

Pučko otvoreno učilište Procesor, Rimska 15, Sisak

**Program obrazovanja
za stjecanje mikro kvalifikacije
objektno orijentirano programiranje**

Sisak, 15.09.2022

1. OPĆI DIO

OPĆE INFORMACIJE O PROGRAMU OBRAZOVANJA ZA STJECANJE MIKROKVALIFIKACIJE		
Sektor	Elektrotehnika i računarstvo	
Naziv programa	Program obrazovanja za stjecanje mikrokvalifikacije objektno orijentirano programiranje	
Vrsta programa	Usavršavanje	
Predlagatelj	Naziv ustanove	Učilište Procesor
	Adresa	Rimska 15, 44000 Sisak
Razina kvalifikacije/skupa/ova ishoda učenja prema HKO-u	SIU 1: OSNOVE OBJEKTNO ORIJENTIRANOG PROGRAMIRANJA (razina 4) SIU 2: OSNOVE IZRADE STOLNIH APLIKACIJA (razina 4) SIU 3: NAPREDNE TEHNIKE IZRADE STOLNIH APLIKACIJA (razina 4)	
Obujam u bodovima (CSVET)	9 CSVET SIU 1: OSNOVE OBJEKTNO ORIJENTIRANOG PROGRAMIRANJA (3 CSVET) SIU 2: OSNOVE IZRADE STOLNIH APLIKACIJA (3 CSVET) SIU 3: NAPREDNE TEHNIKE IZRADE STOLNIH APLIKACIJA (3 CSVET)	
Dokumenti na temelju kojih je izrađen program obrazovanja za stjecanje kvalifikacija/skupova ishoda učenja (mikrokvalifikacija)		
Popis standarda zanimanja/skupova kompetencija i datum/i njegove/njihove valjanosti u Registru HKO-a	Popis standarda kvalifikacija/skupova ishoda učenja i datum/i njegove/njihove valjanosti u Registru HKO-a	Sektorski kurikulum
SZ Tehničar za računarstvo SKOMP 1: Rješavanje problema pomoću algoritama https://hko.srce.hr/registar/skup-kompetencija/detalji/157 SKOMP2: Razvoj programskih sustava i baza podataka https://hko.srce.hr/registar/skup-kompetencija/detalji/160 31.12.2022.	SIU Osnove objektno orijentiranog programiranja SIU Osnove izrade stolnih aplikacija SIU Napredne tehnike izrade stolnih aplikacija 15.12.2024.	
Uvjeti za upis u program	Cjelovita kvalifikacija minimalno na razini 4.1	
Uvjeti stjecanja programa (završetka programa)	<ul style="list-style-type: none"> Stečenih 9 CSVET bodova 	

	<ul style="list-style-type: none"> • Uspješna završna provjera stečenih znanja usmenim i/ili pisanim provjerama te vještina polaznika kroz projektne i problemske zadatke, a temeljem unaprijed određenih kriterija vrednovanja postignuća. • Na završnoj provjeri vodi se zapisnik i provodi ju tročlano povjerenstvo. • Svakom polazniku nakon uspješno završene završne provjere izdaje se Uvjerenje o usavršavanju za stjecanje mikrokvalifikacije objektno orijentirano programiranje
Trajanje i načini izvođenja nastave	<p>Program obrazovanja za stjecanje mikrokvalifikacije objektno orijentirano programiranje provodi se redovitom nastavom u trajanju od 225 sati, uz mogućnost izvođenja teorijskog dijela programa na daljinu u realnom vremenu hibridnim modelom podučavanja koristeći kolaboracijske alate sa mentoriranjem. Koristi se flipped classroom metodologija</p> <p>Ishodi učenja ostvaruju se dijelom vođenim procesom učenja i poučavanja u trajanju od 45 sati, dijelom učenjem temeljenom na radu u trajanju od 90 sati, a dijelom samostalnim aktivnostima polaznika u trajanju od 90 sati.</p> <p>Učenje temeljeno na radu obuhvaća situacijsko učenje i izvršenje konkretnih radnih zadataka u stvarnim i/ili simuliranim uvjetima.</p>
Horizontalna prohodnost	Prema kvalifikaciji tehničar za računalstvo
Vertikalna prohodnost	
Materijalni uvjeti i okruženje za učenje koji su potrebni za izvedbu programa	Specijalizirana učionica opremljena računalom koje ima pristup internetu s instaliranom potrebnom programskom potporom za svakoga polaznika, poslužitelj s poslužiteljskim operacijskim sustavom, neprekidno napajanje.
Kompetencije koje se programom stječu	
	<ol style="list-style-type: none"> 1. odabrati odgovarajuće standardizirane algoritme za rješavanje jednostavnijih programskih i/ili sklopovskih problema 2. osmišljavati vlastite algoritme za rješavanje jednostavnijih programskih i/ili sklopovskih problema 3. odabrati programsku paradigmu za izradu algoritama 4. izraditi jednostavnije osmišljene i standardizirane algoritme u odabranom programskom jeziku 5. prilagoditi razvojno okruženje 6. izraditi korisničko sučelje manje razine složenosti 7. testirati komponente, međusobnu povezanost i funkcionalnost jednostavnog programskog sustava 8. izraditi elemente baze podataka 9. izraditi i povezati jednostavne poslovne procese unutar programskog sustava
Preporučeni načini praćenja kvalitete i uspješnosti izvedbe programa	<p>U procesu praćenja kvalitete i uspješnosti izvedbe programa obrazovanja primjenjuju se sljedeće aktivnosti:</p> <ul style="list-style-type: none"> • provodi se istraživanje i anonimno anketiranje polaznika o izvođenju nastave, literaturi i resursima za učenje, strategijama podrške polaznicima, izvođenju i unapređenju procesa učenja i poučavanja, radnom opterećenju polaznika (CSVET), provjerama znanja te komunikaciji s nastavnicima • provodi se istraživanje i anketiranje nastavnika o istim pitanjima navedenim u prethodnoj stavci • provodi se analiza uspjeha, transparentnosti i objektivnosti provjera i ostvarenosti ishoda učenja • provodi se analiza materijalnih i kadrovskih uvjeta potrebnih za izvođenje procesa učenja i poučavanja. <p>Dobivenim rezultatima anketa dobiva se pregled uspješnosti izvedbe programa, kao i procjena kvalitete nastavničkog rada.</p>

	Postupci vrednovanja usmjereni su na praćenje i provjeru postignuća prema ishodima učenja. Ono se provodi usmenim i pisanim provjerama znanja te provjerama stečenih vještina polaznika projektnim i problemskim zadacima te radnim situacijama, a temeljem unaprijed određenih kriterija vrednovanja postignuća.
Datum revizije programa	

2. MODULI I SKUPOVI ISHODA UČENJA

Redni broj	NAZIV MODULA	POPIS SKUPOVA ISHODA UČENJA	Razina	Obujam CSVET	Broj sati			
					VPUP	UTR	SAP	UKUPNO
1.	Objektno orijentirano programiranje	Osnove objektno orijentiranog programiranja	4	3	15	30	30	75
		Osnove izrade stolnih aplikacija	4	3	15	30	30	75
		Napredne tehnike izrade stolnih aplikacija	4	3	15	30	30	75
Ukupno:				9	45	90	90	225

VPUP – vođeni proces učenja i poučavanja

UTR – učenje temeljeno na radu

SAP – samostalne aktivnosti polaznika

3. RAZRADA MODULA I SKUPOVA ISHODA UČENJA

NAZIV MODULA	OBJEKTNO ORIJENTIRANO PROGRAMIRANJE		
Šifra modula			
Kvalifikacije nastavnika koji sudjeluju u realizaciji modula	<ul style="list-style-type: none"> najmanje razina 6 HKO-a – 180 ECTS bodova (preddiplomski sveučilišni studij, preddiplomski stručni studij) odgovarajućeg profila. 		
Obujam modula (CSVET)	9 CSVET		
Načini stjecanja ishoda učenja (od – do, postotak)	Vođeni proces učenja i poučavanja	Oblici učenja temeljenog na radu	Samostalne aktivnosti polaznika
	45 (20 %)	90 (40 %)	90 (40 %)
Status modula (obvezni/izborni)	obvezni		
Cilj (opis) modula	Cilj modula je polaznicima omogućiti stjecanje kompetencija osnovnih načela objektno orijentiranog programiranja (enkapsulacija, nasljeđivanje, kreiranje instanci, obrada grešaka, agregiranje), osnovnih (prozor stolne aplikacije, pretplata i obrađivanje događaja, osnovne i kontejnerske kontrole) i naprednih tehnika izrade		

	stolnih aplikacija (validacija korisničkog unosa, prikaz podataka u <i>gridu</i>) rješavanjem problemskih i projektnih zadataka.
Ključni pojmovi	<i>Klasa, konstruktor, svojstvo, metoda, referenca, instanca, inicijalizacija, član klase, enkapsulacija, modifikatori pristupa, nasljeđivanje, događaj, korisničko sučelje, validacija, korisnički unos, modalni prozor, filtriranje, paginacija</i>
Oblici učenja temeljenog na radu	Učenje temeljeno na radu ostvaruje se realiziranjem radnih zadataka koji se mogu simulirati u specijaliziranim učionicama/praktikumima ili u Regionalnim centrima kompetentnosti. Učenje temeljeno na radu implementirano je u obliku primjera, problemskih i projektnih zadataka koji simuliraju stvarne poslovne situacije po zahtjevu klijenta. Polaznici zadatke analiziraju i izrađuju aplikaciju u odabranom programskom jeziku uz prikladna pojednostavljenja. Nastavnik ima ulogu mentora te ih usmjerava kroz proces učenja i rješavanja zadataka kao npr. implementirati konzola aplikaciju uz zadane parametre u odabranome programskom jeziku, dizajnirati korisničko sučelje i koristiti kontejnerske kontrole i sl.
Literatura i specifična nastavna sredstva potrebna za realizaciju modula	<ol style="list-style-type: none"> 1. Herbert Schildt, Java: The Complete Reference, Tenth Edition, McGraw Hill Professional, 2017. 2. Raoul-Gabriel Urma, Mario Fusco, Alan Mycroft, Java 8 & 9 in Action, Pearson Professional, 2018. 3. Walter Savitch, Kenrick Mock, Absolute Java, Global Edition, Pearson Higher Ed, 2015. 4. R. Manger, Strukture podataka i algoritmi, Element

Skup ishoda učenja iz SK-a¹:	OSNOVE OBJEKTNO ORIJENTIRANOG PROGRAMIRANJA
Ishodi učenja	
<ol style="list-style-type: none"> 1. Definirati privatne i javne članove klase uz ispravnu primjenu enkapsulacije 2. Implementirati konstruktor i ostale metode klase 3. Izraditi baznu i nasljeđenu klasu 4. Koristiti instance klase za rješavanje problema 5. Izraditi klasu koja agregira drugu klasu 6. Opisati obradu iznimki za rješavanje problema 	
Dominantan nastavni sustav i opis načina ostvarivanja SIU	
Dominantan nastavni sustav je temeljen na demonstraciji i poučavanju primjerima uz heuristički pristup popraćen aktivnim praktičnim sudjelovanjem polaznika kroz problemsko i projektno poučavanje. Nastava će se provoditi kombinirajući samostalno istraživanje polaznika, rad u parovima, obrnutom učionicom.	
Nastavne cjeline/teme	<ul style="list-style-type: none"> – Klase, konstruktori, metode – Enkapsulacija – Nasljeđivanje – Obrada iznimki – Agregiranje
Načini i primjer vrjednovanja skupa ishoda učenja	
Potrošnja električne energije kod igranja računalnih igara	
Istražiti najnovije računalne komponente (procesor, memorija, grafička kartica, matična ploča, spremnik podataka, napajanje) koje se koriste za igranje računalnih igara. Zapisati naziv, cijenu, potrošnju električne energije te osnovne	

¹Popunjava se onoliko puta koliko je skupova ishoda učenja u modulu.

parametre svake komponente. Istražiti i zapisati trenutnu cijenu električne energije. Zapisani će se podatci koristiti u aplikaciji.

Potrebno je implementirati konzola aplikaciju u odabranom programskom jeziku u kojoj će korisnik unijeti prosječan broj sati rada računala tijekom dana.

Aplikacija treba ispisati

- ukupnu utrošenu električnu energiju u kWh za 30 dana
- ukupan iznos potreban za platiti u kunama za 30 dana
- detalje računala (ispis detalja svih komponenti)

Zahtjevi implementacije:

- potrebno je izraditi slijedeće klase, popratna svojstva i metode:
 - Komponenta - Naziv, Cijena, Snaga i metoda IspisiDetalje()
 - Procesor - Naziv, Cijena, Snaga, Frekvencija, BrojJezgri i metoda IspisiDetalje()
 - Memorija - Naziv, Cijena, Snaga, KoličinaMemorije i metoda IspisiDetalje()
 - GrafičkaKartica - Naziv, Cijena, Snaga, KoličinaMemorije i metoda IspisiDetalje()
 - MatičnaPloča - Naziv, Cijena, Snaga, VrstaProcesora i metoda IspisiDetalje()
 - SpremnikPodataka - Naziv, Cijena, Snaga, KoličinaMemorije i metoda IspisiDetalje()
 - Napajanje - Naziv, Cijena, Snaga, Učinkovitost i metoda IspisiDetalje()
 - Računalo – Naziv, Procesor, Memorija, GrafičkaKartica, MatičnaPloča, SpremnikPodataka, Napajanje i metode IzračunajSnagu() i IspisiDetaljeRačunala()
- Potrebno je prepoznati i implementirati nasljeđivanje.
- Potrebno je prepoznati i implementirati agregiranje.
- Klase Procesor, Memorija, GrafičkaKartica, MatičnaPloča, SpremnikPodataka i Napajanje sadrže metodu IspisiDetalje() koja ispisuje sva svojstva klase. Potrebno je prepoznati i implementirati premošćivanje.
- Izvan klase Računalo ne smije biti dopušten pristup klasama Procesor, Memorija, GrafičkaKartica, MatičnaPloča, SpremnikPodataka i Napajanje. Potrebno je prepoznati i implementirati kontrolu pristupa.
- Klasa Računalo sadrži metodu IzračunajSnagu() koja zbraja snagu svih komponenti. Potrebno je prepoznati i implementirati enkapsulaciju. Metoda IzračunajSnagu() ne smije se moći pozvati izvan klase. Napomena: Svojstvo Snaga klase Napajanje ne koristi se u ovom izračunu.
- Ukoliko je izračunata snaga komponenti računala veća od snage napajanja baca se iznimka sa odgovarajućom porukom. Potrebno je prepoznati i implementirati obradu iznimki te ne prekinuti izvršavanje programa.
- Klasa Računalo sadrži konstruktor koji prima slijedeće parametre:
 - Procesor
 - Memorija
 - GrafičkaKartica
 - MatičnaPloča
 - SpremnikPodataka
 - Napajanje
- Potrebno je kreirati instancu Računalo i navedene komponente te ih proslijediti instanci Računalo preko konstruktora u glavnoj metodi.
- Klasa Računalo sadrži metodu IspisiDetaljeRačunala() koja ispisuje naziv računala, detalje svake komponente i ukupnu snagu računala. Metoda ispisuje i detalje o ukupnoj utrošenoj energiji u kWh za 30 dana te ukupan iznos potreban za platiti u kunama za 30 dana. Napomena: metodi je potrebno proslijediti informaciju o prosječnom broju sati rada računala tijekom dana koja se koristi u proračunu zajedno s cijenom električne energije.
- Pozvati metodu IspisiDetaljeRačunala() u glavnoj metodi. Potrebno je prepoznati i implementirati kontrolu pristupa.

Vrednovanje naučenog

Polaznici će prezentirati način izrade aplikacije, opisati iskustva rada u paru, procijeniti svoj doprinos i suradnički vrednovati rad drugog člana.

Prilagodba iskustava učenja za polaznike/osobe s invaliditetom

(Izraditi način i primjer vrjednovanja skupa ishoda učenja za polaznike/osobe s invaliditetom ako je primjenjivo)

Skup ishoda učenja iz SK-a:	OSNOVE IZRADE STOLNIH APLIKACIJA
Ishodi učenja	
<ol style="list-style-type: none">1. Izraditi prozor stolne aplikacije te podesiti veličinu, položaj, boju, font i ostala osnovna svojstva2. Pretplatiti se i obrađivati osnovne događaje3. Koristiti osnovne kontrole (gumbe, tekstualne kućice, labele, radio gumbe i potvrđne kućice) u izradi sučelja prema korisniku4. Koristiti kontejnerske kontrole za povezivanje ostalih kontrola u smislene cjeline5. Prepoznati elemente objektno orijentiranog programiranja u stolnoj aplikaciji	
Dominantan nastavni sustav i opis načina ostvarivanja SIU	
Dominantan nastavni sustav je učenje temeljeno na radu kroz realne radne situacije, popraćene teorijskim spoznajama, koji se provodi kombinirajući samostalan rad, rad u parovima i projektnu nastavu.	
Nastavne cjeline/teme	<ul style="list-style-type: none">– Izradba forme– Događaji– Jednostavne kontrole korisničkog sučelja– Kontejnerske kontrole korisničkog sučelja
Načini i primjer vrjednovanja skupa ishoda učenja	
<p style="text-align: center;">Izrada kalkulatora</p> <p>Istražiti i zapisati osnovne matematičke funkcije koje bi kalkulator sadržavao. Potrebno je implementirati stolnu aplikaciju u odabranom programskom jeziku u kojoj će korisnik moći:</p> <ul style="list-style-type: none">– Unijeti brojeve– Odabrati matematičke operacije <p>Aplikacija treba ispisati</p> <ul style="list-style-type: none">– Rezultat matematičke operacije– Pregled unesenih brojeva i operacija <p>Zahtjevi implementacije:</p> <ul style="list-style-type: none">– potrebno je odabrati odgovarajuću kontrolu za prikaz prozora stolne aplikacije– dizajnirati korisničko sučelje– koristiti kontejnerske kontrole– pretplatiti se na događaje– osmisliti način prikaza trenutno odabrane funkcije i unesenih brojeva– personalizirati stolnu aplikaciju vlastitim odabirom veličine, boje i fonta prozora stolne aplikacije <p>Uputa: Polaznici rade projektni zadatak samostalno, a vrednuje se funkcionalnost, izgled i prezentacija izrađene aplikacije.</p>	
Prilagodba iskustava učenja za polaznike/osobe s invaliditetom	
<i>(Izraditi način i primjer vrjednovanja skupa ishoda učenja za polaznike/osobe s invaliditetom ako je primjenjivo)</i>	

Skup ishoda učenja iz SK-a:	NAPREDNE TEHNIKE IZRADE STOLNIH APLIKACIJA
Ishodi učenja	
<ol style="list-style-type: none"> 1. Odabrati podatke potrebne za validaciju korisničkog sučelja. 2. Koristiti više prozora na modalni i nedomodalni način 3. Koristiti kontrole s više vrijednosti (liste, padajuće izbornike i slično) u izradi sučelja prema korisniku 4. Prikazati veću količinu podataka u <i>grid</i> kontroli 5. Implementirati izbornike te alatne i statusne trake za poboljšanje korisničkog iskustva 	
Dominantan nastavni sustav i opis načina ostvarivanja SIU	
<p>Dominantan nastavni sustav je učenje temeljeno na radu kroz realne radne situacije i popraćenih teorijskim spoznajama koji se provodi kombinirajući samostalan rad, rad u parovima i projektnu nastavu.</p>	
Nastavne cjeline/teme	<ul style="list-style-type: none"> – Validacija korisničkog unosa – Modalni i nedomodalni prozori – Kontrole s više vrijednosti (ListBox, ComboBox) – Grid kontrola – Izbornici – Alatna i statusna traka
Načini i primjer vrjednovanja skupa ishoda učenja	
<p style="text-align: center;">Implementacija korisničkog sučelja stolne aplikacije za unos i prikaz podataka</p> <p>Izraditi korisničko sučelje stolne aplikacije za unos i prikaz podataka o napretku razvoja aplikacije.</p> <p>Potrebno je implementirati:</p> <ul style="list-style-type: none"> • Glavni prozor aplikacije koji sadrži odabir: <ul style="list-style-type: none"> ○ Zadatak ○ Prijavu greške ○ Prijedlog poboljšanja • Navedeni prozori otvaraju se u nedomodalnom načinu rada iz glavnog prozora • Prozor zadatak sadrži: <ul style="list-style-type: none"> ○ Tekstualnu kućicu za filtriranje po nazivu ○ Prikaz podataka pomoću grid kontrole ○ Akciju dodavanja novog zadatka ○ Pokretanjem dodavanja novog zadatka otvara se modalni prozor koji sadrži slijedeće: <ul style="list-style-type: none"> ▪ Naslov ▪ Status ▪ Korisnik ▪ Područje aplikacije ▪ Detalji ▪ Akcije snimi i odustani • Prozor prijave greške sadrži: <ul style="list-style-type: none"> ○ Tekstualnu kućicu za filtriranje po nazivu ○ Prikaz podataka pomoću grid kontrole ○ Akciju dodavanja nove greške ○ Pokretanjem dodavanja novog zadatka otvara se modalni prozor koji sadrži slijedeće: <ul style="list-style-type: none"> ▪ Naslov ▪ Status ▪ Korisnik ▪ Područje aplikacije ▪ Koraci za reproduciranje ▪ Akcije snimi i odustani 	

- Prozor prijedloga za poboljšanje sadrži:
 - Tekstualnu kućicu za filtriranje po nazivu
 - Prikaz podataka pomoću grid kontrole
 - Akciju dodavanja novog prijedloga za poboljšanje
 - Pokretanjem dodavanja novog zadatka otvara se modalni prozor koji sadrži slijedeće:
 - Naslov
 - Detalji
 - Akcije snimi i odustani

Zahtjevi implementacije:

- Za podatke status i korisnik potrebno je odabrati kontrolu lista ili padajući izbornik te popuniti vrijednostima.
- Pokretanje akcije snimi, nakon ispravne validacije, javlja glavnoj formi poruku o uspješnoj validaciji.
- Potrebno je napraviti validaciju unosa svih podataka na navedenim prozorima te dati korisniku odgovarajuću poruku.
- Grid sa podacima treba sadržavati dinamičke kontrole za brisanje i izmjenu podataka.
- Akcija brisanja podatka u gridu treba prvo prikazati upozorenje korisniku o pokrenutoj radnji. Nakon Potvrde korisnika briše se sadržaj odabranog retka.

Uputa: Polaznici rade u paru na projektnom zadatku, a vrednuje se funkcionalnost, izgled i prezentacija izrađene aplikacije.

Prilagodba iskustava učenja za polaznike/osobe s invaliditetom

(Izraditi način i primjer vrjednovanja skupa ishoda učenja za polaznike/osobe s invaliditetom ako je primjenjivo)

***Napomena:**

Riječi i pojmovni sklopovi koji imaju rodno značenje korišteni u ovom dokumentu (uključujući nazive kvalifikacija, zvanja i zanimanja) odnose se jednako na oba roda (muški i ženski) i na oba broja (jedinu i množinu), bez obzira na to jesu li korišteni u muškom ili ženskom rodu, odnosno u jedini ili množini.

Broj i datum mišljenja na program (popunjava Agencija):

KLASA:	
URBROJ:	
Datum izdavanja mišljenja na program:	