

KATALOG ZNANJA

1. Ime modula: OSNOVE PROGRAMIRANJA

2. Usmerjevalni cilji

Dijak:

- razume pravilen zapis programa, zapis algoritma, gradnike algoritmov in lastnosti algoritma,
- razvija algoritmično mišljenje,
- uporablja različne načine zapisov algoritmov: besedilni opis, psevdokodo in grafični zapis (npr. diagram poteka ...),
- piše algoritme z upoštevanjem sintakse in semantike vsaj enega splošno namenskega programskega jezika,
- uporabi več modulov v programu in presodi o smiselnosti lastne knjižnice, ki jo zna implementirati,
- testira programe in uporabi tehnike razhroščevanja programov,
- pozna orodja za izdelavo programov in jih izbere v skladu s problemom,
- dokumentira in pripravlja navodila ter predstavi izdelan program,
- upošteva pravila o varnosti in zdravju pri delu ter varovanju okolja,
- uspešno se sporazumeva, komunicira in se dogovarja za delo v delovni skupini,
- razvija čut odgovornosti,
- sledi splošni strokovni literaturi (Slo-Tech, IEEE Spectrum, Communication ACM, spletni časopisi ...),
- sledi razvoju računalništva in informatike na svojem delovnem področju.

3. Poklicne kompetence

1. Uporabljanje osnovne funkcionalnosti orodij za razvoj programske opreme
2. Programiranje v vsaj enem splošno namenskem programskem jeziku
3. Uporabljanje odprtokodnih rešitev za izdelavo preproste programske opreme
4. Uporabljanje običajne oblike zapisov v dokumentaciji

4. Operativni cilji:

1. Uporabljanje osnovne funkcionalnosti orodij za razvoj programske opreme.

Informativni cilji	Formativni cilji
Dijak: <ul style="list-style-type: none">● opiše pojem programskega jezika in programa,● našteje in opredeli vrste programskih	Dijak: <ul style="list-style-type: none">● določi vlogo prevajalnika in tolmača,● razloži postopek prevajanja in tolmačenja,

<p>jezikov,</p> <ul style="list-style-type: none"> ● spozna in se nauči uporabljati programski paket za pisanje, prevajanje, popravljanje in dokumentiranje programov. 	<ul style="list-style-type: none"> ● uporabi grafično okolje programskega paketa, ● uporabi urejevalnik, prevajalnik, razhroščevalnik, povezovalnik in pomoč.
---	---

2. Programiranje v vsaj enem splošno namenskem programskem jeziku

Informativni cilji	Formativni cilji
<p>Dijak:</p> <ul style="list-style-type: none"> ● formulira pojem algoritma in našteje obvezne in želene lastnosti algoritma, ● našteje osnovne elemente algoritma, ● opiše osnovno zgradbo algoritma, ● razlikuje gradnike algoritma, ● razloži strukturo programa, ● razlikuje med enostavnimi in sestavljenimi podatkovnimi tipi, ● spozna načine pridobivanja podatkov iz standardnega vhoda in zapis podatkov na standardni izhod, ● našteje aritmetične, logične in bitne operatorje in jih razvrsti po prioriteti, ● našteje krmilne stavke, ● navede vrste zank, ● razvrsti zanke glede na značilnosti delovanja, ● pozna načine uporabe vnaprej napisanih podprogramov za delo z numeričnimi in besedilnimi podatki. 	<p>Dijak:</p> <ul style="list-style-type: none"> ● uporabi različne načine podajanja algoritma, ● zapiše algoritem, ● sledi izvajanju algoritma za dani primer, ● deklarira spremenljivke in konstante enostavnih podatkovnih tipov, ● inicializira spremenljivke, ● uporabi prireditveni stavek, ● oblikuje izpis podatkov, ● uporabi krmilne stavke pri reševanju problemov, ● oceni primernost izbire določene zanke pri posameznem primeru in jo uporabi, ● uporabi načine pridobivanja podatkov iz standardnega vhoda in zapis podatkov na standardni izhod, ● piše preproste algoritme za delo z numeričnimi in besedilnimi podatki.

3. Uporabljanje odprtokodnih rešitev za izdelavo preproste programske opreme

Informativni cilji	Formativni cilji
<p>Dijak:</p> <ul style="list-style-type: none"> ● pozna načine uporabe vnaprej napisanih podprogramov za delo z numeričnimi in besedilnimi podatki, ● formulira strukturo podprograma, ● opiše možnosti za prenos parametrov, ● pozna načine uporabe vnaprej napisanih podprogramov za delo z numeričnimi in besedilnimi podatki, ● razume zgradbo tabelaričnih podatkovnih tipov, 	<p>Dijak:</p> <ul style="list-style-type: none"> ● uporablja podprograme navedene v dokumentaciji izbranega programskega jezika, ● uporablja vnaprej napisane podprograme za delo z numeričnimi in besedilnimi podatki, ● implementira lastne podprograme za delo z numeričnimi in besedilnimi podatki, ● razčleni zahtevnejši program v

<ul style="list-style-type: none"> ● razlikuje med strukturnim in objektnim programiranjem, ● opredeli razliko med strukturiranim in objektnim programiranjem, ● pozna temeljne značilnosti objektnega programiranja (razred, objekt, lastnosti, metode, skrivanje podatkov, dedovanje, polimorfizem), ● razloži prednosti in načine skrivanja podatkov, ● razloži različne vrste dedovanja, ● razloži polimorfno obnašanje objektov. 	<p>manjše module/podprograme,</p> <ul style="list-style-type: none"> ● deklarira in uporabi tabelarične spremenljivke, ● implementira lastne razrede in jih uporabi v programih, ● implementira različne vrste konstruktorjev in destruktorjev, ● uporabi statične lastnosti in metode, ● uporabi prijateljske funkcije, ● prekriva operatorje, ● uporabi prednosti in načine skrivanja podatkov, ● uporabi različne vrste dedovanja, ● uporabi polimorfno obnašanje objektov.
---	---

4. Uporabljanje običajne oblike zapisov v dokumentaciji

Informativni cilji	Formativni cilji
<p>Dijak:</p> <ul style="list-style-type: none"> ● opiše pomen in načine testiranja programskega izdelka, ● spozna ustrezen način dokumentiranja programskih izdelkov. 	<p>Dijak:</p> <ul style="list-style-type: none"> ● za podano programsko rešitev zbere ustrezne testne podatke, ● ugotovi in odpravi vzrok napake, ● izdelava načrt za testiranje programskega izdelka, ● sestavi obrazec, na katerem dobi opombe (obvestila) o nepravilnosti v delovanju programa, ● analizira in razvrsti pripombe in se odloča o ukrepih, ● sprejema in obravnava poročila testnih uporabnikov, ● napiše programsko dokumentacijo.