

KATALOG ZNANJA

1. IME PREDMETA

PROGRAMIRANJE 1

2. CILJI PREDMETA

Splošni cilji predmeta so:

- ozavestiti pomen nenehnega izpopolnjevanja znanja in veščin;
- ozavestiti pomen sprotne spremljanja novosti na strokovnem področju;
- razviti kritično vrednotenje lastnega dela in dela drugih;
- razviti sposobnost samostojnega reševanja problemov in načrtovanja, organizacije dela ter sodelovanja v timu;
- ozavestiti pomen pozitivnega odnosa do sodelavcev, dela in širše okolice.

Specifično strokovno usmerjeni cilji so:

- razumeti delovanje, izdelavo in vzdrževanje programskih rešitev;
- načrtovati programske rešitve;
- preizkušati delovanje sklopov ali celot programskih rešitev;
- dokumentirati programske rešitve;
- svetovati in sodelovati pri načrtovanju razvoja programske opreme.

3. PREDMETNO SPECIFIČNE KOMPETENCE

Pri predmetu si študenti poleg generičnih pridobijo naslednje kompetence:

1. spremljanje postopka razvoja programa;
2. izdelovanje strukturiranih konzolnih programov;
3. razvijanje objektno orientiranih programov;
4. analiziranje vhodno/izhodnih operacij;
5. upravljanje izjem;
6. izdelovanje okenskih aplikacij.

4. OPERATIVNI CILJI

INFORMATIVNI CILJI	FORMATIVNI CILJI
Študent:	Študent:
1. Spremljanje postopka razvoja programa	
<ul style="list-style-type: none">• razlikuje med različnimi programskimi jeziki;• loči med podatkovnimi tipi;• opiše spremenljivke in konstante;• pojasni sintakso in semantiko;• opredeli koncept algoritma;• razloži delovanje programa• opredeli težavo in opiše strategijo reševanja.	<ul style="list-style-type: none">• sestavi, prevede in izvede preproste programe v izbranem razvojnem okolju;• prebere in izpiše vrednosti spremenljivk v konzolnem oknu;• izdelava preproste algoritme;• izvede zagon in ustavitvev programa;• uporabi integrirana razvojna okolja.
2. Izdelovanje strukturiranih konzolnih programov	

<ul style="list-style-type: none"> • pojasni zgradbo programa; • razlikuje operatorje, stavke in zanke; • razloži uporabo krmilnih stavkov; • opiše namen uporabe polja/preglednice; • opredeli eno- in večdimenzionalne tabele; • opiše nize; • pojasni delovanje izsekov; programske kode. 	<ul style="list-style-type: none"> • deklarira in inicializira spremenljivke; • izvede preproste vhode in izhode podatkov; • izvede aritmetično-logične operacije; • izdelava programe za izvajanje izračunov z uporabo vhodno/izhodnih operacij; • uporabi krmilne stavke (odločitve, zanke); • izdelava metode in preveri delovanje metod; • uporabi polja/tabele spremenljivk.
3. Razvijanje objektno orientiranih programov	
<ul style="list-style-type: none"> • razloži prednosti objektno usmerjenega načina programiranja; • opredeli ustvarjanje objekta na podlagi razredov; • opiše delovanje; • pojasni omejevanje dostopov; • opredeli delovanje virtualnih metod; • opiše rabo abstraktnih razredov in vmesnikov; • utemelji polimorfizem in njegovo uporabnost; • opiše razlike med abstraktnim razredom in vmesnikom; • navede primere uporabe polimorfizma. 	<ul style="list-style-type: none"> • opredeli razrede za predloge objektov; • napiše preproste razrede; • iz razredov izpelje objekte; • uporabi konstruktorje razredov in napiše destruktor; • objektne in razredne spremenljivke skriva v opredelitvi razreda, dostop do njih omogoči prek javnih metod; • izdelava hierarhijo razredov in iz načrta napiše programsko kodo.
4. Analiziranje vhodno/izhodnih operacij	
<ul style="list-style-type: none"> • opiše vhodno-izhodne tokove; • utemelji shranjevanje podatkov na datotečni sistem; • razlikuje jezikovno neodvisne formate za izmenjavo podatkov. 	<ul style="list-style-type: none"> • napiše metode za branje iz dokumentov in pisanje v dokumente; • uporabi razrede za delo z datotekami; • napiše metode za serializacijo in deserializacijo vrednosti objektov.
5. Upravljanje izjem	
<ul style="list-style-type: none"> • pojasni obravnavanje izjem; • loči proženje in prestrezanje izjem; • poišče logične napake z razhroščevalnikom; • razlikuje logične napake in napake med prevajanjem in izvajanjem. 	<ul style="list-style-type: none"> • preizkusi poljubni program in izdelava poročilo o izjemah in logičnih napakah, ki jih je v programu odkril; • napisanemu programu doda obravnavo izjem; • uporabi razrede izjem razvojnega okolja in napiše lastne razrede izjem.
6. Izdelovanje okenskih aplikacij	
<ul style="list-style-type: none"> • pojasni dogodkovni model; • opiše izdelavo grafičnih vmesnikov programsko in v razvojnem okolju; • utemelji uporabo obstoječih kontrolnikov; • pojasni kreiranje lastnih kontrolnikov in njihovo uporabo. 	<ul style="list-style-type: none"> • izdelava program, v katerem loči logiko programa od uporabniškega vmesnika; • za reševanje istega problema uporabi konzolni in grafični vmesnik; • grafični vmesnik izdelava v razvojnem okolju; • izdelava lastni grafični kontrolnik in ga uporabi.

5. OBVEZNOSTI ŠTUDENTOV IN POSEBNOSTI V IZVEDBI

Število kontaktnih ur: 84 ur (36 ur predavanj, 48 ur laboratorijskih vaj).

Število ur samostojnega dela: 116 ur (54 ur študij literature, 30 ur vaj, 42 ur projektna naloga).