Врз основа на член 55 став 1 од Законот за организација и работа на органите на државната управа (,,Службен весник на Република Македонија” бр. 58/00, 44/02 и 82/08, 167/10 и 51/11) и врз основа на член 22 став 1 од Законот за средно образование (,,Службен весник на Република Македонија” бр. 44/95, 24/96, 34/96, 35/97, 82/99, 29/02, 40/03, 42/03, 67/04, 55/05, 113/05, 35/06, 30/07, 49/07, 81/08, 92/08, 33/10, 116/10, 156/10, 18/11, 51/11, 6/12, 100/2012 и 24/2013) министерот за образование и наука ја донесе Наставната програма по **програмски јазици** за трета година гимназиско образование.

НАСТАВНА ПРОГРАМА

МИНИСТЕРСТВО ЗА ОБРАЗОВАНИЕ И НАУКА
БИРО ЗА РАЗВОЈ НА ОБРАЗОВАНИЕТО

ПРОГРАМСКИ ЈАЗИЦИ

III ГОДИНА

ГИМНАЗИСКО ОБРАЗОВАНИЕ

Скопје, мај 2013

1. ВОВЕД

Наставата по предметот **програмски јазици** на учениците им овозможува проширување на знаењата и здобивање на вештини од областа на информатиката, како и можност за креативно користење на компјутерот во наставата и учењето.

Според наставниот план за гимназиско образование во природо - математичкото подрачје – комбинација Б, предметот програмски јазици во трета година има статус на изборен предмет и се изучува два часа неделно, односно 72 часа годишно.

Учениците со овој наставен предмет ќе развиваат способности за успешно користење на интегрирани околини за програмирање при изготвување на едноставни компјутерски програми што ќе им овозможи успешно да продолжат со понатамошно изучување на програмирањето. Оспособеноста на учениците за програмирање ќе им овозможи и олеснување во процесот на учење, како и нивно вклучување во процесот на новите предизвици на животот.

2. ОПШТИ ЦЕЛИ

Ученикот/ученичката:

- да усвојува поими, правила, постапки и процедури за изработка на алгоритми и програми со користење на даден програмски јазик;
- да се запознае со поимите, правилата, постапките и процедурите за примена на програмирањето и програмските јазици во компјутерската технологија;
- да се оспособи правилно да користи интегрирани околини за програмирање;
- да се оспособи да изработува програми со одреден програмски јазик;
- да се запознае со концептот аритметички операции, променливи и изрази во програмскиот јазик;
- да усвојува и применува дополнителни специфики на програмскиот јазик;
- да се оспособи за изработка на едноставни програми со структури за избор (две или повеќе можности) и повторување;
- да се оспособува за комбинирање различни техники при изработка на готови програми;
- да се оспособи за изготвување на програми со едукативна, креативна и/или забавна содржина;
- да усвојува поими, правила, постапки и процедури за изработка на проекти со програмирање;
- да усвојува поими, правила, постапки и процедури за изготвување на продукти со програмирање;
- да се оспособува да истражува и да ја оценува точноста, релевантноста, соодветноста и можностите на електронските информациски извори;
- да го проширува и продлабочува стекнатото информатичко знаење низ практично реализирање на програми во изучените програмски јазици;
- да се оспособи да ја планира и организира својата работа на конкретен алгоритамски проблем;
- да се оспособи да комбинира елементи во нова целина;
- да се оспособи да изготви целосен проект.

3. КОНКРЕТНИ ЦЕЛИ

Програмско подрачје 1: Програмирање и програмски јазици (5 часа)

Цели	*Содржини*	*Поими*	*Активности и методи*
Ученикот/ученичката: - да знае за концептите за програмирање и програмски јазик; - да се запознае со историјатот на програмските јазици; - да умее да направи дистинкција помеѓу програмски јазици на пониско и повисоко ниво (предности и недостатоци); - да се запознае со поделба на програмските јазици според намената; - да се запознае со поделба на програмските јазици според начинот на обработка на податоците; - да умее да направи јасна разлика помеѓу јазици со структурен/процедурален, објектно-ориентиран или функциски пристап; - да умее да ги наброи актуелните програмски јазици и истите да ги класификува според начинот на обработка на податоците.	Програмирање и програмски јазици Историјат на програмските јазици (машински, асембли-симболички, виши програмски јазици) Поделба на програмски јазици: - според намена; - според начин на обработка на податоци Актуелни програмски јазици	Програмирање Програмски јазик Машински јазици Асембли / симболички јазици Виши програмски јазици Императивни јазици Декларативни јазици Процедурални јазици Објектно – ориентирани јазици Функциски јазици Логички јазици FORTRAN Lisp, Prolog, Basic, Pascal Cobol, C, C++, Java C#, Python	- Повторување за концептите за програмирање и програмски јазик. - Дискусија за природни и вештачки јазици, програмирање, програмски јазик, примери на програмски јазици. - Презентација на историјатот на програмските јазици. - Презентација на различните поделби на програмските јазици. - Разгледување на готови пример програми во различни програмски јазици со цел утврдување на типот на јазикот. - Запознавање со актуелните програмски јазици. - Истражување на популарноста на одредени програмски јазици во минатото и сега.

Програмско подрачје 2: Претставување на податоци во компјутер. Општи типови на податоци (11 часа)

Цели	*Содржини*	*Поими*	*Активности и методи*
Ученикот/ученичката: - да ги осознае начините на	Општо претставување на	Броен систем	- Запознавање со начините на

претставување на броеви, знаци, слики и звук во компјутер; - да знае да објасни што е позиционен броен систем; - да разликува бинарен, октален, декаден и хексадекаден систем; - да умее да прави конверзија од еден во друг броен систем за конкретните бројни системи: бинарен <-> декаден и бинарен <-> октален <-> хексадекаден; - да знае да изврши операција собирање на два броја во бинарен броен систем - да објасни што е бит, бајт, збор и како се претставуваат целите броеви во компјутерот; - да умее да наведе неколку различни типови на податоци.	броеви, знаци, слики, звук Бројни системи Бинарен броен систем, декаден броен систем, октален и хексаденаден броен систем Конверзии помеѓу бројни системи (бинарен <-> декаден) (бинарен <-> октален <-> хексадекаден) Пример за операција со бинарни броеви – собирање Повторување за бит и бајт. Мерки за количество податоци Претставување на целите броеви во компјутер. Опсег на броеви кои може да се претставaт Општи типови на податоци	Бинарен броен систем, декаден броен систем, октален броен систем, хексадекаден броен систем Бајт Бит Конверзија Тип на податок	претставување на броеви, знаци, слики и звук во компјутер. - Дискусија за бројни системи, бинарен броен систем и презентација за конверзија на броеви од бинарен во декаден броен систем и обратно. - Презентација за специфична конверзија на броеви во следните бројни системи: бинарен <-> октален <-> хексадекаден. - Вежби за совладување на техниките за конверзија. - Презентирање вежби за совладување на операција собирање на бинарни броеви. - Повторување за бит и бајт. - Презентација на мерки за количество податоци. - Дискусија за примена на бинарниот броен систем и запознавање со еден начин на претставување на целите броеви во компјутер. - Претставување на некои општи типови на податоци.

Програмско подрачје 3: C++ и околина за развој (8 часа)

Цели	***Содржини***	***Поими***	***Активности и методи***
Ученикот/ученичката: - да го познава историјатот на C++ и да знае дека постојат повеќе видови („ароми“) на јазикот	Историјат на C++ и актуелна примена. Стандарден C++ наспроти C++/CLI	Преведувач Програмер Интегрирана околина за	- Презентација на историјатот на C++ и актуелната примена на јазикот. - Запознавање дека постојат

- да знае правилно да користи интегрирана околина за програмирање; - да знае да ја објасни структурата на едноставна програма во C++; - да ги познава основните елементи на програмскиот јазик; - да знае да дефинира и да користи типови на променливи: int, float, double, char, bool; - да се потсети и да ја знае синтаксата и функцијата на наредбите за влез, за излез и за доделување на вредности на променливи; - да се оспособи за изработка на едноставна програма со редоследна структура; - да се оспособи да дебагира едноставна програма.	Запознавање со основните елементи на интегрирана околина за програмирање Едноставна C++ програма (како функција main()) Основни елементи на програмскиот јазик (азбука на јазикот, градбени делови, коментари, величини, идентификатори, искази, клучни зборови, исказ за доделување) Податоци во C++ ; - променливи; - типови на податоци; целоброен, знаковен, реален, логички Читање и печатење на вредности на податоци Синтакса на наредбите за влез и за излез Изработка и извршување на едноставни програми и дебагирање	програмирање Изворна програма Извршна програма Коментар Величина Идентификатор Исказ (наредба) Клучен збор Оператор Аритметички операции Изрази Константа Променлива Тип на променлива int float double char bool Оператор за доделување Наредба за излез Наредба за влез Дебагирање	повеќе видови („ароми“) на јазикот преку пример за некои разлики помеѓу стандарден C++ наспроти C++/CLI. - Запознавање со основните елементи на интегрираната околина за програмирање која ќе се користи. - Повторување на процесот за креирање и преведување на програма. - Повторување за елементите на една програма напишана во C++ преку пример програма. - Дискусија за аритметичките операции, изрази, величини во C++ (константи и променливи). - Потсетување на основните типови на податоци во C++ (int, float,double, char, bool). - Изработка на програми со вклучување на доделувања, аритметички пресметки, читање на вредности од тастатура и прикази на екран. - Прикажување на функцијата на дебагерот.
Програмско подрачје 4: Класи и објекти во C++ (8 часа)			
Цели	***Содржини***	***Поими***	***Активности и методи***
Ученикот/ученичката - да се запознае со концептите на објектно ориентираното програмирање; - да умее да ги распознава	Вовед во објектно – ориентираното програмирање (ООП)	Објектно ориентирано програмирање Класа	- Презентација на концептите на објектно – ориентираното програмирање преку примери од реалноста и дискусија околу тоа

елементите на ООП (класи, објекти, функции-членки, податочни членки) и нивната функција; - да умее да разликува објект од класа; - да ја осознае синтаксата за креирање на класа; - да ја осознае формата и функцијата на функциите за поставување и преземање; - да ја знае улогата на конструкторот; - да се оспособи за примена на изучените концепти во конструкција на класи и изработка на програми; - да знае за можноста за одделување на интерфејс од имплементација.	Класи, објекти, функции-членки, податочни членки Дефинирање на класа со ф-ја членка Податочни членки, функции за поставување и преземање Конструктори Одделување на интерфејс од имплементација Изработка на програми кои содржат дефиниција на класа и примена на објекти – инстанци од таа класа	Објект Инстанца Функција-членка Податочна членка Функции-членки за поставување на вредности (set) Функции-членки за преземање на вредности (get) Конструктор Интерфејс Имплементација	- Дискусија околу елементите на ООП (класи, објекти, функции-членки, податочни членки) и нивната функција. - Презентирање на синтаксата за креирање на класа. - Презентација на формата и функцијата на функциите за поставување и преземање. - Презентација на улогата на конструкторот. - Изработка на пример задачи за примена на изучените концепти за конструкција на класи и изработка на програми. - Запознавање со техниката за одделување на интерфејс од имплементација.
Програмско подрачје 5: Структурни делови и код во ООП (6 часа)			
Цели	**Содржини**	**Поими**	**Активности и методи**
Ученикот/ученичката: - да умее да изработи програми со контролни структури за избор; - да умее да изработи програми со контролни структури за повторување; - да умее да изработи програми со наредбите *break* и *continue* во контролни структури за повторување;	Контролни структури за избор Контролни структури за повторување Примена на наредбите *break* и *continue* во структурите за повторување	Споредбен израз (услов) Логички израз Структура за избор од две можности Блок од искази (блок, сложен исказ) Структура за избор од повеќе можности Услов	- Практична изработка на програми со контролни структури за избор (минимум и максимум од два/три броја, подредување на три броја по неопаѓачки/нерастечки редослед). - Практична изработка на програми со контролни структури за повторување (пресметување на збир и производ, табелирање на вредности на функции,

- да умее да ги вклучи изучените структури во објектно ориентиран програмски код.		Логички оператори Циклус Структура за повторување на циклус до исполнување на услов Бројач Структура за повторување на циклус со броење на циклусите	испитување на својства на цели броеви – прост број, палиндром, нзд). - Практична изработка на програми со наредбите *break* и *continue* во контролни структури за повторување. - Изработка на програми со конструкција на класи и примена на објекти – инстанци на класите.
Програмско подрачје 6: Функции. Функции – членки на класа (10 часа)			
Цели	**Содржини**	**Поими**	**Активности и методи**
Ученикот/ученичката - да умее да го објасни поимот функција; - да се потсети за потребата и предностите од користење (повикување) на функции и за препознавање на проблемски ситуации во кои треба да се примени функција; - да ги запознае основните математички функции од библиотеката cmatx (fabs, sqrt, pow, ceil, floor, sin, cos); - да се оспособи за изработка на функции и нивно повикување од главната програма; - да се оспособи да решава проблеми со примена на функции; - да се оспособи за правилно пренесување на вредности преку	Потсетување за повикување на функција, предавање на вредности во функција преку параметри и за примена на функции Функции од математичка библиотека Функциски прототипови и принуда на аргументи Правила за делокруг Функции без параметри и функции со повеќе параметри Референци и референцни параметри Обременување на функции	Библиотека на функции Прототип на функција Параметар Аргумент Повик на функција Локална променлива Глобална променлива Референца Обременување на функција	- Презентирање на примери на функции и пример програми кои користат функции. - Дискусија за синтаксата на прототип на функција. - Дискусија за повикот на функција, повик на функција-членка и предавањето на параметри. - Изработка на програми во кои се повикуваат математички функции. - Изработка на функции (дел од нив нека се претходно изработени програми). - Изработка на програми кои искористуваат функции преку повик. - Дискусија за локални и глобални променливи.

параметар и преку референца; - да се запознае со правила за делокруг и да се оспособи да разликува глобални од локални променливи; - да се оспособи да препознава и реши проблем со обременување на функции; - да се оспособи да решава проблеми со примена на функции без параметри и функции со повеќе параметри; - да се оспособи да го примени знаењето за функции при изработка на функции – членки на класа.			- Дискусија за пренесување на вредност преку вредносен и референцен параметар. - Дискусија за правила за делокруг, локални и глобални променливи. - Изработка на функции без параметри (на пр. печатење на текст) и функции со повеќе параметри (на пр. замена на вредности на две променливи). - Дискусија за концептот за обременување на функции и негова примена. - Изработка на ОО програми со вклучување на изучените концепти.
Програмско подрачје 7. Низи и вектори (24 часа)			
7.1. Еднодимензионални низи (14 часа)			
Цели	**Содржини**	**Поими**	**Активности и методи**
Ученикот/ученичката: - да умее да ги објаснува и дефинира поимите: структура на податоци и еднодимензионална низа како структура од податоци; - да ја осознае потребата од користење на структура на податоци – еднодимензионална низа; - да знае да декларира, дефинира и користи еднодимензионална низа; - да умее да пристапува до член од еднодимензионална низа; - да знае да користи повторување со броење на циклусите при достапување до сите елементи	Структура на податоци – еднодимензионална низа (во С-стил), дефинирање, декларирање и иницијализација на еднодимензионална низа Пристапување и операции со членовите на еднодимензионална низа, поминување низ еднодимензионална низа Работа со повеќе низи Сортирање на низа по	Структура на податоци Еднодимензи-онална низа Член на низа Индекс на член на низа Поминување низ низа Сортирање на низа	-Дискусија за структури на податоци - Објаснување и дискусија за еднодимезионална низа како едноставен пример за структура на податоци - Дискусија за декларирање и дефинирање на еднодимезио-нална низа преку примери со програмски код. - Дискусија за индекс на низа. - Практична изработка на програми со еднодимензио-нални низи (основни операции со низи, пресметување на просечна вредност, наоѓање на

од низата(поминување низ низа); - да умее да изработува програми со користење на еднодимензионални низи; - да се запознае со алгоритми за сортирање на едносимензионална низа; - да знае да анализира временска сложеност на алгоритми за сортирање на еднодимензионална низа; - да се запознае со алгоритми за пребарување на еднодимензионална низа; - да знае да анализира временска сложеност на алгоритми за пребарување на еднодимензионална низа; - да умее да изработува програми со функции чии аргументи се еднодимензионални низи; - да умее да креира програми во кои се користат еднодимензионални низи од неколку објекти; - да умее да креира класи кои содржат еднодимензионални низи како податочни членки - да умее да го применува класниот шаблон <vector> од стандардната библиотека на C++ како замена за еднодимензионална низа во C-стил.	растечки и опаѓачки редослед (осврт на временска сложеност) Пребарување во низа (осврт на временска сложеност) Предавање низи на функции Креирање на еднодимензионални низи од неколку објекти Креирање на класи кои содржат еднодимензионални низи како податочни членки Класен шаблон <vector> од стандардната библиотека на C++	Пребарување во низа Временска сложеност Низа од објекти	минималната/максималната вредност во низа, сортирање на низа, пребарување во низа). - Практична изработка на програми со предавање на еднодимензионални низи на функции (на пр. функција која го враќа збирот на елементи на низа, читање/печатење на низа...). -Креирање на програми со повеќе низи кои решаваат ситуации од секојдневиот живот. - Изработка на програми кои вклучуваат еднодимензионални низи од објекти од дадена класа (пр. наоѓање просечна оценка за даден предмет за дадена низа од ученици...). - Изработка на програми кои вклучуваат изработка на класа која има еднодимензионални низи за податочна членка (пр. Класа ученик, податочна членка Низа оценки од различни предмети...). - Изработка на програми кои вклучуваат употреба на класен шаблон <vector> наместо стандардни еднодимензионални низи.

7.2. Текстуални низи и класата string(6 часа)

Цели	Содржини	Поими	Активности и методи
Ученикот/ученичката: - да осознае дека текстуална низа (низа од знаци) е вид на еднодимениоанлна низа - да ја познава синтаксата за декларација и дефинирање на текстуални низи; - да ја познава и користи логичката организација на низа од знаци; - да ги знае најчесто користените програмски функции за работа со низи од знаци; - да умее да напише алгоритми и програми за задачи со текстуални низи, со користење на изучените функции за работа со текстуални низи; - да умее да напише алгоритми и програми за задачи со текстуални низи како објекти од класата string.	Текстуална низа (низа од знаци) како еднодимензионална низа, декларирање/дефинирање на текстуална низа Функции за работа со текстуални низи Запознавање со класата string и можноста за примена при работа со низа од знаци	Текстуална низа Елементи на текстуална низа Индекс на член на текстуална низа Фукнции за текстуална низа Поминување низ текстуална низа Класата string	- Објаснување и дискусија за текстуална низа како вид на еднодимензионална низа. - Дискусија за декларирање и дефинирање на текстуална низа преку пример со програмски код. - Практична изработка на програми со текстуални низи (на пр. колку пати се појавува буква во реченица, замена на мали со големи букви, пребројување на зборови во реченица, проверка дали збор е палиндром, составување на реченица од дадени зборови...). - Практична изработка на програми со текстуални низи како објекти од класата string. -Креирање на програми кои решаваат ситуации од секојдневниот живот.

7.3. Дводимензионални низи – матрици (4 часа)

Цели	Содржини	Поими	Активности и методи
Ученикот/ученичката: - да дефинира и опишува дводимензионална низа – матрица како систем на организација на податоците;	Повеќедимензионални низи. Дводимензионални низи (матрици) како пример на повеќедимензионални низи.	Дводимензионалн а низа - матрица Член на матрица	- Дискусија за повеќедимензионални низи и матрица како пример на таква низа.

- да дефинира и опишува начин на пристапување до секој од елементите на матрица; - да разликува двоиндексна променлива, индекс, вредност на двоиндексна променлива; - да креира и практично да реализира внесување и печатење на матрици; - да разликува и да оперира со ред, колона и елемент од матрица; - да креира и практично да реализира програми за основните операции со матрици; - да креира и практично да реализира програми за едноставни проблеми врзани со матрици.	Дефинирање, декларирање и иницијализација на дводимензионални низи (матрици) Креирање и печатење на матрици, изминување низ матрица Операции со ред, колона и елементи од матрица	Индекс на член на матрица Ред Колони Дијагонала	- Презентација на примери од секојдневието каде што се вклучени матриците. - Дискусија за декларирање и дефинирање на матрици преку пример со програмски код. - Изработка на програми со едноставни читања/печатења на матрици. - Дискусија за индекси на член на матрица. - Дискусија за ред, колона, дијагонала на матрица. - Практична изработка на програми со матрици (собирање/множење на елементите, собирање на елементи од дијагоналата, пребројување на позитивни/негативни, парни/непарни... членови во матрица). - Креирање на програми кои решаваат ситуации од секојдневниот живот.

4. ДИДАКТИЧКИ ПРЕПОРАКИ

- Насоки за меѓупредметно поврзување (интегрирано планирање, односно целно и содржинско поврзување меѓу сродни предмети и подрачја):
  - мајчин јазик (синтакса, семантика и граматика);
  - математика (цели броеви, реални броеви, координатен систем, геометриски тела, формули, функции и работа со различни типови на податоци);
  - информатика (оперативни системи, програма за цртање, програма за уредување текст и програма за табеларни пресметки);
  - ликовно образование (креативно и визуелно осмислување кај некои задачи);
  - сите останати предмети чии содржини претставуваат содржини на изработување на проектни задачи.

Препораки за користење на годишниот фонд на часовите
За секое предвидено програмско подрачје предложен е одреден годишен фонд на часови, односно:

- Програмирање и програмски јазици - 5 часа;
- Претставување на податоци во компјутер. Општи типови на податоци – 11 часа;
- C++ и околина за развој – 8 часа;
- Класи и објекти во C++ - 8 часа;
- Структурни делови на код во Објектно ориентирано програмирање - 6 часа;
- Функции. Функции – членки на класа – 10 часа;
- Низи и вектори – 24 часа;
- Дводимензионални низи - 4 часа.

Од предвидениот вкупен фонд на часови по наставните програмски подрачја дозволени се мали отстапувања согласно знаењата на учениците, претходната подготовка на учениците за работа со компјутер, дополнителната опрема со која располага училиштето и слично. Во овој контекст, некои содржини ќе се повторуваат повеќе пати за сметка на други кои ќе се работат со помал фонд на часови.

Наставни средства:

- учебник за програмски јазици избран на ниво на училиштето, одобрен од министерот;
- литература: се препорачува користење на учебник од проектот на Владата ,, Превод на 500 стручни научни книги и учебници “, Пол Дејтел, Харви Дејтел: **C++: КАКО СЕ ПРОГРАМИРА**, , глава 2 – Вовед во C++ програмирање, глава 3 - Вовед во класи и објекти, глава 4 - Контролни наредби –прв дел, глава 5 - Контролни наредби - втор дел, глава 6-Функции и вовед во рекурзија, глава 7- Низи и вектори, Арс ламина, Скопје 2010
- Интернет, образовни софтвери;
- Интегрирана околина за програмирање и демо програми;
- списанија;
- аудио-визуелни средства.

5. ОЦЕНУВАЊЕ НА ПОСТИГАЊАТА НА УЧЕНИЦИТЕ

Во текот на наставата по информатика се препорачува формативно следење кое вклучува изработка и водење портфолио на учениците што опфаќа:

- собирање на показатели (ученички изработки на компјутер) за секој ученик посебно;
- тековни (формативни) однапред подготвени евалвациони листи за секој ученик посебно кои се изработуваат по конкретната негова активност.

На крајот на секое тримесечје, врз основа на сознанијата од формативното оценување се реализира микросумативното оценување.
Согласно природата на програмата по предметот програмски јазици оценувањето може да се реализира усно, практично, со презентација и слично.
Ученикот се оценува со бројчана оценка.

6. ПРОСТОРНИ УСЛОВИ ЗА РЕАЛИЗИРАЊЕ НА НАСТАВНАТА ПРОГРАМА

Програмата во однос на просторните услови се темели на Нормативот за простор, опрема и наставни средства за средно образование донесен од страна на министерот за образование и наука.

7. НОРМАТИВ ЗА НАСТАВЕН КАДАР

- Завршени студии по:
- информатика VII/1, наставна насока и применета насока со педагошко-психолошка и методска подготовка;
- математика-информатика VII /1, наставна насока;
- компјутерска техника, информатика и автоматика, VII /1 со педагошко-психолошка и методска подготовка;
- завршени дипломски четиригодишни студии од областа на информатиката, информациските или информациско - комуникациските технологии на акредитираните универзитети во РМ или во странство со домашна акредитација и со педагошко-психолошка и методска подготовка.

8. ОЧЕКУВАНИ РЕЗУЛТАТИ

Ученикот/ученичката:

- знае поими, правила, постапки и процедури за изработка на проекти и програми со користење на компјутер;
- умее да изготви продукти со компјутерски програми;
- знае да истражува и да ја оценува точноста, релевантноста, соодветноста и можностите на електронските информациските извори;
- умее да применува правила, постапки и процедури за решавање на одредени проблеми;
- ја планира и ја организира својата работа на конкретна задача со компјутер;
- комбинира различни техники за програм при реализација на одредена задача;
- стекнатото знаење за работа на компјутер може да го користи за креативно изготвување на продукти;
- истражува и пронаоѓа начини за изработка на текстови, цртежи, презентаци, тестови, готови продукти и слично со едукативна, контролна, креативна и/или забавна содржина со користење на корисничките програми;
- комбинира елементи во нова целина;
- изготвува целосен проект и готов продукт.

9. ПОЧЕТОК НА ПРИМЕНА НА НАСТАВНАТА ПРОГРАМА

Датум на започнување: 1.09.2013 година

10. Потпис и датум на утврдување на наставната програма

Наставната програма по **програмски јазици** за трета година на гимназиското образование, на предлог на Бирото за развој на образованието, ја утврди

Министер

Спиро Ристовски

РЕПУБЛИКА МАКЕДОНИЈА
МИНИСТЕРСТВО ЗА ОБРАЗОВАНИЕ И НАУКА
Бр.11-5181/1
20.09.2013 год.
СКОПЈЕ

Изготвил: работна група формирана со решение бр. 02-502/2 од 14.03.2013 година, координатор Гордана Јанакиевска, раководител на одделение
Контролирал: Трајче Георгиевски, раководител на одделение
Одобрил: м-р Митко Чешларов, раководител на сектор
Директор: м-р Весна Хорватовиќ,