу. Током излагања пожељно је вући паралеле са програмским језиком који ученици изучавају (на пример, дискутовати типове signed char и int у језику C или са типовима char и int у језику C#).

Увести запис разломљених бројева и то у облику фиксног и у облику покретног зареза. Дискутовати однос ова два типа записа и навести њихове предности и мане, као и могуће ситуације у којима је њихова употреба пожељнија. Insistirати на проблемима који настају због природе записа у покретном зарезу са ограниченим бројем битова мантисе (на пример, не могућност прецизног записа бројева попут 0,1). Дискутовати стандард IEEE754 и поменути записе специјалних вредности које овај стандард уводи (бесконечне и денормализоване вредности). Укратко описати и начине извођења операција над бројевима у покретном зарезу. Дискутовати интервал бројева који се могу записати за различит број битова експонента и мантисе. Током излагања пожељно је вући паралеле са програмским језиком који ученици изучавају (на пример, дискутовати типове float и double у језицима C или C#).

Увести стандардне начине кодирања текста у рачунару. Кренути од таблице ASCII, скупа карактера који она покрива и описати њених својстава. Након тога увести и таблицу Unicode и у њој приказати карактере који се користе за запис текстова на српском језику. Увести кодирања UCS-2, UTF-8, UTF-16 и дискутовати их детаљно, на нивоу битова. Заинтересованим ученицима могуће је приказати и једнобајтна проширења ASCII таблице (на пример, таблице ISO-8859 и Windows-1250, Windows-1251).

Увести начине записа растерских слика и разне начине представљања боја (RGB, CMYK, HSB). Укратко и неформално описати и могуће начине компресије слике. без и са губитком и поменути најчешће коришћене формате за чување слика. Описати и технику дигиталног записа звука. Дефинисати појам узорка (семпла), фреквенцију узорковања и описати везу са Најквист-Шеноновом теоремом. Објаснити и шта је вишеканално снимање звука (стерео, 5+1, 7+1). Укратко описати и идеје технике компресије са губитком и без губитка и поменути и најчешће формате записа некомпримованог и компримованог звука (wav, mp3, flac). Објаснити и начин записа видео-материјала и поменути најчешће формате записа и контејнерске формате (avi, mp4, mkv).

Тему **Логичке основе обраде података** започети кратким подсећањем на Булову алгебру логике (исказне формуле и логичке везнике) коју су ученици већ изучавали у оквиру математике. Увести појам логичке функције, конјунктивне нормалне форме (КНФ) и дисјунктивне нормалне форме (ДНФ) и описати поступак њиховог одређивања (на основу таблице истинитосне вредности, као и трансформацијама дате формуле). Поменути и поступак минимализације КНФ и ДНФ (алгебарским трансформацијама, Карновим мапама). Логичка кола пожељно је изучавати у оквиру неког софтверског симулатора (на пример, LogiSim).

На почетку изучавања комбинаторних логичких кола ученицима приказати симболе за представљање елементарних логичких везника (AND, OR, NOT, XOR, NAND, NOR). Приказати како се на основу ДНФ или КНФ може направити секвенцијално коло које представља произвољну логичку функцију. Приказати како се

помоћу ових кола може направити полусабирач, сабирач, кодер, декодер, мултиплексер и демултиплексер и илустровати улогу сваког од тих кола у оквиру неког ширег система. Приказати како се помоћу ових основних кола може креирати једноставна аритметичко-логичка јединица. Демонстрирати како се вишебитни уређаји могу добити компоновањем уређаја који обрађују појединачне битове.

Дефинисати појам секвенцијалних логичких кола и објаснити њихову разлику у односу на комбинаторна кола. Описати разлику између асинхроних и синхроних кола и увести појам системског часовника тј. сата. Дефинисати флип-флоп (тј. резу, енгл. latch) и описати његове облике (SR, D, JK, T), могуће начине имплементације (помоћу NAND или NOR кола) и улогу у изградњи меморијских ћелија. Приказати како се помоћу више флип-флопова може изградити регистар, померачки регистар и бројач и илустровати употребу тих кола у ширем рачунарском систему.

О склопу теме **Основни организације и архитектуре рачунара** ученицима представити основне хардверске компоненте рачунара, принципе њиховог функционисања и улогу у ширем рачунарском систему. Приказати структуру процесора: аритметичко-логичку јединицу, регистре (програмски доступне регистре, програмски бројач, програмску статусну реч) и контролну јединицу и описати начин његовог функционисања (скуп инструкција, инструкциони циклус и његове фазе). Описати и механизам прекида и улогу прекида као реакције на спољашње догађаје. Приказати различите облике привремене и трајне меморије, приказати њихов однос у светлу брзине, цене и капацитета и дефинисати меморијску хијерархију. Истакнути RAM, ROM и (процесорске) кеш меморије и објаснити њихову улогу у систему. Описати системску магистралу као везу између процесора и главне меморије и различите облике магистрале који се срећу у савременим рачунарским системима. У оквиру прегледа улазно-излазних (периферијских) уређаја обрадити начине комуникације са њима и нагласити улогу система прекида за реализацију улаза-излаза. Објаснити разлику између програмираног улаза-излаза и улаза-излаза уз помоћ уређаја за директан приступ меморији (DMA). Објаснити разлику између меморијски-мапираног и изолованог улаза-излаза. Код улазно-излазних компоненти детаљно објаснити појам и принципе функционисања магнетних дискова, SSD уређаја, графичких картица, тастатуре, мишева, штампача и скенера. Објаснити улогу и принципе функционисања универзалне серијске магистрале USB. Објаснити и појам и намену драјвера за периферијске уређаје и њихову везу са оперативним системима. Са ученицима проанализирати у том тренутку актуелну понуду рачунара и рачунарских компоненти и продискутовати неке рачунарске конфигурације које се продају у том тренутку. Укратко, само на нивоу појма и енциклопедијских информација објаснити напредне теме попут вишенитних и вишејезгарних процесора, суперскаларних и векторских процесора и мултипроцесорских система.

О склопу теме **Асемблерско програмирање** илустровати појам асемблерских и машинских језика и истакнути њихову неопходност коришћења и у савременим рачунарским системима (сви виши програмски језици морају се превести на машински језик да би се могли извршити). Пожељно је ову тему илустровати кроз коришћења неком реалног асемблерског језика за x86 архитектуру која је ученицима доступна (на пример, NASM у окружењу SASM). Ученицима приказати рад у изабраном окружењу (уношење програма, превођење, повезивање, покретање програма, дебаговање). Да би се олакшао улаз и излаз наставник може припремити библиотеке које сакривају одређене техничке детаље од ученика. Поступно и детаљно увести конкретан скуп инструкција и илустровати их кроз једноставне примере. Кренути од аритметичких и логичких инструкција и програма линијске структуре, а након увођења инструкција безусловног и условног скока проширити их на једноставне програме разгранате и цикличне структуре. Увести различите начине адресирања (непосредно, регистарско и меморијско директно, регистарско индиректно без помераја и са померајем (релативно)) и објаснити њихову намену. Након увођења и приказивања примера инструкција објаснити и њихову

имплементацију и везу са машинским језиком и хардвером процесора. Објаснити бинарни начин (формат) записивања машинских инструкција у меморију (поља за код операције и операнде, начине адресирања операнда). Увести појам машинског стека и његову улогу у реализацији потпрограма. Приказати имплементацију стека уз помоћ регистара опште намене и општих инструкција, али и уз помоћ специјализованог регистра (SP) и инструкција (PUSH, POP). Увести појам стек-оквира и приказати конвенције позивања потпрограма и враћања резултата потпрограма. Илустровати све ово кроз примере једноставнијих програма и потпрограма на асемблеру. Приказати поступак повезивања асемблерских програма са програмима написаним у језицима вишег нивоа (пре свега са програмима написаним у језику C).

ОБЈЕКТНО ОРИЈЕНТИСАНО ПРОГРАМИРАЊЕ

Циљ и задаци

Циљ наставног предмета Објектно оријентисано програмирање је стицање основних знања о објектно оријентисаној парадигми и њеној примени у решавању практичних проблема.

Задаци наставе предмета Објектно оријентисано програмирање су да ученици:

- упознају основне концепте објектно оријентисаног програмирања као што су: модел, класа, објекат, енкапсулација, везе између класа и полиморфизам, апстрактне класе, генеричке класе и обраду изузетака;
- разумеју начин решавања практичних проблема применом објектно оријентисане парадигме;
- науче да практичне проблеме решавају применом конкретних објектно оријентисаних програмских језика.

III разред

(3 часа недељно, 105 часова годишње + 30 часова наставе у блоку)

1. Настанак и карактеристике објектно оријентисаног програмирања

- Историјски развој објектно оријентисане парадигме.
- Преглед познатих објектно оријентисаних програмских језика.
- Основне карактеристике објектно оријентисане парадигме.
- Моделовање као основа за решавање проблема.
- Домен реалног проблема.
- Релевантне особине.
- Модел.

2. Основни појмови објектно оријентисаног програмирања

- Класа и објекат.
- Однос између класе и објекта.
- Инстанцирање класе.
- Чланови класе.
- Заједнички чланови класе (*static*)
- улога поља и метода
- декларација поља и методе
- права приступа
- приступ пољу и позив методе
- конструкција и деструкција објекта
- улога и врсте конструктора
- позив конструктора
- начини деструкције објеката (деструктор или *Garbage Collector*)
- примери конструкције и деструкције објеката у различитим објектно оријентисаним програмским језицима.

3. Везе између класа и полиморфизам

- Подела веза између класа.
- Асоцијација.
- Агрегација.
- Композиција.

- Наслеђивање.
- Поља и методе изведене класе.
- Хијерархија класа.
- Улога и врсте полиморфизама.
- Статички и динамички полиморфизам.
- *Ad hoc* полиморфизам.
- Параметарски полиморфизам.
- Подтип.
- Виртуалне методе.
- Апстрактне методе.
- Улога апстрактних класа и интерфејса.

4. Израда пројектног задатка

- Појам и улога генеричких класа.
- Примери примене генеричких класа у различитим објектно оријентисаним програмским језицима.
- Значај обраде изузетака.
- Механизам креирања изузетка.
- Механизам обраде изузетка.
- Примена готових класа.
- Рад са структурама података.
- Рад са датотекама.
- Израда сложеног пројектног задатка.

НАЧИН ОСТВАРИВАЊА ПРОГРАМА

Настанак и карактеристике објектно оријентисаног програмирања (9)

Упознати ученике са настанком и историјским развојем објектно оријентисане парадигме. Направити преглед и анализу основних карактеристика познатих објектно оријентисаних програмских језика (Simula67, Smalltalk, C++, Java, C#). Анализирати основне карактеристике објектно оријентисане парадигме и објектно оријентисани приступ у решавању практичних проблема. Истаћи значај моделовања као основе за решавање проблема у оквиру објектно оријентисане парадигме. На конкретним примерима објаснити поступак моделовања – посматрање домена реалног проблема, избор релевантних особина и добијање модела.

Основни појмови објектно оријентисаног програмирања (24)

Упознати ученике са основним појмовима објектно оријентисаног програмирања – класа и објекат. Објаснити однос између класе и објекта. Објаснити улогу чланова класе – поља и методе. Анализирати начине и права приступа пољима и методама. Кроз једноставне примере (класе *Зраво*, *Троугао*, *Особа* и сл.) упознати ученике са начином израде објектно оријентисаних програма.

Упознати ученике са механизмима конструкције и деструкције објеката. Размотрити врсте конструктора и начине деструкције објекта (деструктор или *Garbage Collector*). Анализирати примере конструкције и деструкције објеката у различитим објектно оријентисаним програмским језицима (нпр. разлике између C++ и Java).

Везе између класа и полиморфизам (30)

Размотрити поделу веза између класа и детаљније објаснити асоцијацију, агрегацију, композицију и наслеђивање. Кроз конкретне примере објаснити однос између власника и компоненте у оквиру композиције, начин конструкције и деструкције компоненте и власника, као и начин коришћења компоненте од стране власника (нпр. класа Ваљак која садржи компоненте Основа и Омотач). Кроз конкретне примере објаснити смисао наслеђивања и добијања хијерархије класа (нпр. из класе *Особа* извести класе *Ученик*, *Студент*, *Запослени*, а из класе *Запослени* извести класу *Наставник*), однос између основне и изведене класе, приступ преузетим члановима у изведеној класи, додавање нових чланова у изведеној класу, као и начин редефинисања метода у изведеној класи.

Упознати ученике са улогом и врстама полиморфизама. Размотрити статички и динамички полиморфизам, *ad hoc* полиморфизам, параметарски полиморфизам, подтип и виртуалне методе.

Кроз конкретне примере објаснити улогу виртуелних метода (нпр. виртуелна метода *представиСе()* из класе *Особа* која је редефинисана у класи *Запослени*).

Објаснити разлику између виртуелних и апстрактних метода. Објаснити дефиницију апстрактних класа и интерфејса. На конкретним примерима објаснити улогу апстрактних класа и интерфејса у хијерархији класа (нпр. апстрактна класа *Фигура* из које се изводе класе *Троугао* и *Круг*).

Израда пројектног задатка (42)

Објаснити појам и улогу генеричких класа. Размотрити идеју увођења генеричких класа у различитим објектно оријентисаним програмским језицима. Размотрити примере примене генеричких класа у различитим објектно оријентисаним програмским језицима. Објаснити могућност појаве отказа у објектно оријентисаном програму и значај механизма обраде изузетака у датим ситуацијама. На конкретним примерима објаснити принципе рада механизма креирања и обраде изузетака (нпр. креирање изузетака у класи *Стек*).

Анализирати библиотека готових класа у различитим објектно оријентисаним програмским језицима. Објаснити значај готових класа у изради сложених објектно оријентисаних програма.

Размотрити рад са структурама података у објектно оријентисаним програмским језицима и ситуације у којима се могу користити готове класе. Размотрити рад са датотекама у објектно оријентисаним програмским језицима. Кроз израду сложеног пројекта повезати стечено знање (нпр. израда апликације за вођење евиденције у школама где би се подаци чували у датотекама и користиле готове класе) и на тај начин упознати ученике са могућностима објектно оријентисаног програмирања.

НАПОМЕНА: Предвиђена су два двочасовна писмена задатка са исправком (6 часова)

Практичне вежбе кроз наставу у блоку (30)

1. Основни појмови објектно оријентисаног програмирања (6)
2. Везе између класа (6)
3. Полиморфизам (6)
4. Израда пројектног задатка (12).

ОПЕРАТИВНИ СИСТЕМИ И РАЧУНАРСКЕ МРЕЖЕ

Циљ и задаци

Циљ наставног предмета Оперативни системи и рачунарске мреже је стицање основних знања о карактеристикама оперативних система и рачунарских мрежа, ради правилног конфигурирања и успешног коришћења у пројектовању савремених рачунарских система.

Задаци наставе предмета Оперативни системи и рачунарске мреже су да ученици:

- упознају основне сервисе оперативних система као што су: управљање датотекама, управљање меморијом, управљање процесима, управљање уређајима и мрежни сервиси;
- разумеју појам рачунарске мреже, упознају се са мрежним слојевима и протоколима, стекну основна знања о безбедности рачунарских мрежа.

II разред

(2 часа недељно, 72 часа годишње)

1. Увод у оперативне системе

- Оперативни системи.
- Основни концепти оперативних система.
- Архитектура оперативних система
- Развој оперативних система и историјат.
- Значајни оперативни системи.

2. Процеси

- Процеси.
- Стања процеса.
- Контролни блок процеса.

- Нити.
- Редови процеса.
- Планери.
- Вишепроцесорски системи.

3. Конкурентност и синхронизација процеса

- Критична секција.
- Решења заснована на активном чекању.
- Решења за заштиту критичне секције без активног чекања.

4. Заглављивање

- Мере за спречавање заглављивања.
- Мере избегавања.
- Детекција заглављивања.

5. Управљање меморијом

- Управљање меморијом при монопрограмирању.
- Управљање меморијом при мултипрограмирању.
- Страничење.
- Сегментација.

6. Систем датотека

- Интерфејс система датотека.
- Директоријуми.
- Права приступа.
- Структура и имплементација система датотека.

7. Управљање улазно-излазним уређајима

- Хардверске компоненте.
- Интерфејс уређаја.
- Драјвери.
- Софтвер за управљање који не зависи од уређаја.
- Интерфејс ка корисничким процесима.

8. Рачунарске мреже

- Класификација мрежа.
- Историја Интернета.
- Архитектура мреже.
- Референтни модел *OSI*.
- Референтни модел *TCP/IP*.
- Безбедност рачунарских мрежа.

НАЧИН ОСТВАРИВАЊА ПРОГРАМА

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

- Увод у оперативне системе (8)
- Процеси (6)
- Конкурентност и синхронизација процеса (10)
- Заглављивање (6)
- Управљање меморијом (12)
- Систем датотека (8)
- Управљање улазно-излазним уређајима (10)
- Рачунарске мреже (12).

Увод у оперативне системе

Упознати ученике са основним концептима оперативних система. Посебну пажњу посветити функцијама језгра оперативног система, системским позивима, драјверима и корисничком окружењу.

Анализирати различите архитектуре оперативних система: Монолитне системе, Слојевите системе, Системе засноване на микројезгру, Хибридне системе и Системе засноване на егзојезгру.

Упознати ученике са историјатом и развојем оперативних система. Представити оперативни системе значајне за развој рачунарства:

- Multics;
- Оперативни системи UNIX фамилије;

- GNU/Linux;
- Оперативни системи компаније Microsoft;
- Оперативни системи компаније Apple;
- Андроид.

Процеси

Објаснити појам процеса и његово место у меморији. Навести и продискутовати стања у којима се може наћи процес. Описати структуру у којој се чувају подаци о процесима – контролни блок процеса.

Објаснити концепт нити и предности које овакав приступ доноси. Илустровати примерима (едитор текста, веб прегледач, сервер, итд.).

Приказати редове процеса и улогу планера за што ефикасније функционисање система.

Продискутовати вишепроцесорске системе.

Конкурентност и синхронизација процеса

У оквиру ове целине треба објаснити појам критичне секције и приказати нека од решења за њену заштиту.

Прво би требало обрадити решења заснована на активном чекању:

- Стриктна алтернација;
- Декеров алгоритам;
- Питерсонов алгоритам;
- Лампортов (пекарски) алгоритам.

Затим решења заснована на коришћењу хардверских инструкција (TAS, SWAP, FAA) и на крају најпознатија решења за заштиту критичне секције без активног чекања:

- Семафори;
- Критични региони;
- Монитори.

Заглављивање

Објаснити шта је заглављивање и који су услови неопходни да би до њега дошло. Приказати мере за спречавање заглављивања:

- Превенција чекања и држања;
- Елиминисање немогућности прекидања;
- Превенција кружног чекања.

Продискутовати Банкарев алгоритам као меру која се предузима да би се избегло заглављивање.

Приказати начине на које се обично проверава да ли је у систему дошло до заглављивања и продискутовати начине на које се систем може опоравити од заглављивања.

Управљање меморијом

Приказати основне проблеме који се јављају при управљању меморијом. Описати страничење као начин за управљање меморијом. Објаснити улогу табеле страница и предности које доноси коришћење асоцијативне меморије. Приказати сегментацију као алтернативни начин за управљање меморијом и упоредити га са страничењем.

Објаснити појам виртуелне меморије као приступа којим се раздваја расположив део меморије од оног који стварно физички постоји. Анализирати страничење на захтев као један од начина за имплементирање виртуелне меморије.

Обрадити алгоритме за избацивање странице:

- Алгоритам заснован на случајном избацивању;
- Беладидејев оптимални алгоритам;
- FIFO алгоритам;
- Алгоритам друге шансе;
- Алгоритам сага.

Систем датотека

Објаснити појам датотеке, система датотека и интерфејса система датотека. Приказати улогу атрибута датотека, операције које се могу извршити над датотекама, структуром и типове датотека.

Објаснити улогу директоријума и могуће организације:

- Организација директоријума – један ниво;
- Организација директоријума – два нивоа;
- Организација директоријума у структуру стабла.

Објаснити шта су то апсолутне а шта релативне путање и на који начин се могу дефинисати права приступа.

Обрадити структуру и имплементацију система датотека, детаљно објаснити како се могу имплементирати датотеке и директоријуми.

Управљање улазно-излазним уређајима

Детаљно описати хардверске компоненте које представљају улазно-излазне уређаје.

Представити начине повезивања уређаја у рачунарски систем. Посебно објаснити улогу прикључака, магистрала, контролера и регистара. Приказати на које начине процесор може комуницирати са уређајима.

Детаљно обрадити основне приступе за управљање уређајима:

- Техника прозивања;
- Прекиди;
- Директан меморијски приступ – ДМА.

Објаснити улогу софтвера за управљање који не зависи од уређаја. Посебно обрадити:

- Планирање улазно-излазних операција;
- Баферовање;
- Обрада грешака;
- Кеширање;
- Спулер.

Описати интерфејс ка корисничким процесима односно механизам којим се корисничким процесима обезбеђује коришћење уређаја на највишем нивоу.

Рачунарске мреже

Дефинисати појам рачунарске мреже и приказати основне класификације мрежа:

- Класификација на основу технологије преноса;
- Класификација на основу величине.

Продискутовати историјски развој интернета и рачунарских мрежа.

Објаснити појам архитектуре мреже и приказати референтни модел TCP/IP. Посебно објаснити начин IP адресирања и рутирања. Представити UDP протокол и TCP протокол.

Приказати главне претње за безбедност рачунарских мрежа и теоријски поделу могућих напада. Посебно обрадити практичне нападе и начине за одбрану од напада на мрежи.

БАЗЕ ПОДАТАКА

Циљ и задаци

Циљ наставног предмета Базе података је стицање основних знања о техникама пројектовања база података као одговора на пословну потребу за информационим системима.

Задаци наставе предмета Базе података су да ученици:

- упознају концепт база података, њихову организацију, коришћење упита за добијање тражених информација из база, прављење извештаја и дистрибуцију података;
- ефикасно користе програмирање и рад са базама података за решавање различитих проблема у даљем образовању, професионалном раду и свакодневном животу.

III разред

(2 часа недељно, 70 часова годишње)

1. Пројектовање база података

- Подаци и потреба за базама података. Релациона база података;
- Логички модел као пројекат за креирање базе података;
- Ентитети (објекти), атрибуту, везе;

- Моделовање специфичних ситуација;
- Нормализација модела.

2. Релационе базе података

- Припрема за израду релационе базе података на основу логичког модела;
- Табела. Примарни кључ, страни кључ и друга ограничења.

3. Упитни језик SQL

- Упитни језик SQL за рад са релационом базом података;
- Упит SELECT са многобројним могућностима;
- Команде језика SQL за креирање табела и погледа;
- Команде језика SQL за обраду података (INSERT, UPDATE, DELETE);
- Трансакције;
- Администрација базе и вишекориснички рад.

IV разред

(2 часа недељно, 62 часова годишње + 30 часова наставе у блоку)

1. Програмирање и базе података (14 + 32)

– Писање програма који у себи имају команде упитног језика SQL и обрађују податке из базе података (на један од два описана начина)

- Писање програма унутар система за управљање базама података (процедуре, функције, тригери, курсори)
- Писање програма у неком програмском језику вишег нивоа (Java, C#...) који има угњежене команде упитног језика SQL.

2. Друге актуелне технологије (8 + 8)

- Други начини организовања података. Актуелне технологије.
- XML као модел података, структура, предности и недостаци.

НАЧИН ОСТВАРИВАЊА ПРОГРАМА

III разред

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Пројектовање база података (24)

Релационе базе података (14)

Упитни језик SQL (32)

Предвиђена су два двочасовна писмена задатка са једночасовним исправкама (6 часова).

Ученици треба да се упознају са потребом креирања база података и са примерима база података које виђају свакодневно. Дефинисати појам релационе базе података и систем за управљање базама података.

Програм реализовати кроз што више практичног рада и примера. Важно је да ученици виде примере из живота у којима се ради са великим количинама података (нпр. библиотека, видеотека, продавница, банка, пошта, школа, спортски клуб...), и да прођу све од пословног захтева до креирања конкретне базе података.

У оквиру тема **Пројектовање база података и Релационе базе података потребно је:**

- Објаснити ученицима важност фазе пројектовања базе података чији је резултат модел објекти-везе.
- Упознати ученике за изабраном нотацијом за опис модела података. Нагласити како се води рачуна о интегритету базе у фази пројектовања.

– Дефинисати ентитет (објекте) и атрибуте, и везе између ентитета. Објаснити кардиналност и опционалност везе и различите типове веза према кардиналности (1:1, 1:M, M:M). Увести појам примарног идентификатора (кандидат за примарни кључ).

– Приказати примере модела којима се решавају потребе за базом података у разним пословањима (нпр. библиотека, видеотека, продавница, банка, пошта, школа, спортски клуб...).

– Објаснити нормализацију и правила прве, друге и треће нормалне форме.

– Описати релациони модел података од атрибута, домена, релацијске шеме, релације. Објаснити шта је интегритет релационог модела података. Дефинисати општа правила интегритета (правила за примарни кључ, страни кључ).

– Приказати како се модели података конвертују у релациони модел, односно релациону базу података. Објаснити промену терминологије (ентитет-табела, атрибут-колона, инстанца-ред, примарни идентификатор-примарни кључ, веза-страни кључ).

Током обраде ових тема, предлаже се да се ученици поделе у тимове (2 до 4 члана) и да раде на пројектном задатку. Потребно је да изабере неко пословање (нпр. спортски клуб, туристичка агенција, рен-а-кар компанија, сервис рачунара, банка, продавница...) и да креирају модел података са пратећом документацијом и презентацијом.

У оквиру теме **Упитни језик SQL** потребно је:

– Упознати ученике са основама програмирања у језику SQL. Истаћи важност упита SELECT којим претражујемо базе података и којим добијамо тражене и корисне информације. Описати селекцију, пројекцију и спајање табела.

– Вежбати са ученицима задатке којима се тражи да из базе података добију информације помоћу упита SELECT (пројекцијом, селекцијом, спајањем табела). Приказати и провежбати сложене упите и подупите. Кроз примере приказати и провежбати употребу разних функција. Објаснити важност креирања погледа VIEW и дати примере.

– Упознати ученике са наредбама за креирање објеката. Посебну пажњу посветити наредби CREATE TABLE и различитим типовима података. Објаснити појам ограничења (NOT NULL и UNIQUE KEY), као и примарни и страни кључ. Упознати ученике са другим објектима у бази (секвенце, индекси, процедуре, функције).

– Објаснити и провежбати наредбе за манипулисање подацима: унос у базу, брисање и измену (INSERT, DELETE и UPDATE).

– Истаћи важност администрације базе података: корисници, додељивање и одузимање права корисницима, роле.

Током часова посвећених овим темама, важно је да ученици вежбају задатке који се решавају у језику SQL. Највише пажње посветити упиту SELECT којим се добијају информације из података који се чувају у бази. Потребно је да ученици и креирају бар једну базу података наредбама CREATE TABLE.

IV разред

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Програмирање и базе података (14 + 32)

Друге актуелне технологије (8 + 8)

Предвиђена су два двочасовна писмена задатка са једночасовним исправкама (6 часова).

Објаснити како се ради са базама података у апликацијама кроз примере писања програма који у себи имају команде упитног језика SQL и обрађују податке из базе података. Изабрати један од два начина:

– Писање програма унутар система за управљање базама података (процедуре, функције, тригери, курсори)

– Писање програма у неком програмском језику вишег нивоа (Java, C#...) који има угњеждене команде упитног језика SQL.

У оквиру теме **Друге актуелне технологије** постоји простор да се ученици упознају и са другим начинима организовања података, као што је XML, као и са актуелним технологијама. Важно је да се ученици заинтересују да прате промене које се дешавају у свету информационих технологија. Потребно је код њих створити широку слику о томе које све технологије постоје за исте области рада, и развити механизме да могу да упоређују и анализирају различите алате и методе.

Настава у блоку (30)

1. Пример једне базе података од модела до коначног решења и рада са подацима (10)

2. Пројектни задатак – пројектовање базе (10)

3. Пројектни задатак – креирање базе података и рад са подацима (10).

ПРОГРАМСКЕ ПАРАДИГМЕ

Циљ наставног предмета Програмске парадигме је развој различитих приступа решавању проблема код ученика коришћењем различитих концепата програмирања.

Задаци наставе предмета Програмске парадигме су да ученици:

– овладају напреднијим концептима програмирања;

– упознају се са различитим приступима решавању проблема програмирањем, различитим програмским парадигмама и савладају различите програмске језике који те парадигме илуструју;

– јачају способност решавања проблема развојем логичког и критичког мишљења;

– јачају способност за прецизно и концизно дефинисање проблема; упознају се са потребом за коришћењем решавања проблема програмирањем и у другим областима (нпр. у математици и техници или у дефинисању пословних процедура и протокола);

– оспособе се за рад на пројектима, који захтевају примену знања из других наставних предмета, и који подразумевају креирање решења на рачунару за дефинисани проблеме и израду конкретних апликација, са пратећом документацијом и презентацијама;

– унапреде стратегије и технике самосталног учења користећи могућности рачунара, изграде спремност за праћење нових решења у области информатичке технологије и развију спремност за учење током целог живота.

IV разред

(2 часа недељно, 62 часа годишње)

1. Увод.
2. Исказна логика.
3. Предикатска логика.
4. Логичко програмирање.
5. Функционално програмирање.

НАЧИН ОСТВАРИВАЊА ПРОГРАМА

У оквиру теме **Увод (4)** потребно је:

Упознати ученике са основним особинама декларативног програмирања, разликом између релационих и функционалних програмских језика, начином описивања проблема у декларативним програмским језицима.

У оквиру теме **Исказна логика (8)** потребно је:

Обновити градиво из исказне логике, обрађивано у оквиру математичке групе предмета, посебно у Дискретној математици (Искази, Исказне формуле, Истинитосна вредност исказних формула, конјунктивне нормалне форме (КНФ) и дисјунктивне нормалне форме (ДНФ)).

Упознати ученике са методама провере да ли је тврђење логичка последица других тврђења или не.

– ДПЛЛ алгоритам – *Davis-Putnam-Logemann-Loveland* алгоритам, са провером обе могуће истинитосне вредности које се појављују у формули.

– Метод резолуције.

У оквиру теме **Предикатска логика (10)** потребно је:

Дефинисати предикатске формуле и њихову интерпретацију. Приказати ученицима представљање произвољне предикатске формуле у облику логичког програма кроз фазе:

– Пренекс нормална форма.

– Сколемизација.

– Супституција.

– Унификација.

Објаснити метод резолуције.

У оквиру теме **Логичко програмирање (22)** потребно је:

- упознати ученике са синтаксом изабраног програмског језика;
- упознати ученике са различитим врстама програмских клаузула (чињенице, правила и циљеви);
- упознати ученике са процесом израчунавање одговора, објаснити стабло израчунавања одговора;
- дефинисати сложене структуре података, листе, као структуре разноврсних података са утврђеним редоследом, чијим елементима се приступа од првог елемента. Обавезно нагласити рекурзивну структуру листе.

Дефинисати основна правила за рад са листама

- Припадност листи.
- Спајање две листе.
- Брисање елемента из листе.

Дефинисати правила за решавање комбинаторних проблема

- Пермутације.
- Варијације.
- Комбинације.

Дефинисати правила за решавање логичких проблема

- Ајнштајнов проблем кућа.
- Мисионари и људождери.

Дефинисати појам експертског система и креирати једноставан експертски систем за препознавање различитих облика, животиња, предмета...

У оквиру теме **Функционално програмирање (14)** потребно је:

- упознати ученике са типовима и класама типова;
- упознати ученике са синтаксом израза и функцијама (ламбда изрази, Каријев запис, let ... in, where, if-then-else, case);
- упознати ученике са рекурзивним функцијама и функцијама вишег реда (map, filter, fold) и реализовати са ученицима неке рекурзивне функције;
- дефинисати типове и класе типова (алгебарски типови података, параметарски полиморфизам);
- дефинисати функторе, монаде;
- упознати ученике са улазом/излазом програма и стандардним библиотекама.

ВЕБ ПРОГРАМИРАЊЕ

Циљ и задаци

Циљ наставног предмета Веб програмирање је упознавање са савременим веб-технологијама и њихово коришћење у циљу креирања веб-садржаја који одговарају савременим пословним и личним потребама корисника.

Задаци наставе предмета Веб програмирање су да ученици:

- упознају концепте веб програмирања и језике за опис садржаја, изгледа и понашања веб страна;
- буду оспособљени да, на основу познавања језика HTML и CSS, протумаче елементе веб-странице, прилагоде их и креирају визуелно допадљиве странице које садрже линкове, слике, листе, табеле и сличне елементе;
- буду оспособљени да креирају сложеније вишеслојне веб-апликације које омогућавају кориснику да кроз формулар уноси одређене податке, добије жељене податке, претражује, ажурира или брише податке из базе података.

IV разред

(2 часа недељно, 62 часа годишње)

1. Рачунарске мреже

- Појам рачунарске мреже.
- Компоненте рачунарских мрежа.

- Врсте мрежа.
- Топологија мрежа.
- Сложитост мрежа.

2. Интернет сервиси и протоколи

- Историјат интернета.
- Интернет сервиси.
- Интернет протоколи.

3. Описни језик HTML

- Основе језика HTML.
- Структура HTML стране.
- Заглавље веб-странице и метаподаци.
- Целине у HTML документу.
- Пасуси, листе и адресе.
- Опис текста.
- Везе.
- Табеле.
- Уметнути садржај.
- Глобални атрибути.
- Генерички блок елементи.
- Формулари.

4. Стилски листови – језик CSS

- Увод у CSS.
- Укључивање стилских листова у HTML документе.
- Селектори.
- Најчешће коришћена својства и њихове вредности.
- Пример стилизовања веб-странице.

5. Скрипт језик JavaScript за клијентско програмирање

- Потреба и стандардизација.
- Уметање скрипта у веб-странице.
- Основе језика JavaScript
- наредбе
- променљиве, елементарни типови и оператори
- контролне структуре
- JavaScript објекти.

6. Серверско програмирање

Представити један од два најчешће коришћена серверска окружења:

PHP, MySQL

- Увод у PHP.
- Програмирање са PHP.
- Креирање динамичких веб-сајтова.
- Увод у MySQL.
- Пројектовање базе података.
- PHP и MySQL.
- Развијање веб апликација.
- Управљање сесијама у PHP.
- Објектно оријентисани PHP.
- Примери:
 - формулар за регистрацију;
 - веб форум;
 - интернет продавница.

ASP.NET, Visual studio, C#

- Окружење Visual Studio.
- Преглед програмског језика C#.
- Увод у базе података.
- Приказивање база података.
- Приступање бази података помоћу кода.
- Креирање ASP.NET веб-сајта.

НАЧИН ОСТВАРИВАЊА ПРОГРАМА

Ради лакшег планирања наставе даје се оријентациони предлог броја часова по темама.

Рачунарске мреже (4)

Интернет сервис и протоколи (4)

Описни језик HTML (8)

Стилски листови – језик CSS (8)

Скрипт језик JavaScript за клијентско програмирање (8)

Серверско програмирање (30).

Рачунарске мреже

Ученике подсетити, јер се са овим појмовима срећу у другим предметима у другом и трећем разреду, шта чини рачунарску мрежу. Навести све основне улоге рачунарске мреже и основне компоненте рачунарске мреже: мрежни хардвер, мрежни софтвер и комуникационе канале. Обновити појмове који се односе на мрежни хардвер (рутере, свичеве, хабове, мостове, модеме, ...) и комуникациону опрему (каблове и технике бежичне комуникације).

Поновити различите врсте рачунарске мреже (*LAN*, *WAN*), четири главна типа топологије мреже. Објаснити значај слојевитости мрежа, представити два основна референтна модела (*OSI* модел, *TCP/IP* модел) и описати најзначајније слојеве референтног модела (физички слој, мрежни слој, транспортни слој, апликациони слој).

Интернет сервис и протоколи

Потребно је да ученици: разумеју све функције интернета, од почетне идеје глобалног умрежавања, проналажења и давања информација; да разумеју појам веб-а (*www*). Само информативно ученике упознати са историјатом интернета при чему више пажње посветити технологијама приступа Интернету (*DSL* технологије, кабловски интернет, бежичне технологије, мреже мобилне телефоније).

Навести основне интернет сервисе: базични сервис (електронска пошта, комуникација између удаљених рачунара, пренос података између удаљених рачунара), јавни информациони сервис, дискусионе групе, конференцијски сервис и сервис за претраживање интернета.

Код интернет протокола објаснити мрежне протоколе, протоколе транспортног слоја и протоколе апликационог слоја.

Описни језик HTML

Ученике упознати са два основна приступа код креирања мултимедијалних садржаја: *WYSIWYG* (енгл. *what you see is what you get*) где корисник одмах види шта креира и други где се користи посебан едитор при чему се користе инструкције језика за обележавање.

Указати на основну поделу на језике који описују садржај веб-странице, језике који описују стил веб-странице (избор фонта, боја, форматирање текста...) и језике за опис понашања веб-странице.

Објаснити да је најкоришћенији језик за опис садржаја веб-странице *HTML* језик и представити најновије верзије и стандарде који су везани за овај језик.

У опису синтаксе *HTML* језика објаснити шта чини документ, како се они означавају (тагови) и коришћење атрибута за њихово додатно описивање. Као препоруку дати најједноставнији тип едитора (*Notepad*) или напреднију верзију овог едитора (*Notepad ++*). Објаснити појам метаподатака и зашто се баш они везују за *HTML* језик. Дати поделу садржаја елемента *html* на елементе *head* и *body*. Објаснити елемент за опис наслова веб-странице *title* и друге елементе који се користе за опис метаподатака у заглављу странице: *meta*, *link*, *style* и *script*. У оквиру *meta* елемента посебно објаснити кодирање карактера *UTF-8* и кодне шеме за кодирање ћириличног и латиничног текста. Дати поделу елемената који описују целине *HTML* документа (*main*, *header*, *footer*, *nav*, *aside*), као и елементе за означавање нивоа наслова. Објаснити елементе за опис посебних секција *section* и *article*.

Дати преглед елемената за опис пасуса (*paragraph*), листе (*ul*, *ol*, *il*), адресе (*address*), цитата (*blockquote*) и за опис текста (*i*, *b*, *u*, *em*, *strong*, *small*). Објаснити апсолутно и релативно адресирање.

Код описа табела дати преглед елемената *table*, *tr*, *td*, *th* и *td*. Објаснити елементе за уметање мултимедијалних садржаја као што су слика (*img*), друга веб-страница (*iframe*), аудио и видео-записи (*audio*, *video*, *source*).

Објаснити потребу коришћења генеричких елемената који су додељују свим елементима и дати пример најчешће коришћених. Објаснити елементе *div* и *span* за груписање садржаја. Урадити пример најједноставнијег формулара и при томе користити кратке програме скрипт језика.

Стилски листови језик CSS

Ученике треба упознати са разлогом увођења *CSS* језика, појмом декларације и начином записивања. Дати преглед основних правила који се користе у оквиру стилова.

Код опште синтаксе стилских листова представити основне селекторе за запис елемената. Описати начине укључивања стилских листова у *HTML* документ и наслеђивање стилских описа. Објаснити најчешће коришћене селекторе, својства и њихове вредности: фонт, назив фонта, величина фонта, варијанте фонта, стилизовање текст, поравњање текста, боја. Представити модел кутије (*box*), садржаја (*content*) и оквира (*border*). Објаснити *width* и *height* својства за одређивање ширине и висине елемената, *margin* за одређивање спољне маргине, разне варијанте својство *border* за одређивање оквира (дебљина, врста линије и боја, својство *background* за одређивање позадине елемената. Дати својства за стилизовање листа и табела. Објаснити релативно, статичко и апсолутно позиционирање и својство *position* које то описује.

При реализацији ове и претходне тематске целине радити са најједноставнијим едитором, подстаћи ученике да примене стечена знања из рада са програмима за обраду текста и програмима за обраду слика и тако припреме садржај за креирање сопствене веб-странице.

Као проектни рад ученици треба да своје теме за матурски рад обраде коришћењем *HTML* и *CSS* језика и презентирају га у облику веб-странице. При бирању садржаја инсистирати на критичком приступу информацијама и његовању естетике.

Скрипт језик JavaScript за клијентско програмирање

Објаснити појам скрипт језика, објаснити разлоге и предности увођења *JavaScript* језика, разлику у односу на програмски језик *Java*. Дати *JavaScript* синтаксу и објаснити уметање скриптова унутар веб-странице.

Објаснити декларисање променљивих и коришћење *var* наредбе, основне типове података (бројевни тип, ниске карактера, Буловски тип...). Дати преглед оператора (аритметички, релациони, логички, битски и оператори доделе). Код котролних структура представити стандардне и коришћење наредбе *with*.

Објаснити функције, кључну реч *function* и особине функције у *JavaScript*, променљива чија је вредност функцијског типа. Објаснити обраду изузетака *try catch* блоком.

Дати преглед *JavaScript* објеката: објекти типа *String*, објекат *Math*, објекти типа *Date*, објекти типа *Array*.

На примерима показати израду веб-странице са једноставним формуларима при чему се користе скриптови.

Серверско програмирање

Ученика упознати основном карактеристиком серверског програмирања: корисник (клијент) шаље захтев серверској страни при чему се скрипт извршава на серверу пре него што се клијенту назад пошаље одговор. За разлику од клијентских скриптова као што је *JavaScript*, сервер обезбеђује већину рачунарских ресурса пре слања странице клијенту за приказ у веб прегледачу. Навести мане серверског програмирања као што су мања брзина одзива, зависност од брзине мреже и самих карактеристика сервера. На примерима представити један од два најчешће коришћена серверска окружења и језике који се користе за серверско програмирање:

– представити пример софтвера отвореног кода са бесплатном лиценцом: *PHP* – скриптни серверски програмски језик који је намењен за израду динамичких веб сајтова и најчешће коришћен са њим *MySQL* – систем за управљање релационим базама података;

– представити *ASP.NET* (енгл. *Active Server Pages .NET*) – веб технологију за креирање динамичких веб сајтова, веб апликација и веб сервиса и *C#* – програмски језик који је намењен је изради апликација за *.NET Framework* платформу.

PHP и MySQL

Ученицима објаснити главне карактеристике динамичких веб-сајтова. Нагласити главне карактеристике *PHP* језика да се врло лако уграђује у *HTML*, да је намењен да би се писали веб скриптови и да је серверска технологија (неопходна је апликација веб-сервера да би оно што се напише у *PHP* радило и било функционално). Навести разлоге коришћења *PHP* као језика који је бржи и бољи него други када је реч о изради динамичких веб-сајтова.

Код увода у *PHP* прорадити: основну синтаксу језика, слање података у веб претраживач, писање коментара, променљиве, увођење стрингова, састављање стрингова, увођење бројева, увођење константни, откривање грешке.

Код програмирања са *PHP*-ом прорадити: прављење *HTML* обрасца, руковање *HTML* обрасцем, услове и операторе, испитивање ваљаности података у обрасцу, увођење низова, петље *FOR* и *WHILE*.

Важно је објаснити да приликом креирања динамичких веб-апликација је боље развојити *PHP* скриптове и *HTML* код у посебне датотеке. У вези са тим објаснити апсолутне и релативне путање. Објаснити израду трајних образаца (*HTML* образаца који памте како су попуњени). Објаснити функције: `include ()`, `include_once ()`, `getsite ()`.

Објаснити именовање елемената једне конкретне базе података на *MySQL* серверу, бирање типова и осталих својстава колоне. Објаснити администрирање *MySQL* сервера. Код креирања потребне базе података прорадити: прављење базе података и табела, додавање записа, селектовање података, употребу услова, сортирање резултата упита, ажурирање података, брисање података и употребу функција. Како су се ученици већ срели у трећој години са пројектовањем база података обновити поступак нормализације, доделу примарног кључа, релације међу табелама, прављење индекса, ограничење страних кључева.

Објаснити повезивање са *MySQL* сервером, урадити пример са формом са једноставним упитима (*INSERT*, *UPDATE*, *DELETE*), издвајање резултата упита, ажурирање записа и слање вредности у скрипт.

Као примере најчешће коришћених веб-апликација урадити или показати слање електронске поште помоћу *PHP*, отпремање датотека путем *HTML* обрасца, употребу *JavaScript* -а где треба нагласити разлику између ових технологија. Приликом креирања странице за пријављивање корисника објаснити функције за пријављивање, коришћење колацића и сесија.

Дати основну синтаксу код објектно оријентисаног *PHP*-а, објаснити рад са *MySQL*-ом, једноставне упите и класу *Date Time*.

Поделити ученике у тимове који ће као пројектни задатак да ураде неку од веб апликација: форум, формулар за регистрацију или једноставну интернет продавницу.

ASP.NET и Visual Studio

Представити *Visual Studio* као веб-технологију за креирање динамичких веб-сајтова и веб-апликација. Дати кратак преглед програмског језика *C#* као одабраног језика за креирање веб-апликација, тј прорадити: типове података, операторе, приоритет оператора, управљачке структуре (секвенца, гранање и петља), наредбе скока, низове, класе и објекте, наслеђивање, креирање објеката, приступ члановима класе, конструкторе, деструкторе, изведене класе, апстрактне класе, статичке чланове класе, права приступа, заштићена својства, обраду грешака, именски простор.

Рад у *Visual Studio* окружењу започети прављењем обичног веб-обрасца при чему треба представити све његове компоненте: контроле за податке и системске компоненте. Представити делове *.NET Framework*-а: заједничко језичко извршно окружење

(*CLR*) и библиотеку класа која обезбеђује основне програмске функције. Потребно је објаснити:

- отварање почетне странице као новог или постојећег пројекта и покретање пројекта;
- организацију апликација у пројекте и решења;
- прозоре за документе и прозоре за алатке;
- уређивање веб-обрасца и *HTML* странице визуелно или у *HTML*-у;
- писање кода коришћењем аутоматских функција едитора кода и мењање параметара *Visual Studio .NET*-а;
- постављање преломних тачака и провера променљивих;
- извршавање наредби;
- прављење конекције на серверску базу података;
- контроле за брисање и измену података (*GridView*), за унос и ажурирање података (*FormView*, *ListView*);
- креирање и извршавање апликације, као и отклањање програмских грешака из њих.

Поделити ученике у тимове који ће као пројектни задатак да ураде неку од *ASP.NET* апликација: календар са школским активностима, формулар за регистрацију (проверу идентитета) или једноставну интернет продавницу.

САДРЖАЈ И НАЧИН ПОЛАГАЊА МАТУРСКОГ ИСПИТА

Матурским испитом утврђује се зрелост и оспособљеност ученика за даље школовање.

Матурски испит полагају ученици који су успешно завршили четврти разред гимназије.

САДРЖАЈ МАТУРСКОГ ИСПИТА

Матурски испит састоји се из:

- писменог испита из матерњег језика и књижевности;
- писменог и усменог испита из једног од трипредмета: *програмирање, објектно оријентисано програмирање и базе података*;
- израде и одбране матурског рада.

Сви предмети полагају се према наставном програму који је ученик савладао током четворогодишњег образовања у гимназији за ученике са посебним способностима за рачунарство и информатику.

1. Писмени испит из матерњег језика и књижевности

При оцењивању писменог задатка, испитна комисија има у виду ширину обраде теме, избор и интерпретацију грађе, композицију, стил и језик.

2. Писмени и усмени испит из једног од трипредмета:

програмирање, објектно оријентисано програмирање и базе података.

Приликом оцењивања писменог задатка, испитна комисија има у виду креативност и способност у спровођењу поступка у решавању задатака, као и тачност решавања истих.

3. Матурски рад

Матурски рад са одбраном је самостално обрађена тема коју ученик бира са списка одабраних тема у оквиру једног од следећих предмета:

- математика;
- дискретна математика;
- примена рачунара;
- програмирање;
- рачунарски системи;
- оперативни системи и рачунарске мреже;
- објектно оријентисано програмирање;
- базе података;
- програмске парадигме;
- веб програмирање.

Теме за матурски рад утврђује наставничко веће школе на предлог стручног већа за област предмета. Списак утврђених тема објављује се на огласној табли или доставља ученицима на увид на други погодан начин почетком другог полугодишта за текућу школску годину.

Сврха матурског рада је да ученик покаже колико влада материјом у вези са темом, у којој мери је усвојио методе и приступ обраде теме, како се служи литературом, да ли је оспособљен да анализира, критички размишља и да самостално изрази свој лични став у односу на тему коју обрађује.

Ученик ради матурски рад у току завршног разреда уз помоћ наставника – ментора.

У току израде матурског рада обавезно је организовање најмање четири консултације на којима је ментор дужан прати рад сваког ученика и пружи потребну помоћ упућивањем на потребну литературу и избору начина и структуре израде рада.

Одбрана матурског рада

На усменој одбрани матурског рада ученик је дужан да изложи концепцију свог рада, да наведе литературу и друге изворе знања које је користио, да образложи посебне методе и поступке којима се руководио у току израде матурског рада.

У току одбране матурског рада кандидат треба да покаже знање из целокупног садржаја предмета из којег брани рад.

После одбране матурског рада испитна комисија утврђује једну оцену која се изводи из вредности рада и одбране матурског рада са аспекта способности кандидата да самостално интерпретира материју и да користи савремене методе и изворе информација у процесу стицања новог знања.

ОРГАНИЗАЦИЈА И НАЧИН ПОЛАГАЊА МАТУРСКОГ ИСПИТА

Матурски испит полаже се у два редовна матурска испитна рока: јунском и августовском. После августовског испитног рока ученици полажу ванредно, у роковима које утврди школа.

За полагање матурског испита ученик подноси пријаву школи у року који одреди школа. У пријави се наводи један од предмета који жели да полаже и назив теме за матурски рад. Уз пријаву се прилаже сведочанство о завршеним разредима гимназије и извод из матичне књиге рођених.

Ученику који се пријави за полагање матурског испита и из оправданих разлога буде спречен да полаже испит у целини или поједине делове испита, испитни одбор може да одобри полагање ван редовних испитних рокова.

Ученик може да одустане од полагања испита три дана пре почетка испита, о чему обавештава испитни одбор.

Начин полагања писмених испита

Писмени испит из истог предмета полажу сви ученици истог дана, по правилу, у истој просторији, у присуству најмање једног дежурног наставника.

Писмени испит траје четири школска часа.

Између два писмена испита ученик мора да има слободан дан.

Приликом полагања писменог испита није дозвољено коришћење помоћне литературе.

Теме и задатке за писмени испит предлажу предметни наставници, а испитни одбор, на дан испита, од предложених тема утврђује три теме, односно групе задатака од којих ученик бира једну.

Теме и задатке за писмени испит ученици добијају непосредно пред почетак писменог испита.

Ученик не сме да прекрши испитна правила која утврди школа. На пример: не сме да напусти просторију у којој се обавља писмени испит без одобрења дежурног наставника, не сме да користи недозвољена средства, да преписује од других, да омета друге и слично.

Писменом испиту, поред дежурног наставника (дежурних наставника), могу да присуствују председник испитног одбора и стручњаци које делегира министарство надлежно за послове образовања.

Начин полагања усмених испита

Усмени испит из једног од три наведена предмета: *програмирање, објектно оријентисано програмирање и базе података* полажу ученици који су положили писмени испит.

Полагање усменог испита почиње најраније два дана после положеног писменог испита.

На усменом делу испита ученик извучи испитни листић на коме су исписана три питања, односно задатка. Уколико ученик процени да не може да одговори на питање, може да промени листић, што може да утиче на оцену.

Испитни листић не може два пута бити употребљен истог дана.

Број испитних листића већи је, за сваку испитну комисију, за десет одсто од броја пријављених кандидата.

Списак испитних питања припремају предметни наставници у сарадњи са стручним већем за област предмета и благовремено дају ученицима да би се припремили за матуру.

Одговори ученика на усменом испиту трају до тридесет минута, укључујући и време за припрему ученика за давање одговора.

Усменом испиту, поред чланова испитне комисије, могу да присуствују чланови испитног одбора, наставници школе, стручњаци које делегира министарство надлежно за послове образовања и ученици.

Начин одбране матурског рада

Тему за матурски рад ученик бира са списка утврђених тема. Исту тему за матурски рад не могу радити два или више ученика у истом испитном року.

Ученик предаје матурски рад у року који одреди испитни одбор. Уколико га не преда у предвиђеном року, сматра се да је одустао од полагања матурског испита.

Одбрана матурског рада траје до тридесет минута.

Одбрани матурског рада, поред чланова испитне комисије, могу да присуствују чланови испитног одбора, наставници школе, стручњаци које делегира министарство надлежно за послове образовања и ученици.

Материјал који садржи списак тема и задатака, питања за писмени испит и испитне листиће за усмени испит чувају се као пословна тајна до почетка испита. Материјал чува директор школе.

Испитни одбор и испитне комисије

За спровођење матурског испита директор школе формира испитни одбор и испитне комисије за сваки предмет који се полаже на матурском испиту. Ако један предмет или део испита полаже велики број ученика, директор може да именује већи број испитних комисија за исти предмет. Школа може да ангажује, као чланове испитних комисија, и спољне сараднике.

Испитни одбор чине председник испитног одбора, његов заменик и чланови. Председник испитног одбора је по правилу директор школе. Сви чланови испитних комисија су истовремено чланови испитног одбора. Испитну комисију чине три члана: председник, испитивач и стални члан. Два члана морају бити стручњаци за предмет из кога се полаже испит.

Директор одређује ко ће бити председник испитне комисије, ко испитивач, а који ће члан водити записник о раду испитне комисије. Записник о раду испитног одбора води секретар кога именује директор.

Испитни одбор евидентира:

- теме за матурски рад;
- кандидате за матурски испит са подацима о једном од изабраних предмета и називу теме за матурски рад;
- рокове и распоред полагања појединих делова испита;
- наставнике који ће да дежурају за време писмених испита;
- наставнике – менторе које ће ученици консултовати у току израде матурског рада;
- утврђује теме и задатке за писмене испите;
- утврђује општу оцену на матурском испиту;
- утврђује коначну оцену у случају несагласности чланова испитне комисије приликом закључивања оцена за поједине предмете.

Испитни одбор усваја одлуке већином гласова присутних чланова, а може да одлучује ако су присутне две трећине свих чланова. Испитне комисије предлажу оцене из предмета и матурског рада.

ОЦЕЊИВАЊЕ И ОСЛОБАЂАЊЕ ПОЛАГАЊА ИСПИТА

Успех ученика из појединих предмета и матурског испита оцењује се једном оценом која се изводи на основу оцена добијених на писменом и усменом делу испита и матурском раду и одбрани матурског рада.

Општи успех на матурском испиту исказује се једном оценом као средња аритметичка вредност оцена добијених за поједине предмете који су полагани на матурском испиту и оцене из матурског рада.

Оцена из матурског рада изводи се на основу оцена добијених на матурском раду и одбрани тога рада.

Оцене појединих предмета утврђује испитна комисија на предлог предметног испитивача, а оцену општег успеха испитни одбор на основу извештаја испитних комисија. Ако испитна комисија не може да утврди појединачне оцене једногласно, ако је један оцењивач дао позитивну оцену, други негативну, или је разлика између позитивних оцена две и више, испитни одбор утврђује коначну оцену.

Ученик је положио матурски испит ако је из свих делова испита добио позитивну оцену.

Ученик који је на матурском испиту добио једну или две недовољне оцене полаже поправни испит из тих предмета. Уколико не положи поправни, поново полаже матурски испит у целини, као ванредан ученик, у роковима које одреди школа.

Ученик може бити и неоцењен или оцењен негативном оценом, без полагања испита. Неоцењен остаје ученик који прекине писмени испит из оправданих разлога и ученик који је због кршења испитних правила удаљен са испита.

Негативном оценом оцењује се ученик који прекине писмени испит без оправданих разлога, ученик који није предао писмени задатак, ученик који је напустио просторију у којој се полаже испит, без дозволе дежурног наставника. Негативном оценом оцењује се и ученик за кога се недвосмислено докаже да је у току испита или после испита користио недозвољена средства или да је рад преписао.

Ослобађају се усменог дела испита из предмета *програмирање, објектно оријентисано програмирање и базе података ученици* који су писмени испит положили са одличном оценом.

ЕВИДЕНЦИЈА И ЈАВНЕ ИСПРАВЕ

О току полагања писмених и усмених испита води се записник. За време дежурства на писменом испиту дежурни наставник уноси у записник све што није у складу са утврђеним правилима о току писменог испита. Записници се воде посебно о раду испитних комисија, а посебно о раду испитног одбора.

Записник о матурском испиту обухвата податке о ученику, податке о испитним предметима, члановима испитног одбора и испитних комисија, податке о темама, односно задацима, као и питања за предмете и успех за сваки део испита.