

НАСТАВНИ ПРОГРАМ ЗА ПРЕДМЕТ: РАЧУНАРСТВО И ИНФОРМАТИКА
СМЈЕР: СПОРТСКИ
РАЗРЕД: ДРУГИ

СЕДМИЧНИ БРОЈ ЧАСОВА: 2

ГОДИШЉИ БРОЈ ЧАСОВА: 72

ОПШТИ И ПОСЕБИ ЦИЉЕВИ ПРОГРАМА:

Општи циљеви:

- Разумијевање значаја и улоге коју ИКТ има у савременом друштву;
- Развијање основне информатичке писмености и афинитета за употребу информационо-комуникационих технологија;
- Развијање свијести о употреби рачунара као средства за креирање софтвера;
- Развијање свијести о обавези поштовања правне регулативе и моралних начела у области ИКТ-а;
- Оспособљавање ученика за тимски рад;
- Развијање комуникацијских вјештина, те вјештине презентовања резултата сопственог рада или рада тима;
- Развијање тачности, правовремености и прецизности у раду, упорности и истрајности при рјешавању задатака;
- Развој критичког мишљења и креативности.

Посебни циљеви:

- Развијање способности рјешавања проблема употребом рачунара, кроз анализу проблема, формулисање рјешења и примјену алгоритамских техника;
- Развијање способности алгоритамског начина размишљања, кроз разумијевање и примјену основних алгоритамских структура (линијска, гранање, понављање);
- Развијање вјештина програмирања;
- Разумијевање и примјена принципа креирања добро структурисаних програма;
- Разумијевање везе између развоја програма и алгоритамског начина размишљања;
- Оспособљавање ученика за израду функционалних и визуелно прилагођених интернет страница примјеном стандардних веб технологија и система за управљање садржајем;
- Подстицање креативности и иновативности у примјени алгоритама, програмирања и веб развоја, кроз пројектне задатке и практичну примјену знања.

ПРЕГЛЕД ТЕМАТСКИХ ЦЈЕЛИНА:

Ред. број	Наставна тема	Оквирни број часова
1.	Алгоритми	8-10
2.	Програмирање	30-32
3.	Израда веб презентације	32-34

САДРЖАЈ ПРОГРАМА

1. Тема: Алгоритми	(8-10) Оквирни број часова
---------------------------	-------------------------------

Оперативни циљеви / Исходи	Садржај програма / појмови	Корелација са другим наставним предметима
Ученик: <ul style="list-style-type: none"> – Дефинише алгоритам и наводи основне особине алгоритама (детерминисаност, коначност, ефикасност); – Објашњава разлике између линијских, разгранатих и цикличних алгоритама; – Приказује алгоритме на различите начине (природним језиком, графички – дијаграмом тока, текстуално - псеудокодом); – Идентификује и разликује улазне и излазне податке алгоритама; – Осмишљава и израђује алгоритме за рјешавање конкретних проблемских ситуација из домена елементарног програмирања. 	<ul style="list-style-type: none"> – Појам и основне особине алгоритама. Примјери алгоритама из свакодневног живота. – Линијске, разгранате и цикличне алгоритамаске структуре, сличности и разлике. – Представљање алгоритама на различите начине (природним језиком, графички – дијаграмом тока, текстуално - псеудокодом). – Улазни и излазни подаци алгоритама. Улога промјенивих у алгоритмима. Обрада података у алгоритмима. – Израда алгоритама из домена елементарног програмирања. 	<p>Математика, физика, хемија:</p> <p>Логичко размишљање и алгоритамски приступ рјешавању проблема</p> <p>Енглески језик:</p> <p>Разумијевање ИТ терминологије. Читање и разумијевање техничке документације</p> <p>Комуникација у ИТ индустрији (форуми, упутства, техничка подршка).</p> <p>Осмишљавањем квалитетних задатака, у којима ће примјењивати научено, могуће је остварити корелацију и са другим наставним предметима.</p>

2. Тема: Програмирање	(30-32) Оквирни број часова
------------------------------	--------------------------------

Оперативни циљеви / Исходи	Садржај програма / појмови	Корелација са другим наставним предметима
Ученик: <ul style="list-style-type: none"> – Објашњава значај програмирања у рјешавању проблема у различитим областима струке, науке и савременог живота; – Разликује савремене програмске језике по намјени и карактеристикама; 	<ul style="list-style-type: none"> – Значај програмирања у рјешавању проблема у различитим областима струке, науке и савременог живота. – Врсте савремених програмских језика. Разлике у намјени и карактеристикама савремених програмских језика. – Фазе у развоју програма (анализа проблема, алгоритам, кодирање, тестирање, одржавање). 	<p>Математика, физика, хемија:</p> <p>Логичко размишљање и алгоритамски приступ рјешавању проблема</p> <p>Енглески језик:</p> <p>Разумијевање ИТ терминологије. Читање и разумијевање техничке документације</p>

<ul style="list-style-type: none"> – Објашњава фазе у развоју програма; – Подешава и прилагођава развојно окружење за израду програма у конкретном програмском језику; – Објашњава основну структуру програма; – Разликује основне елементе програмског језика; – Разликује просте и сложене типове података у програмском језику; – Изводи основне аритметичке и логичке операције на простим типовима података; – Развија линијске алгоритме, алгоритме грањања и алгоритме који укључују наредбе понављања; – Креира програме за рјешавање конкретних проблема који укључују наредбе улаза и излаза, те наредбе грањања и наредбе понављања; – Развија алгоритме и креира програме за рјешавање проблема са једнодимензионим низовима; – Самостално рјешава сложене задатке комбинацијом различитих алгоритамских структура и различитих врста података. 	<ul style="list-style-type: none"> –Алатке и интегрисана окружења за развој програма. Подешавање развојног окружења за израду програма. –Основна структура програма у конкретном програмском језику. –Основни елементи синтаксе и семантике процедуралних језика (променљиве, константе, типови, изрази, наредбе). –Прости и сложени типови података. Бројевни типови (скуп вриједности, константе, оператори, библиотечке функције). Карактерски тип (константе, оператори, библиотечке функције). –Основне аритметичке и логичке операције. –Алгоритми и програми који укључују наредбе улаз и излаз, (имплементација формула, цјелобројне аритметике, логичких израза), наредбе грањања (једноставно и сложено (угнијежжено, хијерархијско грањање); наредбе понављања (линеарна обрада серија података (учитавање, испис, статистике (број, збир, производ, минимум, максимум и сл), претрага, угнијежжене петље). –Низови (декларација низа; једнодимензионални низови; ниске карактера, аритметичке операције са низовима, сортирање низова). –Задаци који комбинују различите алгоритамске структуре и различите врсте података. 	<p>Комуникација у ИТ индустрији (форуми, упутства, техничка подршка).</p> <p>Српски језик: Поштовање синтаксичких и семантичких правила, гдје је јасноћа, структура и прецизност у коду подједнако важна као и у правилно написаном тексту.</p>
---	---	---

Тема: Израда веб презентације	(32-34) Оквирни број часова
--------------------------------------	--------------------------------

Оперативни циљеви / Исходи	Садржај програма / појмови	Корелација са другим наставним предметима
-------------------------------	----------------------------	--

<p>Ученик:</p> <ul style="list-style-type: none"> – Упореджује и класификује различите платформе за израду веб презентације; – Објашњава принципе рада система за управљање садржајем на веб-у; – Прикупља дигитални садржај и прилагођава га за потребе објављивања на веб-у; – Објашњава принципе рада система за управљање садржајем на вебу; – Креира основни веб сајт користећи систем за управљање садржајем; – Бира и прилагођава тему веб сајта на основу доступних шаблона; – Конфигурише почетну страницу сајта и додаје основне елементе; – Обавља основне административне задатке: управљање менијима, страницама и садржајем; – Управља корисницима сајта и њиховим дозволама; – Креира функционалан, садржајан и уредно дизајниран веб сајт. 	<ul style="list-style-type: none"> – Појам и значај веб презентација. Преглед врста веб платформи: кодиране веб странице (HTML, CSS, JavaScript), платформе за управљање садржајем. Предности и недостаци различитих платформи. Критеријуми за одабир одговарајуће платформе (намјена, цијена, услуге које платформе нуде). – Системи за управљање садржајем (енгл. Content Management System). Дефиниција, функција и подешавање система за управљање садржајем. Основне компоненте система. Администраторски панел. Улоге корисника у CMS систему. Разлике између класичног развоја веб страница (кодирање HTML + CSS + JavaScript) и развоја уз употребу платформе за управљање садржајем. – Типови дигиталног садржаја (текст, слике, видео и др). Формати и прилагођавање слика за веб. Легалност коришћења садржаја на веб-у. – Основне компоненте система. Администраторски панел. Улоге корисника у CMS систему. – Инсталација локалног веб сервера (XAMPP, WAMP, MAMP). Преузимање и инсталација платформе за управљање садржајем. Основна конфигурација (подешавање базе података, администратора). Рад у локалном окружењу. – Подешавање основних страница (Home, About, Contact). Коришћење администраторског панела и визуелних едитора за израду појединачних страница. Додавање текстуалног садржаја и медија. Организација сајта – категорије, тагови и навигација. – Тема веб сајта. Инсталација и измјена тема. Основна прилагођавања (боје, фонт, изглед странице). Коришћење CSS-а за додатна прилагођавања. 	<p>Математика: Логичко размишљање и алгоритамски приступ рјешавању проблема</p> <p>Енглески језик: Разумијевање ИТ терминологије (термини попут "content management system", "plugin", "theme") Читање и разумијевање техничке документације. Комуникација у ИТ индустрији (форуми, упутства, техничка подршка).</p> <p>Ликовна умјетност: Дизајнирање изгледа веб сајта (боје, типографија, изглед страница). Оптимизација слика за веб. Рад са векторском и растерском графиком.</p> <p>Српски језик: Форматирање и структурирање текстова за веб. Прилагођавање садржаја циљној публици. Креирање блогова, чланака и мултимедијалних садржаја.</p>
---	---	---

	<ul style="list-style-type: none"> –Подешавање почетне странице. Додавање менија и бочних трака. Респонсибилност странице. –Креирање и уређивање страница. Додавање и структура менија. Рад са виџетима и додацима. Управљање објавама и коментарима. –Улоге корисника у CMS-у (администратор, уредник, аутор, посјетилац). Подешавање дозвола и приступа. Управљање регистрацијама корисника. Безбиједност налога и превенција хакерских напада. –Дефинисање сврхе и структуре сајта као самосталног пројекта једног ученика или групног пројекта више ученика. Додавање смисленог, исправно организованог и уређеног садржаја у веб сајт. 	
--	---	--

ДИДАКТИЧКА УПУТСТВА И ПРЕПОРУКЕ:

Квалитетно организована настава подразумева самосталан рад ученика на рачунару. Наставницима који реализују Наставни програм остављена је могућност избора програма у складу са степеном опремљености информатичке учионице у којој се реализује настава.

Како би се утврдио ниво знања које су ученици стекли у претходном школовању препоручено је на почетку сваке наставне теме исто провјерити (нпр. практичним тестом), те резултате користити приликом планирања и припремања наставе.

При реализацији прве наставне теме *Алгоритми* неопходно је код ученика развијати алгоритамски начин размишљања. Потребно је да схвате шта је алгоритам, које су његове предности, о чему треба водити рачуна приликом његове израде.

Наставник може навести примјере алгоритама из свакодневног живота и подстаћи ученике да и они предложе реалне ситуације када алгоритамски начин размишљања доводи до рјешавања проблема.

Такође, важно је инсистирати на употреби одговарајућих типова улазних и излазних величина, те разјаснити како се алгоритми користе при програмирању. Треба да разликују линијске, разгранате и цикличне алгоритме, те да науче алгоритме да представљају на различите начине. Ученике је потребно оспособити да анализирају алгоритам, да идентификују његове предности и недостатке, те да исте презентују другим. Препоручено је припремити квалитетне задатке, којима ће се остварити повезивање са другим наставним предметима.

У оквиру наставне теме *Програмирање* потребно је навести ученике да размишљају о значају програмирања при рјешавању проблема у различитим областима струке, науке и свакодневног живота, упоредити како се различити послови обављају уз помоћ програма, а како су се обављали без њега, направити паралелу између различитих програмских језика (намјена, карактеристике), дискутовати о могућим правцима развоја програмирања, у склопу развоја ИКТ-а, у будућности.

Неопходно је да ученици схвате како се развија програм, да разумију да је програмирање процес који се састоји из више фаза, те да схвате значај сваке од њих. Ученици треба да науче како се подешава развојно окружење, те који су основни елементи програмског језика (којег ће изучавати). Потребно их је оспособити да доносе одлуке и врше избор одговарајуће програмске структуре (линијске, разгранате или цикличне) у зависности од проблема који рјешавају, да их комбинују, користе одговарајуће типове податак (просте и сложене), користе одговарајуће наредбе (нпр. петља са или без унапријед познатог броја понављања) и сл.

Ученике је потребно оспособити да вреднују различита програмска рјешења водећи рачуна о ефикасности са становишта употребе рачунарских ресурса и времена.

Препоручено је користити програмски језик C/C++ (препоручује се *Code::Blocks*, као интегрисано развојно окружење (*IDE*), али је могуће користити и друга интегрисана развојна окружења за C/C++ (*Eclipse*, *Visual Studio*, *Visual Studio Code*, *Dev C++*).

Ученицима треба нагласити да су грешке при програмирању саставни дио процеса учења и подстицати их да грешке самостално анализирају и исправљају. Охрабривати их да прво сами тумаче поруке о грешкама и користе исписивање вриједности промјењивих (*print debugging*). Важно је развијати навику тестирања кода након сваке измјене и подстицати међусобну анализу програма. Ученици треба да науче како да ефикасно претражују рјешења на интернету и користе документацију програмског језика. Наставник треба да његује културу истраживања и аналитичког размишљања, чиме развија вјештину самосталног рјешавања проблема.

За реализацију исхода учења на тему *Израда веб презентације*, наставник треба да, у складу са техничким могућностима и сопственом стручном процјеном одабере одговарајући систем за управљање садржајем на вебу. Препоручује се да се користи Wordpress или нека друга бесплатна платформа. За инсталацију платформе, потребно је обезбиједити интернет конекцију и софтвер за веб развој који обухвата веб сервер, базу података и одговарајући програмски језик (нпр. XAMPP). Наставник може ученицима да помогне да на локалном рачунару инсталирају неопходне програме за покретање система за управљање садржајем. Од ученика не треба захтијевати да у потпуности владају технологијама које стоје „испод“ самог система, односно није неопходно дубоко познавање рада веб сервера, база података и програмирања у скриптном језику. Препоручљиво је да ученици понове градиво деветог разреда које се односи на HTML и принципе које треба поштовати приликом припреме и објављивања садржаја на вебу.

Ученици се могу подијелити у групе, чиме се подстиче сарадња и заједнички рад. Треба мотивисати ученике да приликом припреме дигиталног садржаја и организације садржаја на страници покажу и развијају сопствену креативност. Треба развијати критичко размишљање при употреби информација на интернету.

Водити рачуна о заштити личних података и безбједном понашању на интернету.

Дискутовати о ауторским правима и правној етици коришћења дигиталног садржаја.

Указивати ученицима на значај дигиталних технологија на друштво и савремени живот.

Поучавање и учење наставног предмета Рачунарство и информатика неопходно је организовати у информатичкој учионици, која посједује радно мјесто за сваког ученика опремљено умреженим рачунаром спојеним на интернет. Рачунари морају имати могућност покретања свих потребних програма, који на њима морају бити и инсталирани, као и чувања свих материјала потребних за учење. Опрема у учионици мора се одржавати у функционалном стању. Препоручено је да учионица буде опремљена и неким од софтвера за управљање учионицом.

Рад у учионици препоручено је организовати тако да сваки ученик има своје радно мјесто, које дијели са ученицима других разреда и одјељења. Поучавање и учење потребно је организовати тако да ученици стекну функционално знање, инсистирати дакле на практичној примјени знања, развијању вјештине употребе ИКТ-а у свакодневном животу и раду.

Исходи учења не морају се реализовати хронолошким редослиједом, дозвољено је креативно планирати процес поучавања и учења и ускладити га са другим исходима у оквиру наставног предмета Рачунарство и информатика, као и са исходима учења других наставних предмета. Поједини исходи учења не захтијевају једнако вријеме учења.

Настава треба да буде интерактивна и усмјерена на ученика. Препоручује се комбиновање различитих наставних метода нпр: методе демонстрације (нпр. упознавање са основним концептима путем презентација и примјера), методе разговора (нпр. подстицање ученика да анализирају функционалност, дизајн, безбиједност и укупан квалитет веб сајтова, да анализирају и пореде различита програмска рјешења), метод практичног рада, те различитих облика рада: индивидуални рад (нпр. задаци који подстичу анализу и примјену знања), рад у паровима или групама (нпр. сарадничко учење кроз пројекте и дискусије).

Такође, препоручује се и примјена: проблемског учења (нпр. истраживање и прикупљање садржаја везаног за важне теме из свакодневног живота), те пројектних задатак (нпр. креирање сопствених веб сајтова као крајњи исход учења; креирање алгорита и/или програма за рјешавање конкретних проблема).