

ГИМНАЗИЈА
РАЗРЕД: ДРУГИ
СМЈЕР: РАЧУНАРСКО-ИНФОРМАТИЧКИ
НАСТАВНИ ПРЕДМЕТ: ПРОГРАМИРАЊЕ

Годишњи број часова: 108

Седмични број часова: 3

СВРХА НАСТАВНОГ ПРЕДМЕТА

Оспособити ученике за развијање алгоритамског и логичког размишљања, самостално рјешавање проблема и моделовање процеса, уз ефикасну, рационалну и етичку примјену концепата програмирања у различитим областима.

ОПШТИ ЦИЉЕВИ ПРЕДМЕТА

- Развијање алгоритамског и логичког размишљања те способности структурирања и рјешавања сложенијих проблема;
- Оспособљавање ученика за анализу, моделовање и самостално формулисање рјешења;
- Подстицање тачности, прецизности, упорности и систематичности у раду;
- Примјена концепата програмирања и дигиталних компетенција у различитим областима;
- Развијање способности анализе и разлагања сложених проблема на подпроблеме;
- Развијање способности за тимски рад и презентовање рјешења на јасан и логички организован начин;
- Развијање логичког и критичког размишљања при рјешавању различитих типова задатака;
- Развијање критичког и аналитичког приступа према изазовима и проблемима у контексту информационих технологија.

ПОСЕБНИ ЦИЉЕВИ ПРЕДМЕТА

- Оспособљавање ученика за формулисање алгоритама и приказивање рјешења у псеудокоду или дијаграмима;
- Препознавање и избор адекватних типова података за представљање реалних информација;
- Подстицање креативности и иновативности кроз експериментисање са алгоритамским рјешењима;
- Оспособљавање ученика за процјену ефикасности и исправности алгоритама;
- Развијање свијести о етичким и одговорним аспектима употребе информационих технологија;
- Подстицање самосталности у истраживању и примјени концепата програмирања.

НАСТАВНЕ ТЕМЕ:

1. Алгоритми: основни појмови и принципи
2. Основни елементи програмског језика
3. Наредбе гранања и наредбе понављања
4. Функције и модуларност
5. Једнодимензионални и дводимензионални низови
6. Рад са улазним и излазним датотекама
7. Сложени алгоритми и примјена у реалним проблемима

Наставна тема: Алгоритми: основни појмови и принципи			
Исходи учења	Активности ученика	Међупредметно повезивање	
Ученик ће бити способан да: - објасни појам алгоритма и његову улогу у рјешавању проблема; - идентификује улазне и излазне податке и повезује их са алгоритамским рјешењем; - анализира алгоритамске кораке и идентификује логичке грешке у једноставним алгоритмима; - разликује начине представљања алгоритама (опис ријечима, псеудокод, дијаграм тока) и оцјени погодност сваког начина за конкретан проблем; - разликује типове алгоритама према структури (линијски, разгранати и циклични); - процијени јасноћу и читљивост алгоритма са аспекта разумијевања и примјене; - аргументује избор типа алгоритма за рјешавање конкретног проблема; - објасни значај коначности и јасноће корака у алгоритамском рјешавању проблема; - идентификује примјере неисправних или недовршених алгоритама; - креира и представи једноставне алгоритме за рјешавање датих проблема.	Ученик ће да: - анализира примјере алгоритама из свакодневног живота и одреди тип алгоритма; - тумачи псеудокодове и дијаграме тока; - дискутује о редослиједу извршавања алгоритамских корака и аргументовано објашњава могуће посљедице промјене редослиједа; - идентификује и исправља логичке грешке у једноставним алгоритмима представљеним псеудокодом или дијаграмом тока; - анализира примјере исправних и неисправних алгоритама у контексту коначности алгоритама; - самостално и у групи осмишљава једноставне алгоритме.	Предмет: Математика Физика Хемија	Тема: Све наставне теме Све наставне теме Киселине, безе, соли; Метали s, p, d блока; Метали и неметали

Наставна тема: Основни елементи програмског језика			
Исходи учења	Активности ученика	Међупредметно повезивање	
Ученик ће бити способан да: - разликује основне елементе програмског језика (идентификаторе, промјенљиве, константе, операторе, типове података, наредбе); - објасни улогу различитих типова података у програму; - повеже појам промјенљиве са меморијским простором и током извршавања програма; - разликује основне типове оператора: аритметичке, логичке и релационе; - дискутује о избору типова података и оператора за ефикасно рјешавање задатка; - саставља и процјењује изразе користећи комбинацију оператора и промјенљивих; - одреди резултат извршавања једноставних дијелова кода на основу датих израза; - анализира сложеније изразе и предвиђа резултате њиховог извршавања; - примјени основне елементе програмског језика и изразе у једноставним алгоритмима; - идентификује и исправља синтаксне и логичке грешке у једноставним линијским програмима; - процијени исправност и редослијед извршавања наредби у једноставним задацима.	Ученик ће да: - креира и користи промјенљиве и друге идентификаторе различитог типа у једноставним програмима; - саставља изразе који користе различите операторе; - користи наредбе израза у једноставним програмима; - представља и аргументовано објашњава избор типова података и оператора; - рјешава једноставне проблеме употребом линијских алгоритама и основних елемената програмског језика; - симулира извршавање програма корак по корак и прати промјене вриједности промјенљивих.	Предмет: Математика Физика Психологија Хемија	Тема: Све наставне теме Термодинамика, Флуиди, Електростатика, Електрична струја Конгитивни процеси Киселине и соли pH вриједност и пуфери

Наставна тема: Наредбе гранања и наредбе понављања			
Исходи учења	Активности ученика	Међупредметно повезивање	
Ученик ће бити способан да: - објасни појам услова, одлука и итерација у алгоритмима; - примијени наредбе гранања и вишеструког гранања за рјешавање различитих проблема; - примијени једноставне и сложене услове користећи логичке операторе; - примијени различите наредбе понављања за рјешавање датог проблема; - упореди различите врсте наредби понављања и изабере најпогоднију у односу на задати проблем; - анализира услове понављања и предвиђа број понављања; - објашњава појаву бесконачног понављања итеративне наредбе и предлаже начине превазилажења; - примијени угнијеждене наредбе понављања за рјешавање датог проблема; - комбинује гранање и понављања у алгоритмима за рјешавање комплекснијих задатака; - контролише ток извршавања наредби понављања прекидом понављања или прескакањем појединих корака у зависности од испуњености услова;	Ученик ће да: - анализира примјере алгоритама са условима и понављањима; - креира алгоритме користећи наредбе гранања и вишеструког гранања; - анализира примјере погрешно написаних услова и предлаже корекције; - креира алгоритме користећи различите наредбе понављања; - креира алгоритме користећи угнијеждене наредбе понављања; - рјешава задатке који комбинују више услова и понављања; - прати ток извршавања алгоритма корак по корак и биљежи промјене вриједности промјенљивих у свакој итерацији; - рјешава задатке у којима је потребно одредити број понављања без извршавања програма; - рјешава задатке у којима је потребно, у зависности од	Предмет: Математика Физика Хемија	Тема: Све наставне теме Термодинамика, Флуиди, Електростатика, Електрична струја Киселине и соли рН вриједност и пуфери

- анализира ток извршавања алгоритама са више грана и понављања и предвиђа могуће резултате; - процијени исправност и ефикасност алгоритама који садрже гранање и понављање; - процијени читљивост и логичку исправност алгоритама са више угнијеждених грана и понављања; - идентификује могуће грешке у сложеним условима и понављањима и предлаже исправке.	услова, прекинути даље извршавање понављања или прескочити поједине кораке алгорита; - прати корак по корак алгоритам са више грана и понављања радњи и биљежи промјене вриједности промјенљивих; - тестира и оцјењује исправност написаних алгоритама; - идентификује грешке и предлаже исправке у датом програму.		
---	--	--	--

Наставна тема: Функције и модуларност			
Исходи учења	Активности ученика	Међупредметно повезивање	
Ученик ће бити способан да: - објасни појам функције и употребу функција у програму; - анализира потребу за функцијама у сложенијим алгоритмима и идентификује дуплирање радњи које могу бити апстраховане у функције; - разликује улазне параметре и повратне вриједности функција;	Ученик ће да: - анализира примјере алгоритама у којима се понављају исти дијелови кода и уочава потребу за употребом функција; - пише једноставне функције уз јасно дефинисане улазне податке и резултат;	Предмет: Математика Физика Хемија	Тема: Све наставне теме Термодинамика, Флуиди, Електростатика, Електрична струја Киселине и соли pH вриједност и пуфери

- примијени модуларни приступ у рјешавању проблема разлажући сложенији проблем на мање, функционалне цјелине; - комбинује функције са наредбама гранања и понављања у алгоритмима; - објасни принцип рекурзивног позивања функција и примијени рекурзивне функције у рјешавању једноставних проблема; - процјењује ефекте модуларности на прегледност, исправност и поновљиву употребу алгоритма.	- рјешава задатке разлагањем сложенијих проблема на мање, логички повезане цјелине; - примјењује модуларни приступ при осмишљавању алгоритама који укључују гранање и понављање; - рјешава задатке примјеном рекурзивних функција; - упоређује два рјешења истог задатка - једно са функцијама, друго без функција и дискутује о предностима и недостацима модуларног приступа.		
--	--	--	--

Наставна тема: Једнодимензионални и дводимензионални низови			
Исходи учења	Активности ученика	Међупредметно повезивање	
Ученик ће бити способан да: - објасни појам низа као структуре за складиштење података; - разликује једнодимензионалне и дводимензионалне низове; - креира низове на различите начине у зависности од захтјева проблема; - примјени операције над низом као структуром података;	Ученик ће да: - анализира примјере коришћења низова у алгоритмима; - креира алгоритме за обраду података који се складиште у једнодимензионалним низовима (нпр. унос, испис, сума, производ, највећи, најмањи елемент и сл.);	Предмет: Математика Физика Хемија	Тема: Све наставне теме Термодинамика, Флуиди, Електростатика, Електрична струја Киселине и соли рН вриједност и пуфери

- користи низове у комбинацији са наредбама гранања, понављања и функцијама; - примјени низове за складиштење и обраду података из реалних примјера; - рјешава задатке који захтијевају обраду података који се чувају у низовним структурама; - анализира алгоритме који користе низове, процјењујући ефикасност проналажења елемента по задатом критеријуму; - упореди различите алгоритамске приступе обраде података који се чувају у низовима и процијени њихову ефикасност.	- користи угнијежене петље за обраду дводимензионалних низова; - обради реалне податке који су сачувани у низовима; - рјешава практичне задатке са једнодимензионалним и дводимензионалним низовима; - упоређује различите приступе проналажења елемента низа по задатом критеријуму, анализира ефикасност приступа и предложи побољшања.	Географија	Становништво и насеља
---	--	------------	-----------------------

Наставна тема: Рад са улазним и излазним датотекама			
Исходи учења	Активности ученика	Међупредметно повезивање	
Ученик ће бити способан да: - објасни појмове улазне и излазне датотеке и њихову улогу у складиштењу података; - разликује улазне и излазне датотеке; - описује основне операције рада са улазним и излазним датотекама (отварање, читање, писање, затварање); - примјењује модуларни приступ при раду са подацима у датотекама, користећи	Ученик ће да: - анализира примјере програма који користе улазне и/или излазне датотеке; - пише једноставне програме за читање података из улазне датотеке и упис података у излазну датотеку; - разлаже задатак на функције: једна функција за читање	Предмет: Сви наставни предмети	Тема: Све наставне теме када се организују и складиште подаци са циљем њиховог чувања и обраде.

функције за читање, обраду и писање података; - рјешава задатке у којима се подаци читају из улазних датотека, обрађују и резултати се уписују у излазне датотеке; - примјењује одговорно и етичко поступање при читању, обради и чувању података.	података, друга за обраду, трећа за упис резултата; - комбинује рад са улазно/излазним датотекама са наредбама гранања и понављања; - рјешава практичне задатке из реалног контекста за које су подаци сачувани у датотекама.		
--	---	--	--

Наставна тема: Сложени алгоритми и примјена у реалним проблемима			
Исходи учења	Активности ученика	Међупредметно повезивање	
Ученик ће бити способан да: - анализира сложеније проблеме из домена елементарног програмирања и разлаже их на мање цјелине; - комбинује различите програмске конструкције у рјешавању задатака; - примијени алгоритамско размишљање у новим и непознатим ситуацијама; - развија и тестира програмска рјешења за комплексније задатке; - процијени ефикасност и исправност алгоритамских рјешења;	Ученик ће да: - рјешава задатке који обједињују употребу наредби гранања, понављања, функција, низова и датотека; - осмишљава алгоритамска рјешења за реалне проблеме из домена елементарног програмирања; - анализира и унапређује постојећа програмска рјешења; - представља и аргументује избор алгоритама за рјешавање задатака;	Предмет: Предузетништво Математика Физика Хемија	Тема: Истраживање тржишта Бизни план Тимски рад и лидерство Све наставне теме Термодинамика, Флуиди, Електростатика, Електрична струја Киселине и соли

- анализира сложене алгоритме, идентификује могуће грешке и предлаже побољшања; - примијени алгоритамско размишљање у малим пројектним задацима и групним активностима.	- анализира дати сложени алгоритам, идентификује могуће грешке или непотребне кораке, предлаже побољшања, имплементира измјене и тестира побољшано рјешење; - самостално и у групи рјешава сложеније програмске задатке у оквиру мини-пројеката.	Географија	pH вриједност и пуфери Становништво и насеља
--	---	------------	--

Дидактичко-методичка упутства и препоруке

- Наставни процес треба организовати као комбинацију кратког теоријског увода, вођених вјежби и самосталног рада ученика на рачунару. Сложеност задатака потребно је постепено повећавати, водећи рачуна о предзнању ученика и специфичностима смјера. Акценат наставе не смије бити искључиво на синтакси програмског језика, већ на развоју алгоритамског начина размишљања и разумијевању логике рјешавања проблема.
- У оквиру наставне теме *Алгоритми - основни појмови и принципи* умјесто теоријских дефиниција, настава треба да буде усмјерена на искуствено учење. Наставник треба да моделује активности тако да ученици прво анализирају алгоритме из свакодневног живота (нпр. рецепт за кување, упутство за склапање уређаја, упутство за употребу лијека и сл). Овај приступ служи као основа за касније објашњавање појма алгоритма и његове улоге у рјешавању комплексних проблема. Препоручује се увођење методе декомпозиције (рашчлањивања), те прије писања самог алгоритма, ученици треба да идентификују главне фазе сложеног проблема, што им омогућава лакши прелазак са опште идеје на конкретне алгоритамске кораке.

Приликом рјешавања задатака, наставник треба да инсистира на томе да ученици идентификују улазне и излазне податке прије него што крену у писање корака. То помаже у успостављању јасне везе између проблема и алгоритамског рјешења.

Са циљем разумијевања представљања алгоритма на различите начине предлаже се организација вјежбе гдје исти проблем ученици представљају на три начина: описом ријечима, псеудокодом и дијаграмом тока. Након тога могуће је покрените дискусију како би ученици оцијенили погодност сваког начина за конкретан задатак (нпр. дијаграм тока је често јаснији за гранања, док је псеудокод ближи програмирању). Са циљем развијања апстракције и способности логичког праћења тока информација ученици треба да вјежбају „читање“ туђих псеудокодова и дијаграма тока. Да би се псеудокод заиста приближио програмирању, препоручује се коришћење стандардизованих кључних ријечи (нпр. „АКО“, „ДОК“, „ПРИКАЖИ“), чиме се смањује двосмисленост природног језика.

Анализа логике и грешака у алгоритмима је веома важна за развој критичког мишљења. Наставник треба да подстиче ученике да дискутују о редослиједу извршавања корака (нпр. могуће је поставити питање: „Шта би се догодило ако бисмо замијенили корак 2 и корак 3“). Са циљем развијања способности откривања грешака могуће је користити примјере неисправних или недовршених алгоритама, при чему ученици имају задатак да идентификују логичке грешке и предложе исправке. Посебну пажњу треба посветити анализи примјера гдје алгоритам можда нема крај (бесконачне петље) или су кораци нејасни, како би ученици схватили значај коначности и јасноће за успјешно рјешавање проблема. Поред оцјене погодности начина представљања, ученици треба да пореде два различита, а тачна алгоритма за исти проблем, аргументујући који је ефикаснији у смислу мањег броја корака или бржег доласка до рјешења.

Ученици треба да овладају типологијом алгоритама кроз практичне примјере. То је разлог због којег наставник треба да припреми примјере за сваки тип: линијски, разгранати и циклични. Кроз анализу примјера из живота, ученици треба сами да одреде којем

типу алгоритама припада (нпр. рецепт за колаче је углавном линијски; одлука да ли носити кишобран је разгранати; мијешање зимнице док не прокључа је циклични алгоритам).

Процес креирања алгоритама не треба да буде само индивидуална активност. Ученици треба да самостално и у групи осмишљавају алгоритме. Групни рад је посебно користан за фазу представљања једноставних алгоритама, гдје група мора да објасни своју логику остатку одјељења. Када ученици креирају и представе своје алгоритме, остали ученици могу преузети улогу „извршиоца“ (тј. рачунара) како би провјерили да ли су кораци заиста јасни и воде ли до исправног резултата. Као алат за прецизно „читање“ алгоритама, препоручује се коришћење табела праћења у које ученици уписују промјене вриједности података кроз сваки корак, чиме се објективно идентификују логичке грешке. Ученике треба подстицати да тестирају своје алгоритме са необичним или екстремним подацима (нпр. „шта ако је унос нула“) како би осигурали потпуну предвидивост и јасноћу алгоритама. Током свих активности, наставник треба континуирано да инсистира на јасноћи и прецизности, јер су то темељи алгоритамског размишљања. Оваквим приступом поучавању и организацији наставе наставник код ученика развија дубље разумијевање логичких процеса који су темељ рачунарског размишљања.

Наставну тему *Алгоритми: основни појмови и принципи* могуће је повезати са свим наставним темама у наставном предмету Математика на начин да се алгоритамски представљају различити поступци израчунавања.

На исти начин могуће је остварити повезивање са свим наставним темама у оквиру наставног предмета Физика, кроз алгоритамско представљање поступака рјешавања задатака, анализа појава и експерименталних процедура.

Такође, могуће је остварити повезивање са наставним темама *Киселине, базе, соли; Метали s-, p- и d-блока и Метали и неметали* тако што ће ученици алгоритамски представљати поступке добијања супстанци, анализе њихових својстава, као и поједине индустријске процесе (нпр. добијање алуминијума и гвожђа).

- У оквиру наставне теме *Основни елементи програмског језика* наставни процес треба започети јасним разграничењем основних градивних блокова кода. Потребно је оспособити ученике да самостално разликују основне елементе програмског језика, укључујући идентификаторе, промјенљиве, константе, операторе, типове података и наредбе. Наставник треба да објасни улогу различитих типова података тако што ће их директно повезати са меморијским простором и начином на који се програм извршава. Важно је нагласити да избор типа података одређује колико простора заузима промјенљива и које операције су дозвољене над том промјенљивом. Наставник треба да подстиче ученике да дискутују и аргументовано објашњавају свој избор типова података и оператора. Активности треба конципирати тако да ученици не бирају типове само на основу функционалности, већ и на основу ефикасности рјешавања задатка, разматрајући однос између прецизности података и заузећа меморије. Ученици морају јасно разликовати аритметичке, логичке и релационе операторе. Методички је корисно давати задатке гдје се исти подаци обрађују различитим типовима оператора како би се видјела разлика у резултату. Активности треба усмјерити на то да ученици самостално састављају и процјењују вриједности израза користећи комбинације различитих оператора и промјенљивих. Посебну пажњу треба посветити анализи сложенијих израза. Препоручује се декомпозиција израза на мање функционалне цјелине

како би ученици могли прецизно предвидјети резултате извршавања прије самог покретања кода, узимајући у обзир приоритете оператора. На овај начин развија се способност предикције резултата прије самог покретања кода.

Да би ученици разумјели динамику програма, неопходно је увести методу „ручног проласка кроз код” тј. корак по корак. Потребно је дакле да ученици симулирају извршавање програма корак по корак, уз обавезно праћење промјена вриједности промјенљивих у сваком кораку. Ове симулације доприносе оспособљавању ученика да одреде резултат извршавања дијелова кода и процијене исправност редослиједа наредби у истом. Коначни циљ је примјена знања у реалним ситуацијама.

При обради различитих алгоритамских структура наставник треба да поставља проблемске задатке који природно захтијевају примјену алгоритамских структура, уз употребу различитих елемената програмског језика. На тај начин ученици кроз рјешавање конкретних задатака усвајају начин креирања и коришћења промјенљивих различитих типова унутар програма. Ученици треба да самостално рјешавају једноставне проблеме употребом линијских алгоритама. Препоручује се коришћење реалних сценарија гдје је неопходно примијенити исправан редослијед наредби како би се дошло до тачног рјешења, чиме се знање о основним елементима језика контекстуализује.

Посебан фокус треба ставити на развијање вјештине ученика да сами идентификују и исправљају синтаксне и логичке грешке. Препоручено је да наставник прикаже код са скривеном грешком (нпр. погрешан тип податка или лош редослијед наредби) и води ученике кроз процес њеног проналажења. Поред наставничких примјера, корисна активност је и међусобна идентификација грешака међу ученицима (вршњачка провјера). Овакав приступ подстиче ученике да критички процјењују туђи код, провјеравају исправност редослиједа наредби и заједнички долазе до оптималних исправки.

Активности треба усмјерити тако да ученик, умјесто простог преписивања кода, развија разумијевање улоге свих елемената у програму. Треба да научи користити изразе као функционалне цјелине у програму, те самостално примјењивати основне елементе и изразе у једноставним задацима.

Наставну тему *Основни елементи програмског језика* могуће је повезати са наставном темом *Конгитивни процеси* у наставном предмету Психологија, тако што ће ученици анализирати улогу пажње, памћења и логичког закључивања у праћењу извршавања алгоритама и програма.

Повезивање је могуће и са свим наставним темама у оквиру наставног предмета Математика, при чему ће ученици рјешавати једноставне математичке проблеме примјеном линијских алгоритама и основних елемената програмског језика.

На сличан начин могуће је остварити интеграцију са наставним предметом Физика и то наставним темама: *Термодинамика*, *Флуиди*, *Електростатика* и *Електрична струја*. Ученици ће рјешавати једноставне задатке из поменутих наставних тема користећи једноставне програмске структуре, кроз израду програма за израчунавање физичких величина и примјену одговарајућих формула.

Повезивање је могуће остварити и са наставним предметом Хемија, наставним темама *Киселине и соли*, те *pH вриједност и пуфери*, при чему ће ученици писати једноставне програме за израчунавање pH вриједности раствора и концентрације јона у киселинама и базама.

Препоручено је у договору са наставницима Математике, Физике и Хемије припремити задатке за ученике.

- У оквиру наставне теме *Наредбе гранања и наредбе понављања* настава треба да омогући ученицима да вјежбају примјену једноставних и сложених услова користећи логичке операторе како би прецизно дефинисали путање извршавања. Такође, потребно је организовати активности у којима се креирају алгоритми који не нуде само два пута (да/не), већ захтијевају вишеструко гранање при рјешавању сложенијих проблема. Неопходно је започети са јасним теоријским објашњењем појмова услова, одлука и итерација у алгоритмима како би ученици разумијели апстрактну основу наредби које ће примјењивати.

Са циљем развијања критичког односа према логици програма ученици треба да анализирају примјере погрешно написаних услова и самостално предлажу корекције. Овај процес треба проширити и на идентификацију грешака у сложеним условима и специфичним сценаријима понављања, подстичући ученике да аргументовано предлажу исправке.

Код наредби понављања умјесто механичког понављања, акценат треба ставити на стратешки избор и предвиђање понашања петљи. Потребно је оспособити ученике да упоређују различите врсте наредби понављања и бирају ону која је најпогоднија за конкретан проблем. Препоручује се рјешавање задатака у којима се број понављања одређује без покретања програма, чиме се развија способност логичког разматрања, анализе и моделовања алгоритма. Ученицима треба објаснити зашто долази до бесконачног понављања и тражити од њих да предложе конкретне начине за превазилажење таквих ситуација.

Наставник треба да омогући ученицима рјешавање проблема који имају више димензија или слојева креирањем алгоритама са угнијежденим наредбама понављања. Ученици треба да науче како да прекину извршавање или прескоче кораке унутар петље (користећи концепте попут `break` или `continue`) у зависности од испуњености специфичних услова. У овој фази је веома важно да ученици процијене не само логичку исправност, већ и читљивост алгоритама који садрже више угнијеждених грана и понављања, како би се од почетка развијале добре програмерске праксе. Препоручено је користити задатке који комбинују гранање и понављања, јер то најбоље осликава реалне програмерске изазове.

И у овој наставној теми препоручено је примјењивати провјеру исправности кроз праћење вриједности промјенљивих тј. праћење тока алгорита корак по корак уз биљежење промјена вриједности промјенљивих у свакој итерацији. Ово ће помоћи демистификацији рада процесора. Током ових вјежби, ученици треба да анализирају ток извршавања чак и код комплексних алгоритама са више грана и понављања како би са сигурношћу могли предвидјети коначне резултате.

Ученици дакле не треба само да направе функционалан алгоритам, већ и да процијене његову исправност и ефикасност. Они треба самостално да тестирају и оцјењују исправност својих рјешења, што заокружује циклус од теоријског концепта до функционалног производа. Овакав приступ осигурава да ученици не само користе наредбе, већ дубински разумију алгоритамску логику и знају како да је примијене на оптималан начин.

Наставну тему *Наредбе гранања и наредбе понављања* могуће је повезати са свим наставним темама у наставном предмету Математика кроз рјешавање задатака који захтијевају анализу услова, разматрање различитих случајева и понављање одређених поступака.

Повезивање је могуће остварити и са наставним темама *Термодинамика, Флуиди, Електростатика и Електрична струја* у наставном предмету Физика, при чему ученици могу примјењивати наредбе гранања и понављања приликом обраде резултата мјерења, анализе физичких величина и рјешавања нумеричких проблема.

У наставном предмету Хемија повезивање је могуће остварити са наставним темама *Киселине и соли и рН вриједност и пуфери* кроз рјешавање задатака који укључују класификацију супстанци на основу њихових својстава, као и обраду већег броја података примјеном наредби понављања.

Препоручује се сарадња са наставницима наведених наставних предмета приликом припреме задатака за ученике.

- У оквиру наставне теме *Функције и модуларност* ученицима је потребно објаснити кључне аспекте, који обухватају логичку организацију кода, интеракцију података и специфичне методе рјешавања проблема.

Ученици треба да схвате да је основна сврха функција ефикаснија организација кода. Препоручује се да они анализирају примјере алгоритама гдје се одређени дијелови кода понављају, што указује на директну потребу за коришћењем функција како би се избјегла редундантност. Потребно је оспособити ученике да идентификују конкретне радње које се могу апстраховати у функције током анализе сложенијих алгоритама. Неопходно им је објаснити да писање једноставних функција захтијева прецизно одређивање улазних података (параметара) и очекиваног резултата (повратне вриједности). Такође, важно је да јасно разликују шта функција прима као улаз, а шта враћа као излаз након извршења. Акценат треба ставити на писање једноставних функција уз строго дефинисане улазне податке и резултате како би се осигурала исправност.

Ученици треба да разумију да модуларност није само техничка вјештина, већ начин размишљања, која подразумијева разлагање сложених проблема на мање, функционалне и логички повезане цјелине. Потребно је да ученици увиде да ће на овај начин лакше управљати комплексним задацима и добити прегледнији алгоритам. При том функције не треба да посматрају изоловано, већ као градивне блокове сложенијих алгоритама. Модуларни приступ треба директно примијенити приликом самосталног осмишљавања алгоритама. Важно ја да науче да комбинују функције са наредбама гранања и наредбама понављања.

Посебну пажњу треба обратити на рекурзију. Осим објашњења принципа, ученике треба оспособити за самостално препознавање рекурзивних проблема и њихово рјешавање кроз практичну примјену рекурзивних функција. Ученици треба да разумију начин на који функција може позвати саму себе у сврху рјешавања проблема. Као завршну активност, препоручује се упоређивање два рјешења истог задатка (једног са функцијама и другог без њих) уз дискусију о предностима и недостацима оваквог приступа.

Наставну тему *Функције и модуларност* могуће је повезати са наставним темама у предмету Математика кроз анализу зависности и рјешавање проблема који се могу посматрати као скуп повезаних, понављајућих поступака.

У наставном предмету Физика повезивање је могуће са темама *Термодинамика, Флуиди, Електростатика и Електрична струја* кроз моделовање физичких појава у којима се јављају исте или сличне рачунске процедуре, чиме се уочава потреба за њиховим издвајањем у засебне функционалне цјелине.

У наставном предмету Хемија повезивање је могуће са темама *Киселине и соли, те рН вриједност и пуфери* кроз рјешавање задатака који укључују понављајуће рачунске поступке и анализу зависности између величина, што омогућава увођење модуларног приступа у рјешавању проблема.

- У оквиру наставне теме *Једнодимензионални и дводимензионални низови* наставни процес треба започети јасним дефинисањем појма низа као специфичне структуре за складиштење података. Препоручује се коришћење визуелних приказа како би ученици лакше уочили разлику између једнодимензионалних и дводимензионалних низова. Увести шематске приказе меморијских локација и индекса. Користити аналогije из стварног свијета (нпр. ормарићи у свлачионици за 1D низове, или распоред сједења у учионици за 2D низове) како би се објаснила разлика у димензијама. Методички је ефикасно прво анализирати већ постојеће примјере кориштења низова у алгоритмима прије него што ученици пређу на самостално креирање. Активности треба усмјерити на изградњу логике за обраду података. Ученицима треба задати два различита начина рјешавања истог проблема (нпр. проналажење елемента кроз цијели низ у односу на прекид претраге када се елемент нађе). Наставник треба да подстакне дискусију о томе који је приступ ефикаснији. Кључно је да ученици савладају основне алгоритме над једнодимензионалним низовима, (нпр. унос и испис елемената низа; израчунавање статистичких вриједности попут суме и производа; проналажење специфичних вриједности као што су највећи и најмањи елемент и сл). Да би се успјешно примјењивали низови у сложенијим структурама, наставник мора креирати услове за интегрисање низова са: наредбама гранања и понављања (нпр. користити их за претраживање и филтрирање података); угнијежденим петљама (са циљем оспособљавања за манипулацију дводимензионалним низовима); функцијама (потребно је подстицати ученике да модуларизују свој код користећи функције за специфичне операције над низовима). Наставник треба да фаворизује практичне задатке који су повезани са стварним свијетом. С тим у вези ученици треба да: обрађују реалне податке сачуване у низовима; примјењују низове за складиштење и обраду информација из реалних примјера; рјешавају задатке који захтијевају креирање низова на различите начине, прилагођавајући их специфичним захтјевима проблема укључујући и генерисање случајних вриједности помоћу одговарајућих функција (нпр. функције `rand()` за попуњавање низова случајним вриједностима у одређеном опсегу (нпр. симулација температура током мјесеца)). Препоручује се увођење провјере унесених података (нпр. ако се уноси оцјена у низ, програм не смије прихватити број већи од 5 или мањи од 1), чиме се вјежба комбинација низова са наредбама гранања. Успјешност наставе се огледа у способности ученика да самостално рјешавају задатке који комбинују све претходно наведене елементе. Препоручује се формативно праћење израде алгоритама и практичног рада на рачунару. Фокус не смије бити само на синтакси програмирања, већ на алгоритамској логици која омогућава ученику да одабере одговарајући начин организације података за конкретан проблем. Током практичног рада на рачунару, ученици треба да самостално тестирају своје програме на „граничним случајевима“ (нпр. шта се дешава ако је низ празан или ако су сви елементи у низу исти).

Наставну тему *Једнодимензионални и дводимензионални низови* могуће је повезати са наставним темама у предмету Математика кроз анализу и обраду података представљених у облику низова и табела.

У наставном предмету Физика повезивање је могуће са темама *Термодинамика, Флуиди, Електростатика и Електрична струја* кроз обраду експериментално прикупљених података који се могу организовати у једнодимензионалне и дводимензионалне структуре ради лакше анализе и поређења физичких величина.

У наставном предмету Хемија повезивање је могуће са темама *Киселине и соли и рН вриједност и пуфери* кроз систематизацију резултата мјерења и израчунавања који се могу представити у облику низова ради уочавања правилности и односа између величина.

У наставном предмету Географија повезивање је могуће са темом *Становништво и насеља* кроз анализу демографских и просторних података представљених у табеларном облику, при чему се једнодимензионални низови користе за приказ појединачних серија података, а дводимензионални за упоредну анализу више категорија.

- Препоручује се да се реализација наставе теме *Рад са улазним и излазним датотекама* започне јасним дефинисањем улога улазних и излазних датотека у контексту трајног складиштења података. Наставник треба да користи методу демонстрације на већ написаним примјерима како би ученици могли да анализирају програме који користе ове датотеке прије него што сами почну да програмирају. Важно је инсистирати на разликовању смјера протока података (читање наспрам писања) како би ученици правилно идентификовали тип датотеке с којом раде.

Приликом обраде основних операција (отварање, читање, писање, затварање), посебан методички акценат треба ставити на обавезно затварање датотека након употребе. Ово није само технички корак, већ дио одговорног и етичког поступања са подацима, чиме се спречава губитак или оштећење информација. Ученике треба подстицати да сваку операцију опишу и аргументују прије саме имплементације.

Неопходно је развијати једну од кључних вјештина код ученика, а то је способност да примијене модуларни приступ. Методички је најефикасније захтијевати од ученика да сваки задатак разлажу на најмање три специфичне функције: функцију за читање података из улазне датотеке, функцију за обраду података (логика програма) и функцију за упис резултата у излазну датотеку. Оваква структура олакшава проналажење грешака и подстиче ученике да размишљају о програму као о скупу независних, али повезаних компоненти.

Рад са датотекама не треба изоловати, већ га треба активно комбиновати са наредбама гранања и понављања. Да би учење било сврсисходно, препоручује се задавање практичних задатака из реалног контекста. Умјесто апстрактних примјера, ученици треба да раде са подацима који симулирају стварне ситуације (нпр. спискови ученика, временски подаци, рачуни), гдје се резултати обраде обавезно чувају у излазним датотекама за каснију употребу.

Кроз све активности, наставник треба да провлачи тему етичког поступања при обради и чувању података. Ученицима треба нагласити значај самосталног рада и етичког приступа програмирању, укључујући правилно навођење извора и критичко

коришћење постојећих рјешења. Ово укључује дискусију о приватности података који се читају из датотека и важности интегритета података који се уписују, чиме се ученици припремају за професионални рад у ИТ сектору.

Наставну тему *Рад са улазним и излазним датотекама* је могуће повезати са свим наставним предметима и свим наставним темама у којима се организују и складиште подаци са циљем њиховог чувања и обраде.

- У оквиру наставне теме *Сложени алгоритми и примјена у реалним проблемима* наставу треба конципирати тако да ученици савладају вјештину анализе сложених проблема прије самог писања кода. Препоручено је користити задатке који немају очигледно рјешење, подстичући ученике да примијене алгоритамско размишљање у новим и непознатим ситуацијама. Наставник треба да дјелује као фасилитатор који поставља потпитања, помажући ученицима да сами дођу до структуре рјешења.

Умјесто изолованог вјежбања наредби, акценат треба ставити на задатке који обједињују различите претходно усвојене елементе: наредбе гранања, циклусе, функције, низове и датотеке. Посебан акценат треба ставити на рад са датотекама како би ученици оспособили програме да чувају и користе податке изван тренутне сесије рада, што је кључно за реалне апликације за управљање подацима. Кроз вјежбе треба показати како се ове конструкције међусобно комбинују ради ефикаснијег рјешавања задатака.

Потребно је подстицати ученика да примјењују знања у малим пројектним задацима и групним активностима, па настава треба да укључи елементе тимског рада. Ученицима треба омогућити да у групи рјешавају реалне проблеме из домена елементарног програмирања. Ово подстиче подјелу улога (нпр. један ученик дизајнира алгоритам, други пише код, трећи тестира) и доприноси развоју социјалних вјештина неопходних у ИТ сектору. При том водити рачуна да није довољно само израдити програм који „ради“. Ученике треба оспособити да представе и аргументују свој избор. Са тим циљем потребно је организовати кратке презентације или „код-прегледе“ (code review) гдје ученици образлажу зашто су одабрали баш ту структуру података или одређени тип петље. Ово јача разумијевање ефикасности и логике програмског рјешења.

Наставник треба да прилагоди сложеност задатака различитим нивоима знања ученика. За ученике који брже напредују препоручује се проширивање задатака додатним условима, оптимизацијом алгоритма или анализом ефикасности, док се ученицима којима је потребна подршка обезбјеђују вођени примјери и поступна разрада рјешења. Како би се ученици оспособили за израду сложених алгоритама и идентификовање грешака, настава треба да укључи вјежбе „инспекције кода“. Ученицима треба понудити већ написане, сложене алгоритме који садрже логичке грешке или неефикасне, „непотребне кораке“ које они морају идентификовати, предложити побољшања и тестирати нову верзију. Овај процес помаже ученицима да развију критички приступ и способност процјене исправности туђих, али и властитих алгоритамских рјешења.

Процес учења треба да обухвати не само писање, већ и развој и тестирање програмских рјешења за комплексније задатке. Са тим у вези препоручено је увести праксу креирања тестних примјера. Ученике треба оспособити да предвиде граничне случајеве и да систематски провјеравају исправност свог кода, што је кључна компонента напредног алгоритамског размишљања. Током овог процеса, наставник треба подстицати ученике да експлицитно упоређују различите приступе истом проблему и процијене њихову ефикасност, не само кроз тачност већ и кроз једноставност и брзину извршавања алгоритамских корака.

Потребно је развијати код ученика одговоран однос према програмирању, укључујући самосталан рад, критичко коришћење доступних извора и избјегавање некритичког преузимања готових рјешења. Ученици треба да разумију значај ауторства и правилног навођења извора приликом коришћења туђег кода.

Наставну тему *Сложени алгоритми и примјена у реалним проблемима* могуће је повезати са наставним предметом Предузетништво кроз израду једноставних пројектних задатака који симулирају истраживање тржишта, планирање пословања и организацију тимског рада, при чему ученици развијају алгоритамска рјешења која обухватају више међусобно повезаних корака и услова.

У наставном предмету Математика повезивање је могуће остварити са свим наставним темама, кроз рјешавање сложенијих задатака који укључују анализу датих података, више корака израчунавања и примјену различитих алгоритамских структура у моделовању математичких проблема.

У наставном предмету Физика повезивање је могуће са темама *Термодинамика, Флуиди, Електростатика и Електрична струја* кроз алгоритамско моделовање сложенијих експерименталних ситуација и анализу више повезаних физичких величина.

У наставном предмету Хемија повезивање је могуће са темама *Киселине и соли и рН вриједност и пуфери* кроз анализу сложенијих хемијских система и поступака који се могу описати алгоритмима у више корака.

У наставном предмету Географија повезивање је могуће са темом *Становништво и насеља* кроз обраду и анализу сложених демографских података и њихову интерпретацију помоћу алгоритамских поступака.

Да би се постигла максимална ангажованост, алгоритамска рјешења треба осмишљавати за проблеме из стварног живота. То могу бити једноставне апликације за управљање подацима, симулације природних процеса или алати за аутоматизацију свакодневних задатака.

- Да би се постигла максимална ангажованост, алгоритамска рјешења треба осмишљавати за проблеме из стварног живота. То могу бити једноставне апликације за управљање подацима, симулације природних процеса или алати за аутоматизацију свакодневних задатака.
- За надарене ученике могуће је проширити исходе који се односе на сложеније алгоритамске задатке, продубљену анализу алгоритамских рјешења и елементе такмичарског програмирања, у складу са могућностима и интересовањима ученика.
- Наставник, у складу са расположивим фондом часова и предзнањем ученика, може прилагодити дубину остваривања појединих исхода учења, водећи рачуна да се очува суштина и развој алгоритамског размишљања.

Препоручује се примјена формативног праћења током процеса учења, кроз кратке провјере разумијевања, анализу ученичких алгоритама, вршњачку процјену и самопроцјену. Фокус треба да буде на процесу рјешавања проблема, а не искључиво на коначном резултату.

- Оцјењивање се врши у складу са Законом о средњем образовању и васпитању и Правилником о оцјењивању ученика у настави и полагању испита у средњој школи.
- Користити удбенике одobreне од стране Министарства просвјете и културе Републике Српске.