

Ispit iz Programiranja 2

Šifra zadatka - 1000000000

Ispit traje 150 minuta

Napomene:

- a) Pažljivo proučite Uputstvo pre popunjavanja Oblasca za odgovore.
- b) Vrednost odgovora: tačan = **5**; netačan = **-1.25**; nevažeci (nula ili više zacrnjenih kružića) = **0**.
- c) Na pitanjima se može osvojiti najviše **15** poena. Prvi zadatak nosi **25** poena, dok drugi nosi **30** poena.

I ZADACI

1)Napisati program na programskom jeziku C koji vrši kriptovanje teksta koji se unosi u jednom redu sa standardnog ulaza. Napisati potprogram koji generiše niz od 26 malih slova za preslikavanje na sledeći način: za svako malo slovo iz engleskog alfabeta na slučajan način se pronalazi slovo u koje će se originalno slovo preslikavati, ali tako da među generisanim slovima za preslikavanje nema ponavljanja. Napisati potprogram koji za jedan karakter vrši kriptovanje, tako što se on menja slovom u koje se karakter preslikava. Ukoliko se karakter ne nalazi u skupu malih slova engleskog alfabeta, onda ga ne treba menjati. Napisati glavni program koji čita red teksta sa standardnog ulaza, koristeći traženi potprogram kriptuje pročitani red i na standardni izlaz ispisuje kriptovani tekst. Maksimalna dužina reda je 80 znakova. Smatrati da se ulazni tekst sastoji samo od malih slova i specijalnih karaktera. Primer:

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
d	k	v	q	f	i	b	j	w	p	e	s	c	x	h	t	m	y	a	u	o	l	r	g	z	n
Ulaz: if we wish to replace letters...																									
Izlaz: wi rf rwaj uh yftsdvf sfuufya...																									

2)Napisati program na jeziku C koji razvrstava stavke listinga mobilne telefonije po vrsti saobraćaja. U datoteci listing.txt evidentirane su aktivnosti mobilnog pretplatnika (pozivi, sms i podaci). Svaki red teksta sadrži sledeće podatke o jednoj aktivnosti razdvojene po jednim znakom razmaka: vrsta aktivnosti (karakter, P-poziv, S-sms, D-podaci), kontaktirani broj (string, za D "data"), količina saobraćaja (realan broj, za P broj sekundi, za D broj kilobajta, za S cena sms-a). Napisati potprogram int read\_activity(FILE\*, Activity\*\*); koji čita jednu aktivnost iz listinga, pri čemu Activity predstavlja strukturu za koju je potrebno alocirati prostor u potprogramu i vratiti preko drugog parametra. Potprogram vraća 1 ako uspe, a 0 za grešku. U slučaju greške, iz glavnog programa ispisati poruku o grešci i prekinuti dalje izvršavanje. Napisati glavni program koji preko komandne linije dobija naziv datoteke sa listingom. Program otvara datoteku listinga, učitava jednu po jednu aktivnost i razvrstava po vrsti saobraćaja, pri čemu razvrstane aktivnosti smešta u tri liste, koje treba da budu uređene nerastuće po količini. Na kraju glavni program treba da ispiše razvrstane aktivnosti u tri izlazne datoteke pozivi.txt, sms.txt i podaci.txt. Format reda izlazne datoteke je: <kontaktirani broj><razmak><količina><jedinica>. Za podatke umesto kontaktiranog broja ispisati reč "data". Stavka <jedinica> može da bude, u zavisnosti od vrste saobraćaja, "sek" za pozive, "KB" za podatke, "din" za poruke. Na kraju svake izlazne datoteke potrebno je ispisati ukupnu količinu saobraćaja te vrste uz navođenje odgovarajuće merne jedinice. Voditi računa o pravilnom rukovanju korišćenim resursima. Sadržaj ulazne datoteke je dozvoljeno pročitati samo jednom, a maksimalan broj redova nije unapred poznat.

II PITANJA

1)Šta ispisuje sledeći program na programskom jeziku C?

#include <stdio.h> int f1(int *p) { int *q; return (q=p++, *p = *q++ -1); } int f2(int *q) { return q[2]++; } void main() { int (*f[])(int*) = {f1, f2, f1, f2}; int (**fp)(int*) = f;	int a[8] = {4, 5, 3, 5}, i=1, s= 0; while ((fp+=a[i]%2)<=&f[3]) s+=a[i]=(*fp)(&a[i]), i++; printf("%d", s); }
--	--

A) 5 (B) 8

C) 6

2)Šta ispisuje sledeći program na programskom jeziku C?

#include <stdio.h> int g(int y, int br) { int x = (y & 1) ? 1 : -1; return y ? x + g(y >> 1, br - 1) : -br; } int f(int x, int y) { int z = x & 7; if (y == 0) return 0;	switch (z & 4) { case 0: return g(z, 3) + f(x>>3, y-1); case 4: return g(~z & 7, 3)+f(x>>3, y-1); }  void main() { printf("%d", f(342, 3)); }
---	--

A) 2

B) 1

(C) -3

3)Šta može da se stavi umesto \*\*\* tako da se novi element ispravno umetne u dvostruko ulančanu listu ispred elementa na koji ukazuje pokazivač pc? Smatrati da funkcija readList ispravno učitava nepraznu listu i vraća adresu prvog elementa, kao i da funkcija find vraća pokazivač na element liste ispred kojeg treba ubaciti novi element.

#include <stdlib.h> struct elem { struct elem *prev, *next; } typedef struct elem Elem;	void main () { Elem *lst = readList(); Elem *pc = find(lst); Elem *n = malloc(sizeof(Elem)); *** if (pc == lst) lst = n; }
---	--

(A)

n->next = pc;  
n->prev = n->next->prev;  
if (pc->prev) n->prev->next = n;  
n->next->prev = n;

B)

pc->prev->next = n;  
pc->prev = n;  
n->prev = pc->prev;  
n->next = pc;

(C)

n->next = pc;  
n->prev = pc->prev;  
if (n->prev) n->prev->next = n;  
pc->prev = n;

4)Šta ispisuje sledeći program na programskom jeziku C? Pretpostaviti da je program pozvan sa program.exe 3 1 5 2 6 4, gde je program.exe naziv programa.

#include<stdlib.h> #include<stdio.h> void main(int argc, char* argv[]){ int i , x = 2, s = 0;	for (i = 1; i < argc-2; i++){ s += x = atoi(argv[x]); } printf("%d%d",s, x); }
--	--

A) 105

(B) 156

C) 194