

**KATALOG**

**ZNANJA**

**IZ**

**INFORMATIKE**

Nacionalni savjet za obrazovanje je na 27. sjednici održanoj 17. marta 2014. godine utvrdio izmjene predmetnoga programa INFORMATIKA za I razred gimnazije.

Na zahtijev Pedagoško-psihološke službe Gimnazije „Slobodan Škerović“, Aktiv informatike je razložio katalog znanja iz pomenutog dokumenta po klasifikacionim periodima.

### **Tekst koji slijedi je izvučen iz javno važećeg obrazovnog programa za Gimanziju**

Načini provjeravanja znanja i ocjenjivanja

Provjera znanja obavlja se na osnovu usmenoga ispitivanja i analize praktičnoga rada učenika/učenice.

Usmenim ocjenjivanjem vrednuje se upotreba informatičkih znanja i razumijevanje osnovnih zakonitosti informatike.

Analiza praktičnoga rada izvodi se razgovorom, a dio ocjene čine:

- kvalitet izrade zadatka, odnosno projekta
- uspješnost predstavljanja informacija.

Kod ocjene kvaliteta izrade zadatka posebna pažnja posvećuje se aktuelnosti izabranoga zadatka, preciznosti predloženoga rješenja,

pouzdanosti upotrijebljenih izvora podataka, kvalitetu i aktuelnosti dobijenih podataka, širini predstavljene informacije, tehničkom

izgledu rješenja, bogatstvu i jasnoći izlaganja i uloženoome trudu pri izradi zadatka.

Kod uspješnosti predstavljanja informacija ocjenjuje se samostalnost u radu, sposobnost predstavljanja, sposobnost procjene kvaliteta

izvornih podataka i dobijenih rezultata, umješnost obrazlaganja predloženoga rješenja i upotrijebljenih postupaka.

## Katalog znanja za INFORMATIKU

### Minimalni zahtjevi

#### Učenik/učenica:

- navede faze razvoja informacione tehnologije
- zna osnovne karakteristike brojevnih sistema
- zna da konvertuje brojeve iz jednoga brojevnog sistema u drugi
- prepozna i objasni namjenu osnovnih dijelova računara
- prepozna i objasni namjenu perifernih uređaja računara i portova
- prepoznaje i objasni osnovnu namjenu računarskih memorija
- zna osnovne mjerne jedinice za kapacitet memorije i brzinu rada procesora
- mijenja osnovne postavke radnoga okruženja računara
- prepoznaje različite tipove fajlova i zna da ih sačuva na različitim lokacijama i po različitim tipovima
- pronalazi fajlove i foldere po različitim kriterijumima
- kreira i čuva tekstualne datoteke
- ispravno kuca tekst
- uređuje sadržaj teksta u tekstualnoj datoteci
- formira jednostavne tabele u tekstualnoj datoteci
- umeće i mijenja jednostavne slike i objekte u tekst
- štampa tekstualnu datoteku
- objašnjava osnovne pojmove multimedijalnoga predstavljanja informacija
- crta jednostavne grafičke oblike u programu za obradu crteža
- unosi slike u računar s perifernih uređaja
- koriste dostupne aplikacije za reprodukciju audio- i videozapisa
- kreira jednostavnu prezentaciju
- zna osnovne pojmove i vrste računarskih mreža
- prepoznaje osnovne komponente mrežnoga hardvera
- zna osnovne načine prijenosa podataka i povezivanja na Internet
- prepoznaje osnovne internetske servise i koristi ih
- koristi programe za pregled weba
- kreira i postavlja blog
- zna da na webu pronađe podatke i upotrijebi ih za oblikovanje informacije
- priprema osnovne web-izlaze
- komunicira putem elektronske pošte
- koristi pravila lijepoga ponašanja na mreži
- zna osnovne načine zaštite elektronskih komunikacija
- zna osnovne mjere bezbjednosti na Internetu
- opiše najčešće mogućnosti širenja računarskih virusa i načine zaštite
- zna osnovne načine zaštite elektronskih podataka od neovlaštenoga korišćenja i tehničkih neispravnosti
- predstavi intuitivni pojam algoritma
- nabroji osnovne elemente algoritma
- prepoznaje tipove algoritamskih shema
- samostalno izrađuje jednostavne algoritme linijskim algoritamskim strukturama.

**Osnovni zahtjevi**

**Učenik/učenica:**

- objasni osnovne karakteristike faza računara
- konvertuje brojeve iz jednog brojevnog sistema u drugi
- prepozna i objasni namjenu osnovnih dijelova računara i njihove funkcionalne veze
- prepoznaje i objasni namjenu perifernih uređaja računara i portova
- prepoznaje i objasni namjenu računarskih memorija
- zna mjerne jedinice za kapacitet memorije i brzinu rada procesora, kao i njihovo pretvaranje
- mijenja osnovne postavke radnog okruženja računara
- prepoznaje različite tipove fajlova i zna da ih sačuva na različitim lokacijama i po različitim tipovima
- sortira i pronalazi fajlove i foldere po različitim kriterijumima
- koristi ugrađene sisteme pomoći
- kreira i čuva tekstualne datoteke
- ispravno kuca tekst
- uređuje sadržaj teksta u tekstualnoj datoteci
- formira i uređuje tabele u tekstualnoj datoteci
- umeće i mijenja slike i objekte u tekst
- priprema tekstualnu datoteku za štampu
- objašnjava osnovne pojmove multimedijalnog predstavljanja informacija
- zna ulogu grafike u multimedijalnom predstavljanju informacija
- crta i mijenja jednostavne grafičke oblike u programu za obradu crteža
- unosi slike u računar s perifernih uređaja i mijenja njihove osnovne parametre
- zna ulogu zvuka i videozapisa u multimedijalnom predstavljanju informacija
- koristi dostupne aplikacije za reprodukciju audio- i videozapisa
- zna ulogu prezentacije u multimedijalnom predstavljanju informacija
- kreira i oblikuje prezentaciju s animacijama i bez animacija
- zna osnovne pojmove i vrste računarskih mreža
- prepoznaje osnovne komponente mrežnog hardvera
- zna načine prijenosa podataka i povezivanja na Internet
- prepoznaje internetske servise
- koristi programe za pregled weba,
- zna da na webu pronađe podatke i upotrijebi ih za oblikovanje informacije
- priprema web-izlaze
- komunicira različitim vidovima elektronske komunikacije
- pravilno komunicira putem elektronske pošte
- koristi pravila lijepoga ponašanja na mreži
- zna načine zaštite elektronskih komunikacija
- zna mjere bezbjednosti na Internetu
- opiše mogućnosti širenja računarskih virusa i načine zaštite
- zna načine zaštite elektronskih podataka od neovlašćenoga korišćenja i tehničkih neispravnosti
- predstavi intuitivni pojam algoritma
- nabroji osnovne elemente algoritma
- prepoznaje tipove algoritamskih shema
- samostalno izrađuje algoritme i rješava jednostavne probleme
- razumije ulogu programa i programskih jezika

## Katalog znanja za INFORMATIKU

### Napredni zahtjevi

#### Učenik/učenica:

- predstavi tekstualne i numeričke podatke u računaru
- objasni princip rada računara
- upoređuje karakteristike računara, na osnovu sistemskih podataka
- mijenja različite postavke radnoga okruženja računara
- instalira i deinstalira jednostavne aplikativne programe
- koristi napredne opcije upravljanja fajlovima i folderima
- kreira i kreativno uređuje tekst
- koristi širok dijapazon multimedijalnog predstavljanja informacija
- zna vrste računarskih mreža
- koristi različite načine prijenosa podataka i povezivanja na Internet
- prepoznaje internetske servise i koristi ih
- kreira i postavlja blog
- pravilno komunicira putem elektronske pošte
- koristi pravila lijepoga ponašanja na mreži
- zna načine zaštite elektronskih komunikacija i koristi ih
- zna mjere bezbjednosti na Internetu i koristi ih
- zna načine zaštite elektronskih podataka od neovlašćenoga korišćenja i tehničkih neispravnosti
- samostalno izrađuje algoritme i rješava postavljene probleme
- piše i testira izvršni program.

# **KATALOG ZNANJA**

**RAČUNARSTVO I INFORMATIKA**

**ZA I RAZRED MATEMATIČKE GIMNAZIJE**

**Tekst koji slijedi predstavlja dio smjernica preuzetih iz javno važećeg obrazovnog programa za matematičku gimnaziju**

Računarstvo i informatika je opšteobrazovni predmet koji kombinuje osnove teorije informatike i računarskih nauka sa metodama neposrednog prikupljanja, skladištenja, distribucije i obrade podataka. To podrazumijeva sticanje neophodnog nivoa informatičkih znanja, tzv. informatičke pismenosti, potrebne za život i rad u informacionom društvu.

Ovim programom učenici/e stižu znanja potrebna za razumijevanje i upotrebu informacionih tehnologija u svakodnevnom učenju i radu.

Predmet favorizuje rad na projektima, problemsko učenje, biranje sadržaja prema interesovanjima i sposobnostima učenika/ca i uopštavanje njihovih ideja. Važan aspekt predmeta predstavlja povezivanje sa drugim predmetima, kroz rad sa nastavnicima/ama tih predmeta, spoljnim saradnicima, školama i ustanovama. Posebno se pažnja posvećuje unapređivanju sposobnosti kreativnog i logičkog mišljenja, sa ciljem da se kod učenika/ca razvija razumno i samostalno odlučivanje u novim i nepredviđenim okolnostima.

### **Načini provjeravanja znanja i ocjenjivanja**

Rad učenika treba sistematski pratiti i ocjenjivati u toku cijele školske godine.

Provjera znanja se obavlja na osnovu usmenog ispitivanja, kontrolnih zadataka i analize praktičnog rada učenika/ce.

Usmenim ocjenjivanjem vrednuje se upotreba informatičkih znanja i razumijevanje osnovnih zakonitosti informatike.

Kontrolni zadaci se mogu zadavati poslije svake obrađene teme i treba da sadrže zadatke i pitanja različitog nivoa težine

Analiza praktičnog rada izvodi se razgovorom, a sastavni dio ocjene čine:

- kvalitet izrade zadatka, odnosno projekta,
- uspješnost predstavljanja informacija.

Kod ocjene kvaliteta izrade zadatka posebna pažnja posvećuje se aktuelnosti izabranog zadataka, preciznosti predloženog rješenja, pouzdanosti upotrijebljenih izvora podataka, kvalitetu i aktuelnosti dobijenih podataka, širini predstavljene informacije, tehničkom izgledu rješenja, bogatstvu i jasnoći izlaganja i uloženom trudu pri izradi zadataka.

Kod uspješnosti predstavljanja informacija ocjenjuje se samostalnost u radu, sposobnost predstavljanja, sposobnost procjene kvaliteta izvornih podataka i dobijenih rezultata, umješnost obrazlaganja predloženog rješenja i upotrijebljenih postupaka. Predložena rješenja od strane učenika, analiziraju se i ocjenjuju prema pojedinačnim sposobnostima učenika, vodeći računa o dostupnosti opreme i drugih resursa u školi

<b>Katalog znanja za RAČUNARSTVO I INFORMATIKU</b>
<b>Tema 1: GRAĐA RAČUNARA</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Opiše osnovnu fon-Nojmanovu arhitekturu; Opiše ulogu i funkcije ALU, kontrolne i memorijske jedinice procesora; objasni razlike između tipova primarne memorije; objasni potrebu za magistralama (buses);
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
Identifikuje potrebu za registrima posebne namjene i način njihove upotrebe; Opiše osnovna svojstva, prednosti i mane medija sekundarne memorije (npr. magnetni, optički, solid-state);
<b>Napredni zahtjevi</b> Učenik/učenica treba da:
opiše svojim riječima "Fetch decode execute" ciklus i razumije uticaj faza na pojedine registre.
<b>Tema 2: Algoritmi</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Usvoji intuitivnu definiciju algoritma; Identifikuje potrebu za različitim načinima predstavljanja algoritama; Izvršava korake algoritma zadatog prirodnim jezikom, pseudokodom ili shemom;
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
Opiše svojim riječima šta je rezultat izvršavanja algoritma; Identifikuje ulazne i izlazne vrijednosti problema; Kreira linijske, razgranate i ciklične strukture kojima se rješava problem srednje težine;
<b>Napredni zahtjevi</b> Učenik/učenica treba da:
Analizira rad zadate linijske, razgranate i ciklične strukture pri rješavanju problema srednje težine; Kreira linijske, razgranate i ciklične strukture kojima se rješava problem višeg nivoa zahtijevnosti; Kreira algoritam za dati problem kombinovanjem različitih struktura. Intuitivno definiše pojam funkcije (potprograma);



<b>Tema 3: Matematičke osnove računara</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Učenik/ca treba da: Identifikuje potrebu za postojanjima različitih brojnih sistema; Zapisuje dekadne brojeve u binarnom i heksadecimalnom obliku; zapisuje binarne brojeve u dekadnom i heksadecimalnom obliku; Izvršava operaciju pretvaranja broja iz nekog brojnog sistema u dekadni; Sabira, oduzima, množi i dijeli brojeve u različitim brojnim sistemima definiše funkcije NE, ILI, I, NOT, NI, NILI; predstavi funkcije NE, ILI, I, NOT, NI, NILI odgovarajućim logičkim kolima;
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
opiše metode predstavljanja cijelih brojeva: znak i apsolutna vrijednost i potpuni komplement; demonstrira razumijevanje predstavljanja realnog broja u floating point zapisu; kreira izraz i tablicu na osnovu logičkog sklopa; kreira logički sklop na osnovu izraza; kreira logički sklop i izraz na osnovu tablice.
<b>Napredni zahtjevi</b> Učenik/učenica treba da:
Realizuje minimizirano logičko kolo i minimizira bilo koju funkciju, zadatu na bilo koji način.
<b>Tema 4: GODIŠNJI PROJEKAT</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Napiše esej na zadatu temu; Kreira prezentaciju sa istom temom kao i esej;
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
Prikupi informacije o izabranoj temi koristeći razne izvore: knjige, časopise, internet... formuliše naslove poglavlja eseja; napiše esej primjenom programa za obradu teksta; kreira prezentaciju primjenom odgovarajućeg softvera;
<b>Napredni zahtjevi</b> Učenik/učenica treba da:
Kreira besplatan Google ili Wordpress nalog; Kreira nekoliko veb-stranica primjenom Google sites ili Wordpress-a.

# **KATALOG ZNANJA**

**RAČUNARSTVO I INFORMATIKA**

**ZA II RAZRED MATEMATIČKE GIMNAZIJE**

## **Tekst koji slijedi je izvučen iz javno važećeg obrazovnog programa za matematičku gimnaziju**

Računarstvo i informatika je opšteobrazovni predmet koji kombinuje osnove teorije informatike i računarskih nauka sa metodama neposrednog prikupljanja, skladištenja, distribucije i obrade podataka. To podrazumijeva sticanje neophodnog nivoa informatičkih znanja, tzv. informatičke pismenosti, potrebne za život i rad u informacionom društvu.

Ovim programom učenici/e stižu znanja potrebna za razumijevanje i upotrebu informacionih tehnologija u svakodnevnom učenju i radu.

Predmet favorizuje rad na projektima, problemsko učenje, biranje sadržaja prema interesovanjima i sposobnostima učenika/ca i uopštavanje njihovih ideja. Važan aspekt predmeta predstavlja povezivanje sa drugim predmetima, kroz rad sa nastavnicima/ama tih predmeta, spoljnim saradnicima, školama i ustanovama. Posebno se pažnja posvećuje unapređivanju sposobnosti kreativnog i logičkog mišljenja, sa ciljem da se kod učenika/ca razvija razumno i samostalno odlučivanje u novim i nepredviđenim okolnostima.

### **Načini provjeravanja znanja i ocjenjivanja**

Rad učenika treba sistematski pratiti i ocjenjivati u toku cijele školske godine.

Provjera znanja se obavlja na osnovu usmenog ispitivanja, kontrolnih zadataka i analize praktičnog rada učenika/ce.

Usmenim ocjenjivanjem vrednuje se upotreba informatičkih znanja i razumijevanje osnovnih zakonitosti informatike.

Kontrolni zadaci se mogu zadavati poslije svake obrađene teme i treba da sadrže zadatke i pitanja različitog nivoa težine

Analiza praktičnog rada izvodi se razgovorom, a sastavni dio ocjene čine:

- kvalitet izrade zadatka, odnosno projekta,
- uspješnost predstavljanja informacija.

Kod ocjene kvaliteta izrade zadatka posebna pažnja posvećuje se aktuelnosti izabranog zadataka, preciznosti predloženog rješenja, pouzdanosti upotrijebljenih izvora podataka, kvalitetu i aktuelnosti dobijenih podataka, širini predstavljene informacije, tehničkom izgledu rješenja, bogatstvu i jasnoći izlaganja i uloženom trudu pri izradi zadataka.

Kod uspješnosti predstavljanja informacija ocjenjuje se samostalnost u radu, sposobnost predstavljanja, sposobnost procjene kvaliteta izvornih podataka i dobijenih rezultata, umješnost obrazlaganja predloženog rješenja i upotrijebljenih postupaka. Predložena rješenja od strane učenika, analiziraju se i ocjenjuju prema pojedinačnim sposobnostima učenika, vodeći računa o dostupnosti opreme i drugih resursa u školi

## Katalog znanja za RAČUNARSTVO I INFORMATIKU

### Minimalni zahtjevi

Učenik/učenica treba da:

- Razumije pojam programski jezik;
- zna historijat razvoja i vrste programskih jezika;
- zna postupak rješavanja nekog problema programiranjem;
- zna da upotrijebi osnovne algoritme u procesu programiranja;
- shvata razliku između proceduralnih i objektno orijentisanih programskih jezika;
- zna elemente programskog okruženja za pisanje programa;
- razlikuje tipove podataka;
- koristi aritmetičke i operatore proširenog pridruživanja;
- zna da upotrijebi promjenljivu;
- upotrebljava osnovne ugrađene funkcije;
- upotrebljava relacione i logičke operatore;
- koristi uslovne naredbe if-elif-else i if i if-else u elementarnim zadacima;
- upotrebi FOR petlju u elementarnim zadacima;
- upotrebi WHILE i/ili DO WHILE petlju u elementarnim zadacima;
- iskoristi vlastitu funkciju i modul u pisanju programa u elementarnim zadacima;
- koristi nizove znakova - STRING-ove u izradi vlastitih programa u elementarnim zadacima;

### Osnovni zahtjevi

Učenik/učenica treba da:

- Razumije pojam programski jezik;
- zna historijat razvoja i vrste programskih jezika;
- zna postupak rješavanja nekog problema programiranjem;
- zna da upotrijebi algoritam u procesu programiranja;
- zna razliku između proceduralnih i objektno orijentisanih programskih jezika;
- zna elemente programskog okruženja za pisanje programa;
- razlikuje tipove podataka;
- koristi aritmetičke i operatore proširenog pridruživanja u složenijim zadacima;
- zna da upotrijebi promjenljivu u složenijim zadacima;
- zna postupak čuvanja promjenljive u memoriji;
- upotrebljava osnovne ugrađene funkcije u složenijim zadacima;
- zna da upiše, spremi i izvrši program u programskom okruženju;
- upotrebljava relacione i logičke operatore u složenijim zadacima;
- koristi uslovne naredbe if-elif-else i if i if-else u složenijim zadacima;
- upotrebi FOR petlju; -upotrebi WHILE i/ili DO WHILE petlju u složenijim zadacima;
- iskoristi vlastitu funkciju i modul u pisanju programa u složenijim zadacima;
- koristi nizove znakova - STRING-ove u izradi vlastitih programa u složenijim zadacima;

-koristi metode za rad sa nizovima/listama u izradi vlastitih programa u složenijim zadacim.

### **Napredni zahtjevi**

Učenik/učenica treba da:

- Razumije pojam programski jezik;
- zna istorijat razvoja i vrste programskih jezika;
- zna postupak rješavanja nekog problema programiranjem;
- zna da upotrijebi algoritam u procesu programiranja;
- zna razliku između proceduralnih i objektno orijentisanih programskih jezika;
- zna elemente programskog okruženja za pisanje programa;
- razlikuje tipove podataka;
- koristi aritmetičke i operatore proširenog pridruživanja za složene zadatke;
- zna da upotrijebi promjenljivu u složenim zadacima;
- zna postupak čuvanja promjenljive u memoriji;
- upotrebljava osnovne ugrađene funkcije u složenim zadacima;
- zna da upiše, spremi i izvrši program u programskom okruženju;
- upotrebljava relacione i logičke operatore u složenim zadacima;
- koristi uslovne naredbe if-elif-else i if i if-else u složenim zadacima;
- upotrebi FOR petlju; -upotrebi WHILE i/ili DO WHILE petlju u složenim zadacima;
- iskoristi vlastitu funkciju i modul u pisanju programa u složenim zadacima;
- koristi nizove znakova - STRING-ove u izradi vlastitih programa u složenim zadacima;
- koristi metode za rad sa nizovima/listama u izradi vlastitih programa u složenim zadacima.

# **KATALOG ZNANJA**

**PROGRAMIRANJE I PROGRAMSKI JEZICI**

**ZA III I IV RAZRED MATEMATIČKE GIMNAZIJE**

## **Tekst koji slijedi predstavlja dio smjernica preuzetih iz javno važećeg obrazovnog programa za matematičku gimanziju**

Programiranje i programski jezici je specijalistički predmet koji kombinuje osnove teorije informatike i računarskih nauka sa metodama neposrednog prikupljanja, skladištenja, distribucije i obrade podataka. To podrazumijeva sticanje neophodnog nivoa programerskih znanja potrebnih za život i rad u informacionom društvu i dalji nastavak školovanja.

Ovim programom učenici/e stiču znanja potrebna za razumijevanje i upotrebu različitih programerskih alata i informacionih tehnologija u svakodnevnom učenju i radu i ovladavaju postupcima rješavanja problema razvojem i implementacijom odgovarajućeg softvera.

Predmet favorizuje rad na projektima, problemsko učenje, biranje sadržaja prema interesovanjima i sposobnostima učenika/ca i uopštavanje njihovih ideja. Važan aspekt predmeta predstavlja povezivanje sa drugim predmetima, kroz rad sa nastavnicima/ama tih predmeta, spoljnim saradnicima, školama i ustanovama. Posebno se pažnja posvećuje unapređivanju sposobnosti kreativnog i logičkog mišljenja, sa ciljem da se kod učenika/ca razvija razumno i samostalno odlučivanje u novim i nepredviđenim okolnostima.

Međupredmetne oblasti/teme su sadržaji koji omogućavaju da se u opšteobrazovni kurikulum uključe određeni ciljevi i sadržaji obrazovanja koji nijesu dio formalnih disciplina ili pojedinih predmeta, ili koji su po strukturi interdisciplinarni. Ovi sadržaji doprinose integrativnom pristupu opšteg obrazovanja i u većoj mjeri povezuju sadržaje pojedinih predmeta.

### **Načini provjeravanja znanja i ocjenjivanja**

Rad učenika treba sistematski pratiti i ocjenjivati u toku cijele školske godine.

Provjera znanja se obavlja na osnovu usmenog ispitivanja, kontrolnih zadataka i analize praktičnog rada učenika/ce.

Usmenim ocjenjivanjem vrednuje se upotreba informatičkih znanja i razumijevanje osnovnih koncepata programiranja.

Kontrolni zadaci se mogu zadavati poslije svake obrađene teme i treba da sadrže zadatke i pitanja različitog nivoa težine

Analiza praktičnog rada izvodi se razgovorom, a sastavni dio ocjene čine:

- kvalitet izrade zadatka, odnosno projekta,
- nivo optimizacije konačnog rješenja zadatka,
- originalnost idejnog rješenja

Kod ocjene kvaliteta izrade zadatka posebna pažnja posvećuje se aktuelnosti izabranog zadataka, preciznosti predloženog rješenja,

kvalitetu i aktuelnosti dobijenih podataka, tehničkom izgledu rješenja, bogatstvu i jasnoći izlaganja i uloženom trudu pri izradi zadataka.

Kod uspješnosti predstavljanja informacija ocjenjuje se samostalnost u radu, sposobnost predstavljanja, sposobnost procjene kvaliteta izvornih podataka i dobijenih rezultata, umješnost obrazlaganja predloženog rješenja i upotrijebljenih postupaka. Predložena rješenja od strane učenika, analiziraju se i ocjenjuju prema pojedinačnim sposobnostima učenika, vodeći računa o dostupnosti opreme i drugih resursa u školi

<b>Katalog znanja za PROGRAMIRANJE I PROGRAMSKE JEZIKE (III RAZRED)</b>
<b>Tema 1: RAČUNARSKE MREŽE I INTERNET</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Definiše serijski i paralelni prenos podataka Obrazloži prednosti serijskog u odnosu na paralelni prenos Definiše i poredi sinhroni i asinhroni prenos podataka Opiše ulogu start bita i stop bita Nabroji komponente potrebne za bežično umrežavanje (Wireless network adapter, Wireless access point) i metodama sigurnosti
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
Definiše pojmove: baud rate, bit rate, bandwidth, latency, protocol. Razlikuje baud rate i bit rate. Objasni vezu između bit rate i bandwidth-a. Objasni topologiju zvijezde i logičkog bus-a: Objasni operacije obje topologije Opisuje situacije gdje se mogu koristiti peer to peer i klijent/server Objasni ulogu WiFi Objasni rad komponenti potrebnih za bežično umrežavanje (Wireless network adapter, Wireless access point) i metodama sigurnosti
<b>Napredni zahtjevi</b> Učenik/učenica treba da:
Objasni princip rada protokola CSMA/CA sa i bez RTS/CTS. Bude upoznat sa ulogom Service Set Identifier (SSID). Definiše pojmove WWW, HTTP, FTP, URL Prepoznaje komponente URL-a i IP adrese Objasni razliku između IPv4 i IPv6
<b>Tema 2: VEB DIZAJN</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Objasni princip rada veb-čitača (browser-a) Objasni princip rada veb-servera Opiše strukturu tipične veb stranice Definiše pojmove element (tag), atribut i svojstvo



<p>Primijeni metode dodavanja teksta, slika, linkova, listi, tabela i multimedijalnih sadržaja na veb-stranicu</p>
<p><b>Osnovni zahtjevi</b> Učenik/učenica treba da:</p>
<p>Razumije upotrebu kaskadnih stilova (CSS) Definiše pojmove: selektor, svojstvo, pravilo Kreira sopstvene stilove Uključuje biblioteke/funkcije u svoju veb-stranicu Kreira veb-stranice primjenom HTML-a i CSS-a</p>
<p><b>Napredni zahtjevi</b> Učenik/učenica treba da:</p>
<p>Kreira veb-stranice primjenom HTML-a, CSS-a i JavaScript biblioteka</p>
<p><b>Tema 3: DATOTEKE</b></p>
<p><b>Minimalni zahtjevi</b> Učenik/učenica treba da:</p>
<p>Objasni ulogu datoteka u 4programu Opiše korake za otvaranje datoteke</p>
<p><b>Osnovni zahtjevi</b> Učenik/učenica treba da:</p>
<p>Objasni razlike u metodama za čitanje podatka iz datoteke Objasni razlike u metodama za ispis podataka u datoteku</p>
<p><b>Napredni zahtjevi</b> Učenik/učenica treba da:</p>
<p>Primijeni metode za rad s datotekama u pisanju programa</p>
<p><b>Tema 4: OBJEKTNO ORJENTISANO PROGRAMIRANJE</b></p>
<p><b>Minimalni zahtjevi</b> Učenik/učenica treba da:</p>
<p>Objasni strukturu objektno-orijentisanog programa Prepoznaju ugrađene klase i funkcije i metode Pišu konstruktor klase Objasni konstruktor klase</p>
<p><b>Osnovni zahtjevi</b> Učenik/učenica treba da:</p>
<p>Poveže ugrađene klase i funkcije i metode sa objektno-orijentisanim programiranjem Navodi primjer metoda sa dodatnim parametrima Opiše specijalne metode za rad s objektima Primijeni objektno-orijentisani način programiranja u stvaranju algoritma i programa</p>
<p><b>Napredni zahtjevi</b> Učenik/učenica treba da:</p>
<p>Objasni pojam nasljeđivanja klase</p>

<b>Kreira klasu nasljednika</b>
<b>Tema 5: REKURZIVNI ALGORITMI</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Objasni induktivni pristup rješavanju problema Objasni pojam rekurzije Na primjeru prepozna problem na koji se može primijeniti induktivni pristup Skicira shemu rekurzivnih poziva date funkcije
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
Izvede rekurzivnu relaciju i uslov prekida za dati problem Objasni pojam memoizacija
<b>Napredni zahtjevi</b> Učenik/učenica treba da:
Primijeni rekurzivne algoritme u rješavanju problema
<b>Tema 6: SLOŽENOST ALGORITAMA</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Objasni pojam složenosti algoritma
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
Izvrši kategorizaciju složenosti algoritama prema notaciji veliko O
<b>Napredni zahtjevi</b> Učenik/učenica treba da:
Analizira algoritam linearnog traženja u neuređenoj listi/nizu Analizira algoritam traženja elementa u sortiranoj listi Uporedi vremena traženja elementa u neuređenoj i sortiranoj listi
<b>Tema 7: ALGORITMI SORTIRANJA</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Opiše algoritam sortiranje zamjenom elemenata (exchange sort) Opiše algoritam izbora najmanjeg elementa (selection sort) Opiše algoritam zamjene susjednih elemenata (bubble sort) Opiše algoritam umetanjem (insertion sort) Opiše algoritam sjedinjavanja (merge sort) Opiše algoritam brzog sortiranja (quick sort) Opiše algoritam razvrstavanjem (bucket sort) Uporedi algoritme sortiranja
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
Implementira probleme sortiranja bez špotrebe za primjenom rekurzija
<b>Napredni zahtjevi</b>

Učenik/učenica treba da:
Implementira probleme sortiranja sa primjenom rekurzija
<b>Tema 8: VIŠEDIMENZIONALNI NIZOVI</b>
<b>Minimalni zahtjevi</b>
Učenik/učenica treba da:
Pojašnjava pojam višedimenzionalnog niza i princip čuvanja elemenata višedimenzionalnog niza u memoriji
<b>Osnovni zahtjevi</b>
Učenik/učenica treba da:
Modelira tabelu 2D nizom Piše programe u kojima koristi 2D nizove
<b>Napredni zahtjevi</b>
Učenik/učenica treba da:
Modelira tabelu nD nizom Piše programe u kojima koristi n-dimenzionalne nizove
<b>Tema 9: ELEMENTI 2D GRAFIKE</b>
<b>Minimalni zahtjevi</b>
Učenik/učenica treba da:
Objasni izgled grafičkog prozora Opiše funkcije za pomjeranje pera/olovke Opiše metode za crtanje pravouganika i linije Navede metode za upis i ispis teksta u grafičkom prozoru
<b>Osnovni zahtjevi</b>
Učenik/učenica treba da:
Primijeni algoritam za crtanje pravilnog n-trougla Opiše metodu za crtanje kruga i kružnog isječka Objasni metode za postavljanje boje pera/olovke i popunjavanja.
<b>Napredni zahtjevi</b>
Učenik/učenica treba da:
Primijeni grafičke funkcije u izradi programa

<b>Katalog znanja za PROGRAMIRANJE I PROGRAMSKE JEZIKE (IV RAZRED)</b>
<b>Tema 1: APSTRAKTNE STRUKTURE PODATAKA</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Objasni strukturu olančane liste Primijeni osnovne operacije nad listom
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
Objasni strukturu stoga (stack) Primijeni osnovne operacije nad stogom Izračuna vrijednost aritmetičkog izraza zadatog u infix, postfix ili prefix notaciji Objasni strukturu reda (queue) Primijeni osnovne operacije nad redom Opiše pojmove vezane za strukturu drvo (tree)
<b>Napredni zahtjevi</b> Učenik/učenica treba da:
Primijeni algoritme obilaska drveta (preorder, inorder, postorder) na konkretnom primjeru drveta. Primjenjuju algoritme za dodavanje i brisanje elementa u drvetu Implementiraju programe koji koriste apstraktne strukture podataka.
<b>Tema 2: PROGRAMI SA GRAFIČKIM KORISNIČKIM INTERFEJSOM</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Nabroji osnovne klase za kreiranje grafičkog korisničkog interfejsa Opiše osnovne klase za kreiranje grafičkog korisničkog interfejsa Opiše postupak dodavanja događaja na dugmetu na grafičkom korisničkom interfejsu Demonstrira postupak dodavanja događaja na dugmetu na grafičkom korisničkom interfejsu
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
Objasni postupak uzimanja unesenih podataka iz polja za unos na grafičkom korisničkom interfejsu Napravi aplikaciju sa nekoliko osnovnih elemenata grafičkog korisničkog interfejsa i nekoliko događaja Primijeni klasu za kreiranja liste elemenata na grafičkom korisničkom interfejsu Diskutuje o osnovnim principima kreiranja GUI Primjenjuje postupak uzimanja unesenih podataka iz polja za unos na grafičkom korisničkom interfejsu
<b>Napredni zahtjevi</b> Učenik/učenica treba da:
Kreira GUI pri konkretnim zahtjevima

<b>Tema 3: BAZE PODATAKA</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Objasni osnovne pojmove vezane za baze podataka (entitet, atribut, model, primarni i strani ključ,...) Objasni tipove veza među entitetima Prepozna tipove veza među entitetima Kreira tabelu sa zadatim atributima, tipovima atributa i primarnim ključem
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
Napiše SQL upit za dodavanje podataka u zadatu tabelu Napiše SQL upit za izmjenu određenih podataka unutar tabele Napiše SQL upit za brisanje podataka iz tabele prema zadatom kriterijumu Napiše SQL upit za selekciju podataka iz jedne ili više tabela prema zadatom kriterijumu Skicira model Entiteti-Veze Prevode E-V model u ekvivalentan relacioni model
<b>Napredni zahtjevi</b> Učenik/učenica treba da:
Osmisli jednostavnu bazu podataka sa potrebnim entitetima, atributima, primarnim i stranim ključevima,... za zadati problem Integriše bazu podataka sa programskim jezikom.
<b>Tema 4: ALGORITMI NA GRAFOVIMA</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Pojasni pojam grafa Nabroji osnovne osobine grafova Objasni osnovne osobine grafova Defniše pojmove graf, čvor, grana, stepen čvora, put u grafu Pokazuje na skici grafa elemente grafa Prepozna probleme koji se mogu svesti na graf
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
Primjenjuje različite metode obilaska grafa Određuje razapinjuće drvo grafa Izračuna najkraći put između dva čvora grafa Implementira programe koji koriste grafove. Kombinuje različite strukture podataka za predstavljanje grafa Crta razapinjuće drvo datog grafa
<b>Napredni zahtjevi</b> Učenik/učenica treba da:
Piše programe koji koriste obilazak grafa u širinu (BFS) i dubinu (DFS) Piše program koji nalazi minimalno drvo razapinjanja grafa Piše program koji nalazi najkraći put u grafu.

<b>Tema 5: VIŠENITNO PROGRAMIRANJE</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Identifikuje potrebu za paralelnim izvođenjem dva ili više dijelova programa istovremeno Objasni pojam niti (thread-a) Objasniti osnovna svojstva i metode klasa za rad sa nitima Uočava na primjerima potrebu za paralelnim izvođenjem dvaju ili više dijelova programa istovremeno
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
Idejno osmisli višeprosorsko rješenje jednostavnog problema Kreira program u kojem se uz glavni program pokreće još jedna ili više niti Uoči vremensku razliku u izvođenju zadanog algoritma realizovanog u obliku klasičnog i višenitnog programa
<b>Napredni zahtjevi</b> Učenik/učenica treba da:
Kreira višeprosorsko rješenje složenijeg problema Mjeri vremensku razliku u izvođenju zadanog algoritma realizovanog u obliku klasičnog i višenitnog programa
<b>Tema 6: KRIPTOGRAFIJA</b>
<b>Minimalni zahtjevi</b> Učenik/učenica treba da:
Objasni osnovne pojmove vezane za kriptografiju (otvoreni tekst, kriptovanje, kriptovani tekst...) Opiše postupak kriptovanje s javnim ključem Opiše digitalni potpis.
<b>Osnovni zahtjevi</b> Učenik/učenica treba da:
Prepozna RSA algoritam kriptovanje Prepozna Cezarovo kriptovanje Prepozna kriptovanje sa pomakom Prepozna osnovne operacije u grupama Zn Prepozna Vigenérovu kriptovanje Prepozna transpozicijsko kriptovanje
<b>Napredni zahtjevi</b> Učenik/učenica treba da:
Primijeni RSA algoritam kriptovanje Primijeni Cezarovo kriptovanje Primijeni kriptovanje sa pomakom Primijeni osnovne operacije u grupama Zn Primijeni Vigenérovu kriptovanje Primijeni transpozicijsko kriptovanje

**Tema 7: MREŽNO PROGRAMIRANJE****Minimalni zahtjevi**

Učenik/učenica treba da:

**Objasni osnovne pojmove vezane uz mrežnu komunikaciju (osnovni protokoli, elementi paketa, portovi...)****Osnovni zahtjevi**

Učenik/učenica treba da:

**Objasni pojam socketa****Nabraja osnove naredbe/klase jezika za mrežno programiranje****Koristi osnovne metode klasa za mrežno programiranje****Napiše jednostavnu klijent/server aplikaciju.****Napredni zahtjevi**

Učenik/učenica treba da:

**Kreira aplikaciju klijent/server višeg stepena složenosti**

# **KATALOG ZNANJA**

**ALGORITMI I PROGRAMIRANJE**



## **Tekst koji slijedi predstavlja dio smjernica preuzetih iz javno važećeg obrazovnog programa za matematičku gimnaziju**

Osnovni cilj ovog predmeta je da učenici ovladaju tehnikama algoritamskog pristupa rješavanja problema i osnovnim principima programiranja kroz programski jezik JAVA. Algoritamski pristup rješavanju problema razvija analitičke sposobnosti učenika i upoznaje ih sa principom dekompozicije problema na potprobleme. Programski jezik JAVA je najpopularniji programski jezik za učenje programiranja. Kroz učenje ovog jezika vrši se implementacija naučenih algoritama. Učenici se upoznaju i sa objektno-orijentisanom paradigmom kreiranja softvera i principima pravilnog kreiranja softvera (tzv. softverskog inženjstva). Kroz programiranje učenici mogu da se upoznaju sa osnovama razvoju grafičkog korisničkog interfejsa (GUI). Jezik Java može se koristiti i za veb-programiranje.

Poznavanje osnovnih principa programiranja omogućava učenicima da primjenom računara rješavaju probleme za koje ne postoji adekvatan softver kao i da razvijaju aplikacije koje odgovaraju konkretnim korisničkim zahtjevima. Ovaj predmet predstavlja osnovu za razvijanje naprednih programerskih vještina i uvod u projektovanje softvera.

### **Načini provjeravanja znanja i ocjenjivanja**

Učenici se ocjenjuju kroz izradu određenih vrsta algoritama i programa na zadatu temu. Za naprednije učenike može se zadati složeniji algoritam i program. Moguće je i na početku godine zadati jedan veći projekat koji postupno u toku godine rješavati (npr. školska biblioteka). Pored ovog, moguće je zadati kvizove sa pitanjima koja provjeravaju da li su učenici ovladali pojedinim djelovima predviđenog materijala. Neke od tema su predložene kao neobavezne. Neobavezne teme su uglavnom one koje se najčešće pojavljuju na takmičenjima iz programiranja i one trebaju biti obrađene posebno sa učenicima koji se takmiče. Nastavnik može izabrati samo neke od tih tema ili čak djelove tih tema.

<b>Katalog znanja za nastavni predmet ALGORITMI I PROGRAMIRANJE</b>
<b>Minimalni zahtjevi</b>
Učenik/učenica treba da:
Pojašnjava pojam algoritma Nabraja osnovne elemente algoritamske šeme Nabraja i pojašnjava osobine algoritama (prostorna i vremenska složenost, determinizam, diskretnost). Provjerava ispravnosti algoritma. Razlikuje algoritam i program Nabraja programske paradigme. Raspoznaje programske jezike po paradigmama Zna ulogu prevodioca Razlikuje kompajler i interpreter Navodi primjere kompajlera i interpretera Instalira JAVA prevodioc i editor Podešava radno okruženje (CLASSPATH,...)
Pojašnjava pojam objekta, reference na objekat, klase i poruke. Koristi editor.

Opisuje format JAVA programa.  
Otvora editor, kreira program i snima izmjene  
Kreira elementarni JAVA program (System.out.println).  
Unosi komentare  
Koristi metode objekta PrintStream.  
Kreira objekat u jeziku JAVA  
Zadaje format programa

---

Pojašnjava pojam identifikatora i prepoznaje ih u jeziku JAVA.  
Razlikuje tipove identifikatora i uočava primitivne tipove u JAVI  
Prepoznaje rezervisane riječi.  
Razlikuje Ime.java fajl i Ime.class fajl.  
Pojašnjava šta je Java Byte Code  
Deklariše promjenljivu  
Dodjeljuje vrijednost promjenljivoj  
Kreira različite identifikatore  
Kreira JAVA Bajt Kod i pregled tog koda  
Pronalazi Ime.java fajla i Ime.class fajla poslije prevođenja programa  
Upotrebljava literale, konstante i promjenljive u jednostavnim programima  
Dodjeljuje vrijednosti promjenljivim

---

Kreira objekte putem konstruktora.  
Kreira objekte na primjeru klase String  
Upotrebljava metode FileOutputStream klase  
Zna svojstva i metode PrintStream objekta  
Kreira više konstruktora za klasu  
Otvora i zatvara datoteke (fajla)  
Upisuje podatke u fajl

---

Koristi metode klase FileInputStream, BufferedInputStream, InputStreamReader i BufferedReader  
Shvata razliku između različitih klasa za pristup datotekama (fajlovima)

---

Definiše klasu  
Kreira korisničke klase  
Formira konstruktore za klasu  
Kreira metode klase  
Kontroliše pristup (public, private).  
Koristi "this"  
Kreira Get i Set metode  
Pojašnjava ulogu metoda Main  
Pojašnjava šta su static metodi  
Prepoznaje na primjeru nasljeđivanje  
Projektuje i implementira naslijeđenu klasu  
Prepoznaje problema koje mogu nastati pri nasljeđivanju  
Koristi Math klasu

Razumije uslovno izvršavanje naredbi  
Koristi If naredbu i Switch naredbu  
Shvata princip rada petlje  
Koristi While i For petlju  
Kreira programe sa IF i Switch naredbom  
Kreira programe sa while naredbom  
Kreira programe sa for naredbom

### Osnovni zahtjevi

Učenik/učenica treba da:

Kreira linijske algoritamske šeme  
Kreira razgranate algoritamske šeme  
Kreira ciklične algoritamske šeme  
Predstavlja algoritme pseudokodom  
Izračunava i provjerava složenost algoritma  
Empirijski i teorijski provjerava ispravnost algoritma

Prepoznaje i koristi metode klase String.  
Koristi Time klasu  
Shvata pojam signature  
Na primjeru podstringa koristi overloading  
Shvata princip kaskadnog slanja poruka.  
Kreira programe koji koriste stringove (jednakost stringova, podstringovi, izvalačenje karaktera, transformacije,...)  
Kreira programe koji koriste overloading  
Kreira programe koji koriste kaskadno slanje poruka

Kreira program koje koristi klase FileInputStream,  
BufferedInputStream  
Kreira program koji koristi klase InputStreamReader i BufferedReader

Razumije pojam niza  
Kreira nizove primitivnih tipova  
Kreira nizove objekata  
Shvata pojam višedimenzionalnog niza  
Kreira višedimenzionalni niz  
Kreira programe sa jednodimenzionalnim nizovima (unošenje elemenata, štampanje elemenata, maksimum i minimum, traženje u nizu)  
Kreira programe sa višedimenzionalnim nizovima (matrice)

prepoznaje i koristi metode Vector klase  
prepoznaje i koristi metode Set klase  
razlikuje semantičke sintaksne greške  
uklanja greške u programu (debugging)  
Projektuje program koji koriste Vector klasu  
Kreira program koji koriste klasu Set.

Diskutuje o prednostima i manama klasa Vector i Set.

Razumije principe grafičkog interfejsa (GUI)  
Poznaje osnovne elemente za kreiranje interfejsa (panel, dugme, check-box, option, text-box,...)  
Shvata princip događaja u JAVI (event)  
Implementira događaje za komponente  
Prepoznaje mogućnost upotrebe Thread-ova  
Implementira jednostavnije programe sa thread-ovima  
Kreira programa sa elementima grafičkog interfejsa  
Kreira događaja za pojedine komponente interfejsa  
Kreira programe sa thread-ovima

### **Napredni zahtjevi**

Učenik/učenica treba da:

Kreira applete  
Uključuje applete u veb-stranicu  
Razumije rekurziju kao princip rješavanja problema  
Implementira rekurzivne programe  
Koristi metode grafičkih biblioteka jezika JAVA  
Implementira programe sa grafikom  
Kreira programe koji koriste backtracking (razmještaj dama na šahovskoj tabli tako da se ne napadaju, obilazak šahovske table skakačem, permutacije, obilazak lavirinta, igre,...)  
Kreira programe koji koriste dinamičko programiranje (obilazak tabele po nekim pravilima, najduži zajednički podniz, množenje matrica, ...)  
Kreira programe koji koriste grafove  
Kreira pakete  
Kreira JAR fajl

**KATALOG  
ZNAJJA IZ  
RAČUNARSKIH I  
VEB  
PREZENTACIJA**

## Načini provjeravanja znanja i ocjenjivanja

Učenici/e se ocjenjuju kroz izradu prezentacije na zadatu temu (u dogovoru sa nastavnicima/ama drugih predmeta), koju moraju pripremiti primjenom programa za pripremu prezentacija, uz precizno definisane načine formatiranja dokumenta. Na primjer, učenik/ca je dužan/na da pripremi nekoliko veb-stranica na temu algoritma određivanja najmanjeg elementa niza. Pored ovog, moguće je zadati kvizove sa pitanjima koja provjeravaju da li su učenici/e ovladali/e pojedinim dijelovima XHTML-a i CSS-a.

## STANDARDI ZNANJA

Učenik/ca treba da je u stanju da:

- kreira HTML dokument primjenom neke od aplikacija MS Office-a;
- kreira jednostavnu veb-stranicu i razumije njene osnovne elemente;
- zna da koristi različite tekstualne elemente, tabele i frejmove u HTML dokumentu;
- umije da koristi osnovne elemente za dodavanje stila dokumentima;
- radi sa osnovnim multimedijalnim elementima u prezentacijama;
- vlada načinima povezivanja dokumenata;
- zna da dodaje tabele, liste i slike u HTML dokument.