

ICDL Professional RAČUNARSTVO

Syllabus 1.0



Nastavni plan



Svrha

U ovom dokumentu je detaljno opisan nastavni plan za ICDL modul Računarstvo. Nastavni plan kroz ishode opisuje znanja i veštine koje kandidat treba da poseduje da bi položio test za ovaj modul. Nastavni plan predstavlja osnov za teorijski i praktični deo testa za ovaj modul.

Copyright © 2017 - 2019 ICDL Foundation

Sva prava zadržana. Nijedan deo ovog dokumenta, bez izuzetka, se ne sme reprodukovati bez prethodnog odobrenja ICDL Fondacije. Zahtevi za dobijanje odobrenja se dostavljaju Regionalnoj ICDL kancelariji u Beogradu.

Odricanje od odgovornosti

Iako je ICDL Fondacija pažljivo pripremala ovaj dokument, ICDL Fondacija, kao izdavač, i Regionalna ICDL kancelarija u Beogradu, ne garantuju apsolutnu tačnost informacija sadržanih u nastavnom planu niti snose odgovornost za bilo koju grešku, propust, netačnost, gubitke niti štete koje mogu nastati na osnovu sadržanih informacija ili bilo kog uputstva ili saveta iz ovog teksta. ICDL Fondacija zadržava pravo da vrši izmene, po sopstvenom nahođenju u bilo koje vreme bez prethodne najave.

RAČUNARSTVO

Ovim modulom su obuhvaćeni opšti koncepti i veštine koje se odnose na sposobnost korišćenja računarskog razmišljanja i kodiranja za stvaranje jednostavnih računarskih programa.

CILJEVI MODULA

Uspešan kandidat će biti osposobljen da:

- Razume ključne koncepte vezane za računarstvo i tipične aktivnosti uključene u kreiranju programa.
- Razume i koristi tehnike računarskog razmišljanja kao što su dekompozicija, prepoznavanje uzoraka, apstrakcija i algoritmi za analizu problema i razvoj rešenja.
- Piše, testira i modifikuje algoritme za program upotrebom dijagrama toka (flowcharts) i pseudokoda.
- Razume ključne principe i termine povezane sa kodiranjem kao i važnost dobro strukturiranog i dokumentovanog koda.
- Razume i koristi programske konstrukte kao što su varijable, vrste podataka i logika programa.
- Poboljša efikasnost i funkcionalnost koristeći iteraciju, uslovne izjave, procedure i funkcije, kao i događaje i komande u programu.
- Testira i debuguje program, kao i da proveri ispunjenost svih zahteva pre puštanja.

POGLAVLJE	OBLAST	OZNAKA	CILJEVI
1 Termini u računarstvu	<i>1.1 Ključni koncepti</i>	1.1.1	Definisati termin računarstvo.
		1.1.2	Definisati termin računarsko razmišljanje.
		1.1.3	Definisati termin program.
		1.1.4	Definisati termin kod. Razlikovati izvorni kod, mašinski kod.
		1.1.5	Razumeti termine opis programa i specifikacija.
		1.1.6	Prepoznati tipične aktivnosti u kreiranju programa: analiza, dizajn, programiranje, testiranje, poboljšanje.
		1.1.7	Razumeti razliku između formalnog i "prirodnog" jezika.
2 Metode računarskog razmišljanja	<i>2.1 Analiza problema</i>	2.1.1	Navesti tipične metode u računarskom razmišljanju: dekompozicija, prepoznavanje oblika, apstrakcija, algoritmi.
		2.1.2	Upotrebiti dekompoziciju problema za razlaganje podataka, procesa ili kompleksnih problema na manje delove.
		2.1.3	Identifikovati obrasce između malih, dekomponovanih problema.
		2.1.4	Koristiti apstrakciju za filtriranje nepotrebnih detalja u analiziranju problema.

POGLAVLJE	OBLAST	OZNAKA	CILJEVI
		2.1.5	Razumeti upotrebu algoritma u računarskom razmišljanju.
	<i>2.2 Algoritmi</i>	2.2.1	Definisati termin sekvenca u programskoj konstrukciji. Objasniti svrhu sekvenciranja prilikom dizajniranja algoritama.
		2.2.2	Prepoznati moguće metode za problem reprezentacije kao što su: flowcharts, pseudokod.
		2.2.3	Prepoznati flowchart simbole kao što su: pokreni/zaustavi, proces, odluke, ulaz/izlaz, konektor, strelica.
		2.2.4	Objasniti sekvencu operacija predstavljenim flowchart-om, pseudokodom.
		2.2.5	Napisati tačan algoritam zasnovan na opisu korišćenjem tehnika kao što su: flowchart, pseudokod.
		2.2.6	Popraviti greške u algoritmu kao što su: nedostatak programskih elemenata, neispravna sekvenca, neispravni ishod odluke.
3 Početak kodiranja	<i>3.1 Uvod</i>	3.1.1	Opisati karakteristike dobro strukturiranog i dokumentovanog koda kao što su: uvlačenje (indentation), odgovarajući komentari, opisno imenovanje.
		3.1.2	Koristiti osnovne aritmetičke funkcije za izvršavanje proračuna u programu: +, -, /, *.
		3.1.3	Razumeti prioritet operatora i redosled evaluacije u kompleksnim izrazima. Razumeti kako da koristite zagrade za strukturiranje složenih izraza.
		3.1.4	Razumeti termin parametar. Navesti svrhu parametara u programu.
		3.1.5	Definisati termin komentar u programskoj konstrukciji. Navesti svrhu komentara u programu.
		3.1.6	Upotrebljavati komentare u programu.
	<i>3.2 Varijable i tipovi podataka</i>	3.2.1	Definisati termin varijabla u programskoj konstrukciji. Navesti svrhu varijable u programu.
		3.2.2	Definisati i inicijalizirati varijablu.
		3.2.3	Dodeliti vrednost varijabli.
		3.2.4	Koristiti odgovarajuće nazive varijabli u programu za kalkulacije, skladištenje vrednosti.
		3.2.5	Koristiti tipove podataka u programu: string, character, integer, float, Boolean.
		3.2.6	Koristiti agregirane tipove podataka u programu kao što su: array, list, tuple.

POGLAVLJE	OBLAST	OZNAKA	CILJEVI		
4 Izrada korišćenjem koda	<i>4.1 Logika</i>	3.2.7	Koristiti unete podatke od strane korisnika u programu.		
		3.2.8	Koristiti izlazne podatke za prikaz na ekranu u programu.		
		4.1.1	Definisati termin logički test u programskoj konstrukciji. Navesti svrhu logičkog testiranja u programu.		
	<i>4.2 Iteracija</i>	<i>4.1 Logika</i>	4.1.2	Prepoznati vrste Boolean logičkih izraza za generisanje tačnih ili netačnih vrednosti kao što su: =, >, <, >=, <=, <>, !=, ==, AND, OR, NOT.	
			4.1.3	Koristiti Boolean logičke izraze u programu.	
			4.2.1	Definisati termin petlja (loop) u programskoj konstrukciji. Navesti svrhu i prednosti petlje u programu.	
		<i>4.2 Iteracija</i>	4.2.2	Prepoznati vrste petlji (loops) korišćene za iteraciju: for, while, repeat.	
			4.2.3	Koristiti iteraciju (looping) u programu kao što su: for, while, repeat.	
			4.2.4	Razumeti termin beskonačna petlja (infinite loop).	
			4.2.5	Razumeti termin rekurzija.	
			<i>4.3 Uslovljenost</i>	4.3.1	Definisati termin uslovna izjava u programskoj konstrukciji. Navesti svrhu uslovne izjave u programu.
				4.3.2	Koristiti IF...THEN...ELSE uslovne izjave u programu.
			<i>4.4 Procedure i funkcije</i>	4.4.1	Razumeti termin procedura. Navesti svrhu procedure u programu.
				4.4.2	Napisati i imenovati proceduru u programu.
				4.4.3	Razumeti termin funkcija. Navesti svrhu funkcije u programu.
4.4.4	Napisati i imenovati funkciju u programu.				
<i>4.5 Događaji i komande</i>	4.5.1	Razumeti termin događaj. Navesti svrhu događaja u programu.			
	4.5.2	Koristiti obrađivače događaja (event handlers) kao što su: klik mišem, unos putem tastature, klik dugmetom, tajmer.			
	4.5.3	Koristiti dostupne generičke funkcije kao što su: math, random, time.			
5 Testiranje, debugovanje i puštanje	<i>5.1 Pokretanje, testiranje i debugovanje</i>	5.1.1	Razumeti beneficije debugovanja u programu sa ciljem rešavanja grešaka.		
		5.1.2	Razumeti vrste grešaka u programu kao što su: sintaksičke, logičke.		

POGLAVLJE	OBLAST	OZNAKA	CILJEVI
		5.1.3	Pokrenuti program.
		5.1.4	Identifikovati i popraviti sintaksičke greške u programu kao što su: netačan pravopis, nedostatak interpunkcije.
		5.1.5	Identifikovati i popraviti logičke greške u programu kao što su: netačan Boolean izraz, netačan tip podataka.
	<i>5.2 Puštanje u rad</i>	5.2.1	Proveriti da li je program u skladu sa zahtevima početnog opisa.
		5.2.2	Opisati završen program, ulogu komunikacije i vrednost.
		5.2.3	Identifikovati poboljšanja, unapređenja programa koji mogu zadovoljiti dodatne, povezane potrebe.