

Drugi kolokvijum iz Programiranja 1

Kolokvijum traje 120 minuta

Napomene:

- a) Pažljivo proučite Uputstvo pre popunjavanja Oblasca za odgovore.
- b) Vrednost odgovora: tačan = **15**; netačan = **-3.75**; nevažeći (nula ili više zacrnjenih kružića) = **0**.
- c) Na pitanjima se može osvojiti najviše **45** poena.
- d) Prvi zadatak nosi **30** poena, a drugi zadatak nosi **25** poena.

I ZADACI

1) Jedan student za pripremu ispita iz matematike koristi rusku zbirku koju mu je ostavio deda. Zbog starosti, zbirka se raspala na veći broj delova pa je student neke delove zbirke pozajmio kolegici. Student je prethodno zapisao strane na kojima se nalaze zadaci koje tek treba da provežba, pa mu je potreban program koji će proveriti da li kod sebe ima sve potrebne strane. Napisati program na programskom jeziku Pascal koji pročita broj delova zbirke koji su ostali kod studenta, a zatim za svaki deo učitava prvu i poslednju stranu dela. Potom student unosi brojeve strana koje su mu potrebne. Ukoliko strana postoji među delovima zbirke koji su trenutno kod studenta, program treba da ispiše "postoji", dok u suprotnom ispisuje "nedostaje". Unos brojeva potrebnih strana se ponavlja sve dok se za broj potrebne strane ne unese negativna vrednost, nakon čega se program završava. Smatrati da su strane delova unete ispravno i da su unete po neopadajućem broju strane.

2) Napisati program na programskom jeziku Pascal za kodiranje teksta. Program učitava tekst zapisan u jednom redu, a potom iz narednih redova čita parove brojeva koji predstavljaju pozicije znakova koji treba da zamene mesta. Zamene se rade po redosledu unošenja parova brojeva. Svaki par se primenjuje nad sadržajem teksta u trenutku kada treba da se uradi zamena. Pozicija prvog znaka je 1. Zamena se radi samo ukoliko su oba broja u opsegu validnih rednih brojeva znakova učitanoog teksta. U suprotnom, kodiranje se završava i na standardni izlaz se ispisuje rezultat kodiranja.

II PITANJA

1) Šta ispisuje sledeći program napisan na programskom jeziku Pascal ako se pretpostavi da korisnik u promenljivu *n* unese pozitivnu vrednost manju od 101 i da prilikom aritmetičkih operacija ne dolazi do prekoračenja?

<pre>PROGRAM P1(input,output); VAR n,i,j:integer; niz:ARRAY[0..100]OF integer; BEGIN readln(n); FOR i:=0 TO n DO read(niz[i]); readln; i:=0; j:=1; WHILE j<=n DO BEGIN IF (niz[i] MOD 7 = 0) THEN i:=i+1</pre>	<pre>ELSE IF (niz[j] MOD 7 = 0) THEN BEGIN niz[i] := niz[i] - niz[j]; niz[j] := niz[i] + niz[j]; niz[i] := niz[j] - niz[i]; i:=i+1 END; j:=j+1 END; FOR j:=0 TO n DO write(niz[j], ' '); writeln END.</pre>
---	---

- A) Ispisuje elemente niza po neopadajućoj vrednosti ostatka pri deljenju sa 7.
- B) Ispisuje elemente niza tako da one koji su deljivi sa 7 umanju za vrednost sledećeg elementa.
- (C)** Ispisuje prvo elemente koji su deljivi sa 7, pa potom one koji nisu deljivi sa 7.

2)Šta ispisuje sledeći program napisan na programskom jeziku Pascal ako pri pokretanju korisnik unosi sledeći sadržaj: 4 9 17<enter>-1 8<enter>3 2<enter>? Pretpostaviti da oznaka <enter> znači da je korisnik pritisnuo taster za prelazak u novi red.

<pre>PROGRAM P2(input,output); VAR a, b, c, d, i:integer; BEGIN readln(a,b); read(c); readln(d,i);</pre>	<pre>REPEAT IF a+c+d > 8 THEN IF b MOD d < 2 THEN d:=d-2 ELSE c:=c-4; i:=i-1 UNTIL (i=0) OR (b-c>10); writeln(d+i-c) END.</pre>
--	--

A)9

(B) 12

C)10

3)Koji od ponuđenih odgovora zadovoljava dato sintaksno pravilo u EBNF notaciji za neterminal rec?
rec = "record " red "end".

red = ident_list ":" (pt | ar | rec) ";".

ident_list = (ident_list "," ident | ident).

ident = (malo_slovo | malo_slovo ident | "_" ident | ident cifra).

malo_slovo =

"a"|"b"|"c"|"d"|"e"|"f"|"g"|"h"|"i"|"j"|"k"|"l"|"m"|"n"|"o"|"p"|"q"|"r"|"s"|"t"|"u"|"v"|"w"|"x"|"y"|"z".

cifra = "0"|"1"|"2"|"3"|"4"|"5"|"6"|"7"|"8"|"9".

pt = "char"|"integer"|"real"|"boolean".

ar = "array[" cifra ".." cifra "]" of " (pt | ar | rec).

(A) record tr:array[9..3] of record _b2:integer;_a_b23:record x:char;end;c:boolean;end;end

B)record a:record t:char;end;b:array[1..5] of record _b2:integer;end;end;

C)record b:array[1..5] of array[4..8] of boolean; a_23,_b2:integer;end

4)Koje od ponuđenih pravila je dovoljno dodati datim pravilima u BNF notaciji da bi neterminal start mogao da se upari sa terminalom 1111100111100101110011101?

<start> ::= 11<a>|1

 ::= 101|01

<c> ::= 1100|<c>11|<c>00

A)<a> ::= 1|<a><c>|<a><c>

B)<a> ::= 111|<a>00|<a><c>|<a>

(C) <a> ::= 1|<a>|<a><c>