

PROGRAMIRANJE 1

**Milena Vujošević Janičić, Jovana Kovačević,
Danijela Simić, Anđelka Zečević**

PROGRAMIRANJE 1

Zbirka zadataka

**Beograd
2016.**

Autori:

dr Milena Vujošević Jančić, docent na Matematičkom fakultetu u Beogradu

dr Jovana Kovačević, docent na Matematičkom fakultetu u Beogradu

Danijela Simić, asistent na Matematičkom fakultetu u Beogradu

Anđelka Zečević, asistent na Matematičkom fakultetu u Beogradu

PROGRAMIRANJE 1

Zbirka zadataka

Sadržaj

0.1	Niske	v
0.1.1	Osnovna konverzija	v
0.1.2	Karakteristike (da/ne)	vi
0.1.3	Izbacivanje karaktera	viii
0.1.4	Kopiranje jedne niske u drugu	ix
0.1.5	Numericki	x
0.1.6	Linije sa min/max	xi
0.1.7	Klonovi	xii
0.1.8	Rotacije	xiii
0.1.9	Sifriranja	xiv
0.1.10	Teski	xv
0.1.11	Treba u drugu sekciju	xv
0.1.12	Haoticni - za izbacivanje	xvi

0.1 Niske

0.1.1 Osnovna konverzija

Zadatak 0.1.1 Napisati funkciju koja konvertuje datu nisku tako što mala slova menja u velika a velika u mala. Napisati program koji testira ovu funkciju za učitane nisku maksimalne dužine 10 karaktera.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: BeoGrad
|| bEOgRAD
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: A+B+C
|| a+b+c
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: 12345
|| 12345
```

Zadatak 0.1.2 Napisati funkciju `void modifikacija(char s[])` koja modifikuje nisku `s` tako što svaki drugi karakter zameni zvezdicom. Pretpostaviti

da niska s neće biti duža od 20 karaktera. Napisati i program koji testira rad napisane funkcije.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: 123abc789XY
|| Modifikovana niska je: 1*3*b*7*9*Y
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: zimA
|| Modifikovana niska je: z*m*
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: SNEG
|| Modifikovana niska je: S*E*
```

Primer 4

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: *a*b*c*
|| Modifikovana niska je: *****
```

Zadatak 0.1.3 Napisati sledeće funkcije:

- (a) `int poredjenje(char s1[], char s2[])` koja vraća 1 ako su s_1 i s_2 jednake niske, a 0 u suprotnom
- (b) `void u_velika_slova(char s[])` koja pretvara sva slova niske s u velika, a ostale znakove ne menja

Napisati program koji učitava dve reči dužine najviše 20 znakova i, koristeći napisane funkcije, ispisuje da li su one jednake ako se sva slova pretvore u velika slova.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| isPit2010
|| IsPit2010
|| jesu jednake
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Prog1
|| prog2
|| nisu jednake
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| junski
|| septembarski
|| nisu jednake
```

0.1.2 Karakteristike (da/ne)

Zadatak 0.1.4 Napisati program koji ispituje da li se uneta niska završava samoglasnikom.

- (a) Napisati funkciju `int samoglasnik(char c)` ispituje da li je c samoglasnik
- (b) Napisati funkciju `int samoglasnik_na_kraju(char s[])` koja pomoću prethodne funkcije ispituje da li se niska s završava samoglasnikom
- (c) Napisati program koji učitava nisku maksimalne dužine 20 karaktera i ispisuje da li završava samoglasnikom ili ne.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: abcde
|| Niska se završava samoglasnikom!
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: AaBb+cCdD
|| Niska se ne završava samoglasnikom!
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: pRoGraMiranjE
|| Niska se završava samoglasnikom!
```

Primer 4

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: pRoGraMiranjE1
|| Niska se ne završava samoglasnikom!
```

Zadatak 0.1.5 Napisati program koji učitava nisku s i karakter c i ispituje da li se c pojavljuje u niski s . Ako se pojavljuje, program treba da ispiše indeks prvog pojavljivanja karaktera c u niski s a u suprotnom -1. Možemo pretpostaviti da string ima najviše 20 karaktera.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku:
|| bazen
|| Unesite karakter:
|| z
|| 2
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku:
|| lezaljka
|| Unesite karakter:
|| a
|| 3
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku:
|| limunada
|| Unesite karakter:
|| b
|| -1
```

Zadatak 0.1.6 Napisati funkciju `int sadrzi_veliko(char s[])` koja proverava da li niska s sadrži veliko slovo. Napisati program koji učitava nisku maksimalne dužine 20 karaktera, poziva napisanu funkciju i ispisuje odgovarajuću poruku.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku:
|| naocare
|| Niska ne sadrzi veliko slovo
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku:
|| DiopTrija0.75
|| Niska sadrzi veliko slovo
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku:
|| 21.06.2017.
|| Niska ne sadrzi veliko slovo
```

Jovana: Proveriti naziv funkcije u resenju

Zadatak 0.1.7 Napisati funkciju `int podniska(char s[], char t[])` koja proverava da li je niska t podniska niske s . Napisati program koji učitava dve niske maksimalne dužine 10 karaktera i testira rad napisane funkcije.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku s: abcde
Unesite nisku t: bcd
t je podniska niske s!
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku s: abcde
Unesite nisku t: bCd
t nije podniska niske s!
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku s: abcde
Unesite nisku t: def
t nije podniska niske s!
```

0.1.3 Izbacivanje karaktera

Zadatak 0.1.8 Napisati funkciju `void skрати(char s[])` koja uklanja beline sa kraja date niske. Napisati program koji testira ovu funkciju za učitano nisku maksimalne dužine 10 karaktera. Prikazati učitano i izmenjenu nisku između zvezdica.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: rep belina
ucitana niska: *rep belina *
izmenjena niska: *rep belina*
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: tri tabulatora na kraju
ucitana niska: *tri tabulatora na kraju *
izmenjena niska: *tri tabulatora na kraju*
```

Zadatak 0.1.9 Napisati funkciju `void ukloni_slova(char s[])` koja iz niske `s` uklanja sva mala i sva velika slova. Napisati program koji učitava nisku maksimalne dužine 20 karaktera, poziva napisanu funkciju i ispisuje izmenjenu nisku.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
abcd123ABCD
123
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
1+2=3
1+2=3
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
malaVELIKA
```

Jovana: Proveriti naziv funkcije u resenju

Zadatak 0.1.10 Napisati funkciju `void ukloni(char *s)` koja iz niske uklanja sva slova iza kojih neposredno sledi slovo koje je u abecedi nakon njih, pri čemu se veličina slova zanemaruje. Testirati funkciju u programu za učitano liniju od najviše 100 karaktera.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
zdRaVo suIma
zRVo vma
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
12345AbcD
12345D
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
JeD1aN D52Va.
JeD1N D52Va.
```

0.1.4 Kopiranje jedne niske u drugu

Zadatak 0.1.11 Napisati program koji učitava nisku *src* i formira nisku *dst* trostrukim nadovezivanjem niske *src*. Možemo pretpostaviti da niska *src* sadrži najviše 30 karaktera.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: dan
dandandan
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: 3sesira
3sesira3sesira3sesira
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: a-b=5
a-b=5a-b=5a-b=5
```

Zadatak 0.1.12 Napisati program koji za unetu reč maksimalne dužine 20 karaktera formira rezultujuću reč tako što unetu reč kopira 4 puta pri čemu se između dva kopiranja umeće crtica.

Na primer ako je uneta reč **ana**, formirana reč treba da bude **ana-ana-ana-ana**. Zadatak uraditi:

- (a) pisanjem odgovarajuće funkcije koja vrši nadovezivanje reči,
- (b) koristeći postojeću funkciju iz biblioteke `string.h` (`strcat`).

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: ana
ana-ana-ana-ana
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: 123
123-123-123-123
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: x*y
x*y-x*y-x*y-x*y
```

Zadatak 0.1.13 Napisati funkciju `void kopiraj_n(char t[], char s[], int n)` koja kopira najviše *n* karaktera niske *s* u nisku *t*. Napisati program koji testira rad napisane funkcije. Možemo pretpostaviti da je maksimalna dužina niske *s* 20 karaktera.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: abcdef
Unesite broj n: 3
Rezultujuca niska: abc
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: gromobran
Unesite broj n: 4
Rezultujuca niska: grom
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: abc
Unesite broj n: 15
Rezultujuca niska: abc
```

Zadatak 0.1.14 Napisati funkciju `void dupliranje(char t[], char s[])` koja na osnovu niske *s* formira nisku *t* tako što duplira svaki karakter niske *s*. Napisati program koji učitava nisku maksimalne dužine 20 karaktera i testira rad napisane funkcije.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: zima
zziimmaa
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: A+B+C
AA++BB++CC
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku: C
CC
```

0.1.5 Numericki

Zadatak 0.1.15 Napisati program koji učitava nisku cifara sa eventualnim vodećim znakom i pretvara je u ceo broj. NAPOMENA: *Zadatak realizovati bez korišćenja ugrađene funkcije `atoi` iz biblioteke `stdlib.h`*

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
-1238
-1238
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
73
73
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
+1
1
```

Zadatak 0.1.16 Napisati program koji učitava ceo broj i pretvara ga u nisku.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite ceo broj:
-6543
-6543
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite ceo broj:
84
84
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite ceo broj:
5
5
```

Zadatak 0.1.17 Napisati funkciju `int heksa_broj(char s[])` koja proverava da li je niskom *s* zadat korektan heksadekadni broj. Heksadekadni broj je korektno zadat ako počinje prefiksom *0x* ili *0X* i ako sadrži samo cifre i mala ili velika slova *A*, *B*, *C*, *D*, *E* i *F*. Funkcija treba da vrati vrednost 1 ako je niska korektan heksadekadni broj, odnosno 0 ako nije. Napisati program koji učitava nisku maksimalne dužine 7 karaktera i ispisuje rezultat rada funkcije.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: 0x12EF
|| Korektan heksadekadni broj!
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: 0X22af
|| Korektan heksadekadni broj!
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: 0xErA9
|| Nekorektan heksadekadni broj!
```

Zadatak 0.1.18 Napisati funkciju `int heksa_broj(char s[])` koja izračunava dekadnu vrednost heksadekadnog broja zadatog niskom *s*. Napisati program koji učitava nisku maksimalne dužine 7 karaktera i ispisuje rezultat rada funkcije. Pretpostaviti da je uneta niska korektan heksadekadni broj.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: 0x2A34
|| 10804
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: 0Xff2
|| 4082
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku: 0xE1A9
|| 57769
```

0.1.6 Linije sa min/max

Zadatak 0.1.19 Napisati funkciju `int ucitaj_liniju(char s[], int n)` koja učitava liniju maksimalne dužine *n* u nisku *s* i vraća dužinu učitane linije. Linija može da sadrži blanko znakove ali ne sadrži `\n` ili EOF. Napisati program koji učitava linije do EOF i ispisuje najdužu liniju i njenu dužinu. Ukoliko ima više linija maksimalne dužine, ispisati prvu. Možemo pretpostviti da svaka linija sadrži najviše 80 karaktera.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite linije:
|| Dobar dan!
|| Kako ste, sta ima novo?
|| Ja sam dobro.
|| Kako ste, sta ima novo?
|| 23
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite linije:
|| Prva linija
|| Druga linija
|| Treca linija
|| Druga linija
|| 12
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite linije:
|| Danas je lep dan.
|| Danas je lep dan.
|| 17
```

Zadatak 0.1.20 Napisati sledeće funkcije:

- (a) `int procitaj_recenicu(char *s, int max_len)` koja učitava rečenicu i smešta je u nisku `s`. Funkcija vraća dužinu učitane rečenice. Učitavanje se završava nakon učitanoj karaktera `.` ili nakon `max_len - 1` učitanih karaktera
- (b) `void prebroj(char *s, int *broj_malih, int *broj_velikih)` koja prebrojava mala i velika slova u niski `s`

Napisati program koji učitava rečenice do unosa prazne rečenice i ispisuje onu kod koje je razlika broja malih i velikih slova najveća.

Jovana: Proveriti u resenju da li je unos recenice do unosa prazne recenice. Ovo nema resenje?

0.1.7 Klonovi

Zadatak 0.1.21 Napisati funkciju `char* strchr_klon(char s[], char c)` koja vraća pokazivač na prvo pojavljivanje karaktera `c` u niski `s` ili `NULL` ukoliko se karakter `c` ne pojavljuje u niski `s`. Napisati program koji učitava nisku maksimalne dužine 20 karaktera i dodatni karakter i testira rad napisane funkcije.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku s:
||      programiranje
|| Unesite karakter c: a
|| Karakter se nalazi u niski!
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku s: 123456789
|| Unesite karakter c: y
|| Karakter se ne nalazi u
||      niski!
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku s: leto2017
|| Unesite karakter c: 0
|| Karakter se nalazi u niski!
```

Zadatak 0.1.22 Napisati funkciju `int strspn_klon(char t[], char s[])` koja izračunava dužinu prefiksa niske `t` sastavljenog od karaktera niske `s`. Napisati program koji učitava dve niske maksimalne dužine 20 karaktera i ispisuje rezultat poziva napisane funkcije.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku t: program
|| Unesite nisku s: opqr
|| 3
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku t: aaiioo124
|| Unesite nisku s: aeioa
|| 6
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite nisku t: 5296abc
|| Unesite nisku s: 0123456789
|| 4
```

Zadatak 0.1.23 Napisati funkciju `int f(char s[], char t[])` koja izračunava dužinu početnog dela niske `s` sastavljenog isključivo od karaktera sadržanih

u niski t . Napisati program koji testira ovu funkciju za dve unete niske maksimalne dužine 100 karaktera.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite prvu nisku:
734a.bf62
Unesite drugu nisku:
0123456789
3
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite prvu nisku:
abracadabra
Unesite drugu nisku:
brada
4
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite prvu nisku:
popokatepetl
Unesite drugu nisku:
opna
2
```

Zadatak 0.1.24 Napisati funkciju `char* strstr_klon(char s[], char t[])` koja vraća pokazivač na prvo pojavljivanje niske t u niski s ili `NULL` ukoliko se niska t ne pojavljuje u niski s . Napisati program koji testira napisanu funkciju tako što učitava pet linija i ispisuje sve redne brojeve linija koje sadrže nisku *program*. Ukoliko ne postoji linija sa niskom *program*, ispisati odgovarajuću poruku. Možemo pretpostaviti da je svaka linija maksimalne dužine 100 karaktera kao i da se linije numerišu od broja 1.

Jovana: Tekst je malo izmenjen, prilagoditi resenje

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite pet linija:
novi red*program
c prog. jezik
c? programskih jezik
Programski odbor
<b>program</b>
1 3 5
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite pet linija:
Programske paradigme
su predmet na
trecoj godini
programerskih
smerova.
4
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite pet linija:
U narednim
linijama
necemo navoditi
nisku koja se
trazi.
Nijedna linija ne sadrzi
nisku program.
```

0.1.8 Rotacije

Zadatak 0.1.25 Napisati funkciju `void obrni(char s[], int k)` koja rotira nisku s za k mesta ulevo. Napisati program koji učitava nisku maksimalne dužine 20 karaktera, poziva napisanu funkciju i ispisuje obrnutu nisku.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
sveska
2
eskasv
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
olovka
6
olovka
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
rezac
8
acrez
```

0.1.9 Sifriranje

Zadatak 0.1.26 Napisati program koji šifrira unetu nisku tako što svako slovo zamenjuje sledećim slovom abecede, slova 'z' i 'Z' zamenjuje redom sa 'a' i 'A', a ostale karaktere ostavlja nepromenjene. Možemo podrazumevati da uneta niska nije duža od 20 karaktera.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
AbcXyz
BcdYza
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
lmnopqr123
mnopqrs123
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
1,2,3,4,5
1,2,3,4,5
```

Zadatak 0.1.27 Napisati funkciju `void sifruj(char rec[], char sifra[])` koja na osnovu date reči formira šifru tako što se svako slovo u reči zameni sa naredna tri slova u abecedi. Napisati program koji testira napisanu funkciju za reč maksimalne dužine 20 karaktera.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
tamo
uvwbcnoppqr
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
Zec
ABCfghdef
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
a+b=c
bcd+cde=def
```

Zadatak 0.1.28 Napisati funkciju `void indel(char s1[], char s2[], char c1, char c2)` koja na osnovu niske s_1 formira nisku s_2 udvajanjem svih karaktera c_1 u niski s_1 i izbacivanjem svih karaktera c_2 iz niske s_1 , dok ostali karakteri ostaju nepromenjeni. Napisati program koji testira ovu funkciju za unetu nisku i dva uneta karaktera. Možemo pretpostaviti da uneta niska neće biti duža od 20 karaktera. NAPOMENA: *Dopušteno je koristiti pomoćni niz.*

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
flomaster
Unesite prvi karakter:
m
Unesite drugi karakter:
s
flosaster
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
bojica
Unesite prvi karakter:
b
Unesite drugi karakter:
a
bbojic
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite nisku:
patentara
Unesite prvi karakter:
t
Unesite drugi karakter:
a
pttenttr
```

0.1.10 Teski

Zadatak 0.1.29 Napisati program za rad sa nizovima niski.

- (a) Napisati funkciju

```
int prepis(char a[][21], int na, char b[][21])
```

koja iz niza reči a dužine na prepisuje u niz b one reči koje su sastavljene samo od malih ili samo od velikih slova i vraća dužinu niza b .

- (b) Napisati program koji učitava broj n ($0 < n \leq 50$) i n reči razdvojenih blanko znakom, poziva funkciju `prepis` i ispisuje sve reči sastavljene samo od malih ili samo od velikih slova. Možemo pretpostaviti da su unete reči maksimalne dužine 20 karaktera. U slučaju da je n van dozvoljenog opsega, ispisati odgovarajuću poruku.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| 3 abc ABC aBc  
|| abc ABC
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
|| 2 mM RGa
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:  
|| -3  
|| Nekorektan broj reci!
```

0.1.11 Treba u drugu sekciju

Zadatak 0.1.30 Napisati program za šifrovanje reči na različite načine.

- (a) Uvesti tip podataka **Sifra** kojim se opisuje način šifrovanja alfanumeričkih karaktera. Svaka šifra se opisuje celobrojnomo vrednošću b koja određuje broj pozicija pomeranja, kao i karakterom L ili D koji određuje smer pomeranja (levo ili desno).
- (b) Napisati funkciju `void sifruj(char rec[], Sifra s)` koja transformiše reč rec po šifri s . Reč se šifrjuje tako što se svako slovo zamenjuje slovom za b mesta levo ili desno od njega u abecedi, i to ciklično. Cifre se šifriraju na isti način.
- (c) Napisati program koji učitava način šifrovanja u obliku $b_1c_1 \dots b_mc_m$ ($1 \leq m \leq 20$), broj n i n reči maksimalne dužine 20 karaktera i ispisuje šifrovane reči.

Jovana: Pogledati da li je dužina sifre ograničena na 20 i da li se unosi nova

sifra za svaku rec (bilo je neprecizno napisano). Dodati naredni primer u test primere. Zar ovo ne ide u strukture?

Npr: za $b=2$, i smer='D' : a se menja sa c, b sa d,..., x sa z, y sa a, z sa b, 1 sa 3, .. 8 sa 0, 9 sa 1

0.1.12 Haoticni - za izbacivanje

Zadatak 0.1.31 Napisati program koji učitava dve niske s i t i ako su jednake, izdaje odgovarajuću poruku a u suprotnom ispituje da li je niska t podniska niske t i ukoliko jeste, ispisuje počev od kog indeksa niske s počinje prvo pojavljivanje niske t . Ako niska t nije podniska niske s , ispisati odgovarajuću poruku. Možemo pretpostaviti da niske ne sadrže više od 20 karaktera.

Jovana: konfuzija. Predlazem da izbacimo.

Zadatak 0.1.32 Napisati funkciju `void min_razlika(char s[], char s1[], char s2[])` koja u niski s pronalazi dve reči koje imaju minimalnu razliku između svojih samoglasnika. (Reč je niz karaktera između dve praznine; razmak između samoglasnika reči *dan*as i *jut*ro je 2, a razmak između *sut*rk i *mno*zen je 5). Napisati program koji testira napisanu funkciju za unete niske maksimalne dužine 20 karaktera.

Jovana: Konfuzno. Sta je razlika/razmak? Predlazem da izbacimo.

Zadatak 0.1.33 Napisati funkciju `int pp(char s[], char t[])` koja vraća poziciju pojavljivanja poslednjeg karaktera niske s u niski t , zanemarujući pritom razliku između velikih i malih slova, ili -1 ako takvog karaktera nema. Napisati program koji učitava dve niske maksimalne dužine 20 karaktera i testira napisanu funkciju.

Jovana: Konfuzno. Predlazem da izbacimo.

Primer 1

INTERAKCIJA SA PROGRAMOM:

a4BA3Bc A3b

5

Zadatak 0.1.34 Napisati program za rad sa brojevima zapisanim u različitim brojevnim sistemima.

- (a) Napisati funkciju `unsigned btoi(char s[], unsigned char b)` koja određuje vrednost zapisa datog neoznačenog broja s u datoj osnovi b .

- (b) Napisati funkciju `void itob(unsigned n, unsigned char b, char s[])` koja datu vrednost n zapisuje u datoj osnovi b i smešta rezultat u nisku s .
- (c) Napisati program koji učitava linije do unosa prazne linije. Svaka linija sadrži jedan dekadni, oktalni ili heksadekadni broj (zapisan kako se zapisuju konstante u programskom jeziku C). Program za svaki uneti broj ispisuje njegov binarni zapis. Pretpostaviti da će svi uneti brojevi biti u opsegu tipa `unsigned`.

Jovana: Pogledati resenje i prilagoditi tekst.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| 0x49 0x1ABC
|| 1001001 1101010111100
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| 012 435 0x64FE
|| 1010 110110011 110010011111110
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| 123 0777
|| 1111011 11111111
```

Primer 4

```
|| INTERAKCIJA SA PROGRAMOM:
|| 981
|| 1111010101
```

Zadatak 0.1.35 Napisati funkciju `void sifrat(char rec[], char kljuc[])` koja šifruje rec na sledeći način: za svako slovo reči rec i odgovarajuće slovo reči $kljuc$ određuje koliki je (alfabetski) razmak između njih (k) i potom k -to slovo reči rec zamenjuje k -tim slovom alfabeta. Podrazumeva se da je $kljuc$ duži od reči.

Jovana: konfuzija. Predlazem da izbacimo

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| bac
|| dfge
|| bed
```