

Programiranje 2
Beleške sa vežbi
Školska 2006/2007 godina

Matematički fakultet, Beograd

Jelena Tomašević

May 28, 2007

Sadržaj

1

5

1

Programiranje 1, april 2007. - grupa A

1. Data su dva celobrojna niza, a i b, uređena u rastućem poretku čije su dimenzije redom na i nb. Sastaviti funkciju koja formira niz c koji se sastoji od elemenata koji su zajednički nizovima a i b.
2. Sastaviti funkciju koja za zadatu nisku x formira nisku y koja se sastoji od alfabetskih karaktera niske x. Npr. za x="22.april 2007. god.", y="aprilgod".
3. Napisati funkciju koja proverava da li postoji element datog celobrojnog niza koji je jednak datom broju.
4. Sastaviti program koji sadržaj datoteke seminar.txt (koja sadrži neki ASCII-tekst) izlistava na ekranu i to u blokovima po 25 redova sa 80 karaktera u redu. Kad korisnik unese proizvoljni alfabetski znak sa tastature, ispisuje se sledeći ekran, i tako sve do kraja datoteke.

Programiranje 1, april 2007. - grupa B

1. Data su dva celobrojna niza, a i b, uređena u rastućem poretku čije su dimenzije redom na i nb i čiji su svi elementi različiti. Sastaviti funkciju koja formira niz c koji se sastoji od elemenata koji se javljaju u nizu a ali ne u nizu b.
2. Sastaviti funkciju koja za zadatu nisku x formira nisku y koja se sastoji od nealfabetskih karaktera niske x. Npr. za x="22.april 2007. god.", y="22. 2007".
3. Napisati funkciju koja proverava da li su svi elementi datog celobrojnog niza jednaki datom broju.
4. Sastaviti program koji datoteku ulaz.txt (koja sadrži neki ASCII-tekst) prepisuje u datoteku izlaz.txt tako što zamenjuje svaki znak za kraj reda razmakom (blanko-simbolom). Ako se u rezultujućem tekstu pojavi više uzastopnih razmaka, zameniti ih jednim razmakom.

Osnovi programiranja, februar 2007. - grupa A

1. Reč je niska alfabetskih karaktera. Sastaviti program koji iz datoteke čita reči, a zatim ispisuje različite reči u obrnutom redosledu. Koristiti dinamičke strukture podataka u rešavanju.

2. Sastaviti funkciju koja za dati ceo broj tipa unsigned long int vraća
 - (a) najmanji ceo broj koji se sastoji od istog broja 0 i 1 i koji je tipa unsigned long i
 - (b) broj bitova postavljenih na 1.
3. Sastaviti program koji ispisuje prvih 100 elemenata Fibonačijevog niza zadatog sledećim formulama

$$f_1 = 1, f_2 = 2, f_n = f_{n-1} + f_{n-2} (n > 2)$$

tačno u svakoj cifri. (Napomena: ako je f_n tipa unsigned long moguće je ispisati prva 43 elementa niza.)

4. Napisati funkciju

```
char * rstr_replace(char* str, char* substr, char* dstr);
```

koja zamenjuje poslednje pojavljivanje date podniske (substr) u datoj niski (str) drugom niskom (dstr), i vraća početak druge niske u prvoj ili NULL ako se niska substr ne pojavljuje u niski str.

Programiranje 1, januar 2007. - grupa B

1. Dat je neuređen celobrojni niz **a** čija je dimenzija **na**. Sastaviti funkciju koja iz niza **a** isključuje sve ponovljene elemente i ažurira dimenziju niza.
2. Dve niske su anagrami ako se sastoje od istih karaktera. Sastaviti funkciju koja vraća 1 ako su niske x i y anagrami, inače 0. Npr. niske "vrata" i "trava" su anagrami.
3. Data je struktura

```
struct complex{
    float Re;
    float Im; }
```

Ova struktura opisuje kompleksan broj čiji je realni deo Re a imaginarni Im. Sastaviti funkciju koja izračunava moduo proizvoda dva kompleksna broja. Sastaviti program koji testira ovu funkciju, a rezultat ispisuje u obliku "Moduo proizvoda (1. argument) i (2. argument) je (vrednost modula)".

4. Sastaviti program koji iz datoteke **ulaz.txt** (koja sadrži neki ASCII-tekst) prepisuje u datoteku **izlaz.txt** samo cifre i razmake (blanko), dok ostale karaktere zanemaruje. Po izvršenom kopiranju, program treba da štampa izveštaj o broju karaktera koji nisu prepisani.
5. Sastaviti program koji ispisuje prvih 10 elemenata Fibonačijevog niza zadatog sledećim formulama

$$f_1 = "a" f_2 = "ab" f_n = f_{n-1} * f_{n-2} (n > 2)$$

gde * označava operator dopisivanja niski.

Programiranje 1, januar 2007. - grupa A

1. Data su dva celobrojna niza **a** i **b**, uređena u rastućem poretку čije su dimenzije redom **na** i **nb**. Sastaviti funkciju koja formira niz **c** koji se sastoji od elemenata nizova **a** i **b**, a koji je uređen takođe u rastućem poretку.
2. Sastaviti funkciju koja od dve zadate niske **x** i **y** formira treću nisku **z** koja se sastoji od zajedničkih karaktera niski **x** i **y**. Npr. **x**="Beograd" i **y**="Novi Sad", **z**="oad" ili **z**="ado" ili...(Formirati tačno jednu takvu nisku).
3. Data je struktura

```
struct tacka{
    float a;
    float b;
    char naziv[5];}
```

Ova struktura opisuje tačku u ravni sa koordinatama (**a,b**) kojoj je dodeljeno ime **naziv**. Sastaviti funkciju **rastojanje** koja izračunava euklidsko rastojanje dve tačke tipa **struct tacka**. Sastaviti program koji testira funkciju **rastojanje**. Rezultat ispisuje u obliku "rastojanje od tacke (naziv 1) do tacke (naziv2) je (vrednost rastojanja)".

4. Sastaviti program koji iz datoteke **ulaz.txt** (koja sadrži neki ASCII-tekst) prepisuje u datoteku **izlaz.txt** samo alfabetske karaktere i razmak (blanko), dok ostale karaktere zane-maruje. Po izvršenom kopiranju, program treba da štampa izveštaj o broju prepisanih karaktera.
5. Sastaviti program koji ispisuje elemente Fibonačijevog niza zadatog sledećim formulama

$$f_1 = 1, f_2 = 2, f_n = f_{n-1} + f_{n-2} (n > 2)$$

Proceniti najveću vrednost za **n** ako se **f** deklarise kao **short int**.

Osnovi programiranja, januar 2007. - grupa A

1. Definišimo tipove **Osoba** i **Skup** na sledeći način:

```
struct osoba {
    char ime[MAXIME];
    unsigned int starost;
}
struct skup {
    unsigned int dim;
    Osoba niz[MAX];
}
typedef struct osoba Osoba;
typedef struct skup Skup;
```

gde je **MAXIME** maksimalna dužina karakterske niske, a **MAX** maksimalan broj elemenata skupa. Pretpostavlja se da je niz u strukturi **skup** sortiran u rastućem redosledu (prvo po imenu, a zatim unutar istog imena po starosti). Sastaviti funkciju koja realizuje operaciju unije dva skupa i vraća uniju i 1 ili 0 prema tome da li je operacija uspešno obavljena ili ne.

2. Treba da obradimo datoteku **spisak** sa sledećom strukturom
 - na početku je broj **n** objekata koje sadrži datoteka, a za njim sledi
 - niz od **n** objekata sa strukturom **Osoba** (iz prethodnog zadatka)

Sastaviti funkciju koja čita i-tu osobu iz datoteke i vraća podatke o njoj .

- U datoteci **program.c** nalazi se tekst programa na C-u. Sastaviti program koji formira listu svih deklariranih promenljivih u programu **program.c** i ispisuje ih u leksikografskom poretku. (Promenljive su deklarirane isključivo deklaratorima char, int, float i u tekstu **program.c** nema ni pokazivača, ni funkcija). Koristiti dinamičke strukture podataka.
- Sastaviti funkciju koja kao argumente uzima niske x i y i utvrđuje da li je x prefiks niske y. Ako jeste vraća kao povratnu vrednost 1, inače 0, a preko liste argumenata prefiks ("" ako ga nema).
- Sastaviti program koji ispisuje elemente Fibonačijevog niza zadanog sledećim formulama

$$f_1 = 1, f_2 = 2, f_n = f_{n-1} + f_{n-2} (n > 2)$$

Proceniti najveću vrednost za n ako se f deklarise kao **short int** .

Osnovi programiranja, januar 2007. - grupa B

- Definišimo tipove Osoba i Skup na sledeći način:

```
struct osoba {
    char ime[MAXIME];
    unsigned int starost;
}
struct skup {
    unsigned int dim;
    Osoba niz[MAX];
}
typedef struct osoba Osoba;
typedef struct skup Skup;
```

gde je MAXIME maksimalna dužina karakterske niske, a MAX maksimalan broj elemenata skupa. Pretpostavlja se da je niz u strukturi skup sortiran u rastućem redosledu (prvo po imenu, a zatim unutar istog imena po starosti). Sastaviti funkciju koja ispituje da je jedan skup podskup drugog skupa i vraća 1 ako jeste, 0 inače.

- Treba da obradimo datoteku **spisak** sa sledećom strukturom

- na početku je broj n objekata koje sadrži datoteka, a za njim sledi
- niz od n objekata sa strukturom Osoba (iz prethodnog zadatka)

Sastaviti funkciju koja dodaje osobu na kraj datoteke.

- U datoteci **program.c** nalazi se tekst programa na C-u. Sastaviti program koji formira listu svih izraza s desne strane operatora dodele u tekstu **program.c** i ispisuje ih u leksikografskom poretku. Pod izrazom se podrazumeva u ovom zadatku niska omeđena karakterima = i ; (U tekstu **program.c** nema ni pokazivača, ni funkcija). Koristiti dinamičke strukture podataka.
- Sastaviti funkciju koja kao argumente uzima niske x i y i utvrđuje da li je x sufiks niske y. Ako jeste vraća kao povratnu vrednost 1, inače 0, a preko liste argumenata sufiks ("" ako ga nema)
- Sastaviti program koji ispisuje prvih 10 elemenata Fibonačijevog niza zadanog sledećim formulama

$$f_1 = "a", f_2 = "ab", f_n = f_{n-1} * f_{n-2} (n > 2)$$

gde * označava operator dopisivanja niski.

Osnovi programiranja, oktobar 2006. - prva grupa

1. U ravni je zadato n krugova trojkama $\{x_i, y_i, r_i\}$, $i = 0, 1, \dots, n-1$; pri tome su x_i, y_i koordinate centra, a r_i veličina poluprečnika kruga sa indeksom i .
Napisati funkciju kojoj su argumenti dve trojke (koje određuju dva kruga), a koja vraća 1 (odnosno 0) ako se ova dva kruga seku (odnosno ako se ne seku); pretpostavlja se da krug obuhvata i kružnicu.
Napisati program koji sa standardnog ulaza učitava broj $n \leq 100$ i trojke $\{x_i, y_i, r_i\}$, $i = 0, 1, \dots, n-1$, a zatim ispisuje poruku da li se neka dva među krugovima (određenim ovim trojkama) seku.
2. *List* stabla je čvor koji nema ni jednog sina. Napisati funkciju kojoj je argument pokazivač na koren binarnog stabla, a koja vraća broj listova u stablu.
3. Napistati program koji sa standardnog ulaza učitava brojeve m, n , a zatim u narednih m linija po n elemenata celobrojne matrice. Posle toga program treba da odštampa redni broj one vrste matrice koja ima najveći zbir elemenata; redni broj prve vrste je 1.
4. Napisati funkciju sa jednim celobrojnim argumentom n koja vraća 1 (odnosno 0) ako dekadne cifre broja n čine (odnosno ne čine) rastući niz.
5. *Palindrom* je niska koja glasi isto kad se čita spređa, kao i od pozadi — na primer niska "anavolimilovana". Napisati funkciju kojoj je argument niska s , a koja vraća dužinu najdužeg palindroma sadržanog u s . Na primer, za $s = \text{"xyzanavolimilovanatuv"}$ funkcija treba da vrati broj 15.

Osnovi programiranja, oktobar 2006. - druga grupa

1. U ravni je zadato n kvadrata (sa stranicama paralelnim osama) trojkama $\{x_i, y_i, a_i\}$, $i = 0, 1, \dots, n-1$; pri tome su x_i, y_i koordinate centra, a a_i veličina stranice kvadrata sa indeksom i . Napisati funkciju kojoj su argumenti dve trojke (koje određuju dva kvadrata), a koja vraća 1 (odnosno 0) ako se ova dva kvadrata seku (odnosno ako se ne seku); pretpostavlja se da kvadrat obuhvata i granicu (stranice). Napisati program koji sa standardnog ulaza učitava broj $n \leq 100$ i trojke $\{x_i, y_i, a_i\}$, $i = 0, 1, \dots, n-1$, a zatim ispisuje poruku da li se neka dva među kvadratima (određenim ovim trojkama) seku.
2. *Unutrašnji čvor* stabla je čvor koji ima bar jednog sina. Napisati funkciju kojoj je argument pokazivač na koren binarnog stabla, a koja vraća broj unutrašnjih čvorova stabla.
3. Napistati program koji sa standardnog ulaza učitava brojeve m, n , a zatim u narednih m linija po n elemenata celobrojne matrice. Posle toga program treba da odštampa redni broj one kolone matrice koja ima najveći zbir elemenata; redni broj prve kolone je 1.
4. Napisati funkciju sa jednim celobrojnim argumentom n koja vraća 1 (odnosno 0) ako dekadne cifre broja n čine (odnosno ne čine) opadajući niz.
5. *Palindrom* je niska koja glasi isto kad se čita spređa, kao i od pozadi — na primer niska "anavolimilovana".
Napisati funkciju kojoj je argument niska s , a koja vraća dužinu najdužeg palindroma sadržanog u s .
Na primer, za $s = \text{"xyzanavolimilovanatuv"}$ funkcija treba da vrati broj 15.

Osnovi programiranja, septembar 2006. (grupa A)

1. Sastaviti program koji izračunava i ispisuje dužine rečenica datog teksta izražene brojem karaktera (simbol. !?... obeležava kraj rečenice ako se pojavi ispred velikog slova latinske abecede). Tekst se učitava iz datoteke ulaz.txt. Maksimalna dužina rečenice je 100.
2. Sastaviti program koji omogućava korisniku da unosi sa tastature niz celih pozitivnih ili negativnih brojeva kojih nema više od N. Za svaki uneti broj, negativne vrednosti se upisuju na uzastopne lokacije od početka niza, a pozitivne na lokacije počev od kraja niza. Unos se završava kada se unese 0 ili kada je niz popunjen. Kada je unos završen, program ispisuje sadržaj popunjenog dela niza.
3. Sastaviti funkciju koja iz date liste L izbacuje sva ponovljena pojavljivanja njenih elemenata. Npr. rezultat primene funkcije na listu $L=(b,a,b,c,a,c,a)$ je $L=(b,a,c)$.
4. Neka se u datoteci f nalazi sadržaj nekog kataloga u datotečkom sistemu. Uz ime svake datoteke, naveden je i podatak o datumu poslednjeg pristupa datoteci kao jedan ceo broj (koji izražava broj proteklih sekundi od fiksiranog datuma). Sastaviti program koji čita podatke iz datoteke f i formira od njih uređeno binarno drvo. Svaki čvor u drvetu označava datoteku i sadrži podatke o imenu datoteke i datumu poslednjeg pristupa. Sastaviti funkciju koja obilazi drvo i ispisuje sve datoteke koje su nastale pre datuma zadatog kao argument funkcije.
5. Sa ulaza se učitava preusmeravanjem tekst T koji se sastoji od ascii-karaktera. Sastaviti funkciju koja pronalazi sve reči (=niske alfabetskih karaktera između separatora), a u kojima se pojavljuju bar jednom svi vokali (a, e, i, o, u).

Osnovi programiranja, jun 2006.

1. Visina nepraznog binarnog stabla jednaka je dužini najdužeg puta od korena do nekog lista: visina praznog stabla je -1. Napisati funkciju kojoj je argument pokazivač na koren binarnog stabla, a koja vraća visinu tog stabla, kao i program koji testira ovu funkciju.
2. Datoteka čije se ime učitava iz komandne linije sadrži spisak različitih reči nekog teksta (uređen prema ascii poretku), zajedno sa frekvencijom njihovog pojavljivanja. Napisati program koji čita sadržaj te datoteke i ispisuje na standardni izlaz frekvenciju pojavljivanja zadate reči (učitava se sa standardnog ulaza). Broj različitih reči nije vići od 1000. Za pretraživanje koristiti
 - (a) bibliotečku funkciju `bsearch`
 - (b) sopstvenu rekurzivnu funkciju binarnog pretraživanja
3. Napisati funkciju `int f(int n)` koja za dati prirodan broj `n` vraća zbir oktalnih cifara broja `n`.
4. Svaki red datoteke ULAZ.TXT sadrži ime, prezime i korisničko ime studenta na serveru ALAS (zabrisano u formatu: dvoslovna oznaka smeru, poslednje dve cifre godine upisa, četvorocifren broj indeksa, npr. mr050007, ml040950, aa030034). Napisati program koji generiše HTML datoteku generacija05.htm koja sadrži tabelu sa podacima o studentima upisanim 2005. godine.
5. Napisati program koji sa standardnog ulaza unosi pozitivan ceo broj `n` i ispisuje na standardni izlaz sumu bitova broja `n` na neparnim pozicijama. Smatrati da se bit najmanje težine nalazi na poziciji 0.

Osnovi programiranja, drugi kolokvijum 2006.(I grupa)

1. Napisati funkciju za izračunavanje sume $\sum \frac{(-1)^n x^{2n} 2^{n+1}}{n! 3^n}$ za $n=1$ do ∞ sa zadatom tačnošću ϵ . Iz standardne ulazne datoteke učitavaju se realni brojevi x i ϵ .
2. Napisati program za izračunavanje koeficijenata polinoma $S(x) = (x - a)P_n(x) + Q_m(x)$. Dati su: realan broj a , nenegativni celi brojevi m i n i koeficijenti polinoma $P_n(x)$ i $Q_m(x)$.
3. Napisati program za:
 - (a) formiranje orijentisanog grafa od n čvorova;
 - (b) za par datih čvorova (čvorovi su zadati rednim brojevima) ispitati da li je jedan čvor dostupan iz drugog.

Osnovi programiranja, drugi kolokvijum 2006.(II grupa)

1. Napisati funkciju koja za date prirodne brojeve k i n vraća zbir k -tih stepena prvih n prirodnih brojeva.
2. Napisati program za izračunavanje koeficijenata polinoma $T_n(x)$ (n je dati prirodan broj) ako je poznato da važi $T_0(x) = 1$, $T_1(x) = x$, $T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x)$.
3. Napisati program za:
 - (a) formiranje orijentisanog grafa od n čvorova;
 - (b) nalaženje svih čvorova datog grafa koji nisu dostupni iz zadatog čvora.

Osnovi programiranja, februar 2006. - prva grupa

1. Ime datoteke zadaje se iz komandne linije. Napisati program koji ispisuje sadržaj datoteke na sledeći način: redni broj prvog znaka u liniji, a zatim osam po osam znakova u redu, i to heksadecimalno i "karakterski" kao u donjem primeru:

```

0  23 69 6E 63 6C 75 64 65      #include
8  20 3C 73 74 64 69 6F 73      <stdio.h
16 68 3E 0D 0A 23 69 6E 63      > #incl
```

2. Definišemo strukturu VREME na sledeći način:

```

typedef struct{
int sat, min, sek;
} VREME;
```

- (a) Napisati funkciju sa protipom `VREME *napravi(int sat, int min, int sek)` koja dinamički alokira memorijski prostor u koji će smestiti strukturu VREME, inicijalizovanu vrednostima koje se prenose kao parametri. Funkcija vraća pokazivač na kreiranu strukturu.
 - (b) Sastaviti funkciju sa prototipom `void plus(VREME *t)` koja povećava za jednu sekundu vreme predstavljano strukturom t .
3. Napisati program koji za dato $n \leq 15$ ispisuje prvih n redova trougla od Stirlingovih brojeva I vrste $s(n, m)$, $1 \leq m \leq n$. Stirlingovi brojevi I vrste zadaju se rekurentnom relacijom

$$s(n+1, m) = \begin{cases} -ns(n, m), & m = 1 \\ s(n, m-1) - ns(n, m), & 1 < m \leq n \\ s(n, m-1), & m = n+1 \end{cases}$$

pri čemu je $s(1, 1) = 1$. Koristiti jedan jednodimenzionalni niz. Ispis treba da bude sledećeg oblika:

```

1
-1 1
2 -3 1
-6 11 -6 1
24 -50 35 -10 1
.....

```

4. Napisati funkciju sa jednim argumentom `n` tipa `int` koja vraća razliku broja jedinica na parnim i neparnim pozicijama u binarnom zapisu argumenta.

PRIMER: za $n = 19 = (10011)_2$ izlaz je 1.

5. Grupa od n plesača (na čijim kostimima su u smeru kazaljke na satu redom brojevi od 1 do n) izvodi svoju plesnu tačku tako što formiraju krug iz kog najpre izlazi k -ti plesač (odbrojava se počev od plesača označenog brojem 1 u smeru kretanja kazaljke na satu). Preostali plesači obrazuju manji krug iz kog opet izlazi k -ti plesač (odbrojava se počev od sledećeg suseda prethodno izbačenog, opet u smeru kazaljke na satu). Izlasci iz kruga se nastavljaju sve dok svi plesači ne budu isključeni. Celi brojevi n, k ($k < n$) se učitavaju sa standardnog ulaza. Napisati program koji će na standardni izlaz ispisati redne brojeve plesača u redosledu napuštanja kruga.

PRIMER: za $n = 5, k = 3$ redosled izlaska je 3 1 5 2 4.

Osnovi programiranja, februar 2006. - druga grupa

1. Imena dveju datoteka iste veličine zadaju se iz komandne linije. Napisati program koji upoređuje sadržaje datoteka. Ako je i -ti znak u prvoj datoteci a_i , a i -ti znak u drugoj datoteci b_i , onda program izračunava znakove

$$c_i = \begin{cases} a_i, & \text{ako } a_i = b_i, \text{ a } a_i \text{ nije kontrolni znak - sa ASCII kodom } < 32 \\ ', ', & \text{ako } a_i = b_i, \text{ a } a_i \text{ jeste kontrolni znak} \end{cases}, \text{ ako } a_i \neq b_i$$

Znakove c_i program ispisuje na standardni izlaz, po 16 znakova u jednom redu, pri čemu svaki red počinje rednim brojem prvog znaka u redu.

datoteka 1:	datoteka 2:	izlaz:
Imena dveju dato	Imena dve datote	1 Imena dve.....t.
teka iste velici	ke iste velici	17 .e..... velici
ne zadaju se iz	ne zadaju se iz	33 ne zadaju se iz

2. Definišemo strukturu VREME na sledeći način:

```

typedef struct{
    int sat, min, sek;
} VREME;

```

- (a) Napisati funkciju sa protipom `VREME *napravi(int sat, int min, int sek)` koja dinamički alokira memorijski prostor u koji će smestiti strukturu VREME, inicijalizovanu vrednostima koje se prenose kao parametri. Funkcija vraća pokazivač na kreiranu strukturu.
- (b) Sastaviti funkciju sa prototipom `void plus(VREME *t)` koja povećava za jednu sekundu vreme predstavljano strukturom `t`.

3. Napisati program koji za dato $n \leq 15$ ispisuje prvih n redova trougla od Stirlingovih brojeva II vrste $S(n, m)$, $1 \leq m \leq n$. Stirlingovi brojevi II vrste zadaju se rekurentnom relacijom $S(n, k) = S(n-1, k-1) + kS(n-1, k)$, $1 < k < n$ pri čemu je $S(n, 1) = S(n, n) = 1$. Koristiti jedan jednodimenzionalni niz. Ispis treba da bude sledećeg oblika:

1					
1	1				
1	3	1			
1	7	6	1		
1	15	25	10	1	
1	31	90	65	15	1
.....					

4. Napisati funkciju sa jednim argumentom `n` tipa `int` koja vraća razliku broja jedinica na 16 viših i 16 nižih pozicija (koeficijenti uz $2^0, 2^1, \dots, 2^{15}$) u binarnom zapisu argumenta. Pretpostaviti da je argument veličine 4 bajta (32 bita).

PRIMER: za $n = 7 \times 2^{16} + 3 = (111000000000000011)_2$ izlaz je 1.

5. Na osnovu niza `a` dužine `n`, koji sadrži neku permutaciju brojeva $0, 1, \dots, n-1$, može se izračunati niz `b` iste dužine na sledeći način:

- `b[0]` je indeks broja 0 u `a`; 0 se briše iz `a`; dužina `a` postaje $n-1$;
- `b[1]` je indeks broja 1 u `a`; 1 se briše iz `a`; dužina `a` postaje $n-2$;
- `b[2]` je indeks broja 2 u `a`; 2 se briše iz `a`; dužina `a` postaje $n-3$;
- ...

Napisati funkciju `void tranperm(int n, int a[], int b[])` koja za dati niz `a` (permutaciju) izračunava niz `b`. Pri tome treba izbeći pomeranja članova niza `a`.

PRIMER: za $n = 5$, `a={3,5,0,4,2,1}` rezultat treba da bude `b={2,4,3,0,1,0}`

Zadatak 1 januar 2006. (I grupa) Napisati funkciju `int triplcmp(const char *s, const char *t)` za poređenje, prema dekadnoj vrednosti, dva heksadekadna tripleta `s` i `t` kojima su predstavljene dve boje RGB modela (heksadekadni triplet je oblika `#xxxxxx`, gde je `x` - heksadekadna cifra). Funkcija treba da vrati vrednost `-1` ako je `s < t`, `0` ako je `s = t` i `1` ako je `s > t`. Na primer, za `s=#FFFFFF`, `t=#aa00ee`, funkcija treba da vrati vrednost 1. (Za triplet `#aa00ee` dekadna vrednost je $10 \cdot 16^5 + 10 \cdot 16^4 + 14 \cdot 16 + 14 = 11.141.358$.)

Zadatak 2 januar 2006. (I grupa) Napisati program koji će iz datoteke `seminarski.htm` prepisati nazive međusobno različitih etiketa (bez atributa) u binarno stablo pretrage, a na standardni izlaz ispisati ukupan broj listova drveta. Pretpostaviti da naziv etikete nije duži od 30 karaktera.

Zadatak 3 januar 2006. (I grupa) Napisati funkciju koja za celobrojni niz dimenzije `n`, proverava da li među elementima niza postoje neka dva koja su jednaka.

Zadatak 4 januar 2006. (I grupa)

1. Napisati funkciju `void propol (int n, double a[], int m, double b[], int *k, double c[])` čiji su argumenti `a` i `b` nizovi koeficijenata polinoma stepena `n` i `m`, redom. Funkcija izračunava elemente niza `c` koeficijenata polinoma koji se dobije množenjem polinoma `a` i `b`, i stepen `k` proizvoda.
2. Napisati program koji iz datoteke `ulaz.txt` učitava dva polinoma (stepen prvog polinoma, pa njegovi koeficijenti, počev od slobodnog člana; stepen drugog polinoma, pa njegovi koeficijenti), izračunava njihov proizvod i na standardni izlaz štampa stepen i koeficijente proizvoda.

Zadatak 5 januar 2006. (I grupa) Neka je broj n_1 proizvod cifara datog broja `n`, broj n_2 proizvod cifara broja $n_1, \dots$, broj n_k proizvod cifara broja n_{k-1} , pri čemu je `k` najmanji prirodan broj za koji je n_k jednocifren. Napisati funkciju koja za dato `n` izračunava `k`. Na primer, vrednosti ove funkcije od 10, 25, 39 su redom 1, 2, 3.

Zadatak 6 januar 2006. (II grupa) Napisati funkciju koja u datom celobrojnom nizu A dužine n pronalazi (ako postoji) takav par indeksa (i, j) da je zbir članova niza sa indeksima od i do j jednak zadatom broju m .

Zadatak 7 januar 2006. (II grupa) Napisati program koji će iz datoteke čije se ime unosi kao argument komandne linije prepisati nazive međusobno različitih zatvorenih etiketa u binarno stablo pretrage, a na standardni izlaz ispisati dubinu stabla. Pretpostaviti da zatvorena etiketa počinje sa $" < /"$ i da naziv etikete nije duži od 30 karaktera.

Zadatak 8 januar 2006. (II grupa) Napisati funkciju koja za dve niske koje se prenose kao parametri utvrđuje da li su anagrami ili ne. Dve niske su anagrami ako se sastoje od istog broja istih karaktera. Na primer, niske "anagram" i "ramgana" jesu anagrami, dok "anagram" i "angrm" nisu.

Zadatak 9 januar 2006. (II grupa)

1. Napisati funkciju `void brojanje(int a[], int brojac[], int N)` čiji su argumenti a i $brojac$ celobrojni nizovi dimenzije N . Vrednosti elemenata niza a su između 0 i $N - 1$. Funkcija izračunava elemente niza $brojac$ tako da je $brojac[i]$ jednak broju pojavljivanja broja i u nizu a .
2. Kažemo da je celobrojni niz a dimenzije N permutacija ako sadrži svako i : $0 \leq i < N$. Sastaviti funkciju `int DaLiJePermutacija(int a[], int N)` koja vraća 1 ako je niz a permutacija, a inače 0. (Koristiti funkciju `brojanje`)

Zadatak 10 januar 2006. (II grupa) Hemingovo rastojanje dva cela nenegativna broja jednako je broju cifara u binarnom zapisu tih brojeva, koje su na istim pozicijama a razlikuju se. Na primer, Hemingovo rastojanje brojeva $15 = (1111)_2 = (01111)_2$ i $27 = (11011)_2$ je 2. Napisati funkciju koja izračunava Hemingovo rastojanje dva zadata cela nenegativna broja.

Zadatak 11 I kolokvijum, 18. januar 2006. (I grupa) Napisati program pomoću kojeg se za dati broj n izračunava n -ti član niza $F_n = 3 * F_{n-1} - 2 * F_{n-2} + F_{n-1} * F_{n-2}$ pri čemu je $F_0 = 1$ i $F_1 = 1$. U programu ne koristiti nizove.

Zadatak 12 I kolokvijum, 18. januar 2006. (I grupa)

1. Napisati funkciju `unsigned izdvoj_n(unsigned x, unsigned n)` za izračunavanje broja koji se dobija od n krajnjih desnih bitova broja x . Na primer, ako je $x = 54(00...00110110)_2$, a $n = 3$, tada funkcija treba da vrati broj $6(00...00000110)_2$.
2. Napisati program koji, za učitane vrednosti x , n poziva funkciju `izdvoj_n` na standardni izlaz izdaje rezultat.

Zadatak 13 I kolokvijum, 18. januar 2006. (I grupa) Neka je dat niz X od N nenegativnih celih brojeva. Sastaviti funkciju koja će iz niza X izbacivati sva pojavljivanja broja 0 i popunjavati ta mesta u nizu tako što će se preostali elementi niza pomerati ka početku niza. Odrediti i novu dimenziju N niza X . Npr. ulaz: $N = 10, X = 0 \ 22 \ 11 \ 2 \ 0 \ 17 \ 33 \ 4 \ 0 \ 999 \rightarrow$ izlaz: $N = 7, X = 22 \ 11 \ 2 \ 17 \ 33 \ 4 \ 999$.

Zadatak 14 I kolokvijum, 18. januar 2006. (I grupa) Napisati C program koji kreira i na standardni izlaz izdaje opis HTML tabele sa tri kolone. Zaglavlja kolona su redom niske: n , kvadrat, kub. U prvoj koloni se nalaze vrednosti od 1..15, a u drugoj i trećoj koloni su kvadrati i kubovi tih vrednosti, redom.

Zadatak 15 I kolokvijum, 18. januar 2006. (I grupa) Danas je srebna, 18. januar 2006 godine. Napisati funkciju koja za zadati datum (dan, redni broj meseca, godina, posle 1.1.1900. godine) određuje dan u nedelji. Na primer, za trojku (19,1,2006) funkcija treba da vrati broj 4.

Zadatak 16 I kolokvijum, 18.januar 2006.(II grupa) Napisati program pomoću kojeg se za dati broj n izračunava n -ti član niza $F_n = 2 * F_{n-1} * F_{n-2} - 6 * F_{n-1} + F_{n-2}^2$ pri čemu je $F_0 = 2$ i $F_1 = 3$. U programu ne koristiti nizove.

Zadatak 17 I kolokvijum, 18.januar 2006.(II grupa)

1. Napisati funkciju **unsigned izbacij_n(unsigned x, unsigned n)** za izračunavanje broja koji se dobija brisanjem n krajnjih desnih bitova broja x . Na primer, ako je $x = 54(00...00110110)$, a $n = 3$, tada funkcija treba da vrati broj $48(00...00110000)$.
2. Napisati program koji, za učitane vrednosti x , n poziva funkciju **izdvoj_ni** na standardni izlaz izdaje rezultat.

Zadatak 18 I kolokvijum, 18.januar 2006.(I grupa) Neka je dat niz X od N nenegativnih celih brojeva. Sastaviti funkciju koja će iz niza X izbacivati sva pojavljivanja negativnih brojeva i popunjavati ta mesta u nizu tako što će se preostali elementi niza pomerati ka početku niza. Odrediti i novu dimenziju N niza X . Npr. ulaz: $N = 6, X = 0 \ -2 \ 11 \ 0 \ -333 \rightarrow$ izlaz: $N = 4, X = 0 \ 11 \ 0 \ 0$.

Zadatak 19 I kolokvijum, 18.januar 2006.(II grupa)

1. Napisati funkciju **int palindrom(int broj)** koja proverava da li je broj palindrom i vraća vrednost 1 ako jeste, 0 ako nije. Na primer, brojevi 1, 44, 121, 112211, 12321, i 5665 jesu palindromi, a brojevi 123, 67, 8908 nisu.
2. Napisati program koji proverava da li je uneti broj palindrom.

Zadatak 20 I kolokvijum, 18.januar 2006.(II grupa) Napisati funkciju koja na standardni izlaz ispisuje sve linkove iz HTML dokumenta sadržanog u datoj nisci s . Na primer, u delu niske s

```
<A HREF="http://www.bg.ac.yu">
<IMG SRC="/images/univplavi.gif" ALT="Univerzitet u Beogradu" BORDER=0></A>
```

funkcija treba da pronade

<http://www.bg.ac.yu>.

Zadatak 21 I kolokvijum, februar 2005.

1. Napisati funkciju koja ispituje da li dve niske (koje se prenose kao parametri funkcije) su anagrami. Anagrami su niske koje se sastoje od istih karaktera. Npr. vetar, trave, verat su anagrami.
2. Napisati program koji testira funkciju iz prvog dela.

Zadatak 22 I kolokvijum, februar 2005. Napisati program koji učitava sa standardnog ulaza dve niske sa ne više od 80 karaktera u svakoj i prirodan broj k i ispisuje na standardni izlaz poruku da li se prva niska dobila cikličnim pomeranjem druge niske za k mesta. Na primer za $k=3$, niska "CDEAB" se dobila cikličnim pomeranjem niske "ABCDE"

Zadatak 23 I kolokvijum, februar 2005. Napisati program koje će učitati sa tastature broj s (unsigned int) i brojeve m i n (int), pri čemu je $0 \leq m \leq n < \text{sizeof}(\text{unsigned}) * 8$ i formirati vrednost d (unsigned int) u kojoj je bit na poziciji i jednak 1 akko je $m \leq i \leq n$ (pozicije se broje od nule sdesna na levo). Program treba da na standardnom izlazu ispiše broj koji se dobija od s postavljanjem na 0 svih bitova koji su u d jednaki 1.

Zadatak 24 Septembar, 2005. Napisati program koji na standardni izlaz ispisuje naziv (BEZ ATRIBUTA) najčešće korišćene etikete u datoteci ulaz.htm. Ako ima više takvih, ispisati ma koju. Koristiti uređeno binarno stablo. Pretpostaviti da je ulazna datoteka sintaksno korektna.

Zadatak 25 Septembar, 2005.

1. Napisati funkciju `int poredi(char* p, char* d)` koja vraća -1 ukoliko je $p < d$, 0 ukoliko je $p == d$ a 1 ako je $p > d$, pri čemu su p i d dva velika cela neoznačena broja zadata nizom (niskom) svojih cifara.
2. Argumenti komandne linije su imena dve datoteke koje sadrže cele neoznačene brojeve (po jedan u svakoj liniji, sa maksimalno 1000 cifara) sortirane u rastućem poretku po numeričkoj vrednosti. Broj linija nije unapred poznat.

Napisati program koji upisuje sadržaj ove dve datoteke u datoteku `Spoj.txt` tako da i ona bude sortirana.

Zadatak 26 Septembar, 2005. Napisati program koji sa standardnog ulaza učitava pozitivan ceo broj, a na standardni izlaz ispisuje vrednost tog broja sa razmenjenim vrednostima bitova na poziciji i , j . Pozicije i , j se učitavaju kao parametri komandne linije. Smatrati da krajnji desni bit binarne reprezentacije je 0-ti bit. Pri rešavanju nije dozvoljeno koristiti pomoćni niz niti aritmetičke operatore $+$, $-$, $/$, $*$, $\%$.

Zadatak 27 Septembar, 2005. Sa standardnog ulaza se učitava niz od n ($n < 100$) tačaka u ravni takvih da nikoje tri tačke nisu kolinearne. Tačke se zadaju parom svojih koordinata (celi brojevi). Ispitati da li taj niz tačaka određuje konveksni mnogougao i rezultat ispisati na standardni izlaz.

Zadatak 28 Jun, 2004. Datoteka `Matrice.txt` sadrži dve celobrojne kvadratne matrice. U datoteci su prvo zapisane dimenzije matrica n i m ($n > m$) a zatim i elementi prvo jedne a zatim i druge matrice. Napisati program koji proverava da li se manja matrica sadrži u većoj. Matrica se sadrži u matrici veće dimenzije ukoliko postoji podmatrica veće matrice identična manjoj matrici tj. ako postoji blok veće matrice dimenzije $m \times m$ čiji su elementi jednaki elementima manje matrice na odgovarajućim pozicijama. npr. U matrici

```
1 1 1
2 2 2
3 3 3
```

se sadrži matrica

```
1 1
2 2
```

a ne sadrži matrica

```
1 1
3 3
```

Zadatak 29 Jun, 2004. Napisati funkciju koja računa multiplikativnu otpornost datog pozitivnog broja. Multiplikativna otpornost se računa na sledeći način $n_0 = n$, n_k je jednak proizvodu cifara broja n_{k-1} , $k = 1, 2, \dots$, multiplikativna otpornost je najmanje k za koje je n_k jednocifren broj. Napisati program koji iz datoteke čije se ime zadaje kao prvi argument komandne linije čita brojeve, gde su brojevi zapisani po jedan u svakom redu i u drugu datoteku čije se ime zadaje kao drugi argument komandne linije upisuje red po red date brojeve i njihovu multiplikativnu otpornost.

Zadatak 30 Jun, 2004. Napisati funkciju koja kao argumente prihvata dve niske i proverava da li se prva od zadatih niski može dobiti cikličnim pomeranjem karaktera druge niske.

Zadatak 31 Jun, 2004. Igrupa Data je datotka `brojevi.txt` koja sadrži cele brojeve, po jedan u svakom redu.

1. Napisati funkciju koja iz zadate datoteke učitava brojeve i smešta ih u listu.
2. Napisati funkciju koja u jednom prolazu kroz zadatu listu celih brojeva pronalazi maksimalan strogo rastući podniz.
3. Koristeći funkcije pod a) i b) napisati program koji u datoteku `Rezultat.txt` upisuje nađeni strogo rastući podniz.

Zadatak 32 Jun, 2004. IIgrupa Imena dve datoteke koje sadrže cele brojeve unose se kao argumenti komandne linije.

1. Napisati funkciju koja iz datoteke učitava brojeve i smešta ih u rastuće uređjenu listu (listu čiji su elementi poredjani u rastućem poretku).
2. Napisati funkciju koja od brojeva dve rastući uređjene liste formira treću koja je takodje rastući uređjena.
3. Koristeći funkcije pod a) i b) napisati program koji sortira brojeve (u rastućem poretku) koji se nalaze u datotekama čija su imena argumenti komandne linije i upisuje ih u datoteku `Rezultat.txt`.

Zadatak 33 Prvi kolokvijum za II tok 2004.godine - rad na racunaru Napisati program koji generiše HTML fajl `Boje.html` koji sadrži tabelu boja. Tabela treba da ima 8 kolona pri čemu ćelije neparnih kolona treba da sadrže heksadekadnu vrednost boje i to u formatu `ROGOBO` a ćelije odgovarajuće parne kolone treba da budu obojene tom bojom.

Zadatak 34 Prvi kolokvijum za II tok 2004.godine - rad na racunaru Sa standardnog ulaza se unose veliki, celi, neoznačeni brojevi sa najviše 100 cifara. Ovih brojeva ima manje od 100 ali njihov broj nije unapred poznat. Napisati program koji sabira ovako unete brojeve i na standardni izlaz ispisuje njihov zbir.

Napomena : Svaki broj se unosi u posebnom redu a potrebno je voditi računa o korektnosti ulaznih podataka.

Zadatak 35 Prvi kolokvijum za II tok 2004.godine - rad na papiru Sa standardnog ulaza se unose dve niske koje predstavljaju elemente dva skupa. Skupovi nemaju više od 20 elemenata. Napisati program koji na standardni izlaz ispisuje niske koje predstavljaju:

1. presek,
2. uniju i
3. razliku

elemenata dva skupa.

Zadatak 36 Drugi kolokvijum za II tok 2004.godine - rad na računaru Sa standardnog ulaza se unosi ime datoteke čiji prvi red sadrži dimenziju celobrojne kvadratne matrice n ($n > 100$), a ostali redovi elemente matrice (vrstu po vrstu). Formirati niz b dimenzije n čiji je prvi član suma elemenata glavne dijagonale, drugi suma elemenata na prvoj donjoj dijagonalinoj paraleli (nju čine elementi odmah ispod glavne dijagonale), treći element suma druge donje dijagonaline paralele, itd. Ispisati niz na standardni izlaz. Sve greške štampati na standardni izlaz za greške.

Zadatak 37 Drugi kolokvijum za II tok 2004.godine - rad na računaru Napisati program koji iz tekstualne datoteke čiji je put dat u argumentu komandne linije učitava različite prirodne brojeve i :

1. dodaje ih redom u uređjeno binarno stablo

2. u dobijenom drvetu izračunava dužinu najdužeg puta od korena do nekog lista i
3. štampa u rastućem poretku (bez ponavljanja) sve brojeve koji su nalaze na putevima te dužine od korena do listova.

Zadatak 38 Januar, 2002. Datoteka "izrazi.dat" sadrži izraze koji se sastoje od celobrojnih i realnih konstanti i operacija $+$, $-$, $*$, $/$ i zapisani su u inverznoj poljskoj notaciji (operandi pa operacija). Na primer, izraz $(1+2)/(3-4)$ zapisan je kao $1\ 2\ +\ 3\ 4\ -\ /\$, a izraz $21+7*6$ kao $21\ 7\ 6\ *\ +$. Svaki izraz je u datoteci zapisan u novom redu i podrazumeva se da su izrazi sintaksno ispravni. Napisati program koji izračunava i štampa na ekran vrednosti svih izraza u datoteci. Rešenje napisati modularno i obavezno ga komentarisati.

Zadatak 39 Januar, 2002. Program sa standardnog ulaza učitava raspored 8 topova na šahovskoj tabli. Raspored se sastoji od 8 linija sa po 8 brojeva u svakoj liniji. Svaka linija odgovara jednom redu table, a svaki broj jednom polju. Broj ima vrednost 0 ako na datom polju nema topa i vrednost 1 ako na datom polju postoji top. Program treba da ispita da li je uneseni raspored validan (tj. da li je svaki učitani broj 1 ili 0 i da li ima ukupno 8 topova na tabli), kao i da odredi da li se u datom rasporedu neka dva topa tuku (topovi se tuku ukoliko se nalaze u istom redu ili istoj koloni table). Program treba da ispiše na standardnom izlazu "raspored nije validan" ukoliko ulazni podaci nisu dobri, a u suprotnom "ne tuku se" ukoliko je raspored takav da se nijedan par topova međusobno ne tuče, odn. "tuku se" ukoliko ima topova koji se tuku

Zadatak 40 Januar, 2002. Sa standardnog ulaza se učitava u jednoj liniji prirodan broj n , a potom i linije teksta do markera kraja fajla. Napisati program koji štampa n reči koje se najčešće pojavljuju i to počev od najfrekventije reči. Uz reč odštampati i broj pojava. Reč je po definiciji ma koji niz karaktera koji ne sadrži blanko, tabulator, znak za novi red. Sve poruke o greškama ispisati na standardnom izlazu za poruke o grešci.

Zadatak 41 Januar, 2002. Napisati program koji čita ulaznu datoteku ulaz.htm i štampa na standardni izlaz samo linije koje imaju 70 karaktera van etiketa, pri čemu se tekst markiran u obliku `&entity;` (npr. `<`; `&`;) ili `&#number;` (npr. `č`;) broji kao 1 karakter. Programi da budu pisani čitko i izdašno komentarisani.

Zadatak 42 Februar, 2002. Neka se relacija nad nekim skupom elemenata opisuje kvadratnom matricom na sledeći način: ako je u preseku i -te vrste i j -te kolone 1, to znači da je i -ti element u relaciji sa j -tim, ako je 0 to znači da nije u relaciji. Sa standardnog ulaza zadaje se najpre dimenzija ovakve matrice, pa zatim elementi matrice, jedan za drugim, po vrstama. Dimenzija matrice nije ograničena. napisati program koji, pošto proveri korektnost ulaza, za ovako zadatu relaciju ispituje njenu refleksivnost, simetričnost i tranzitivnost i odgovarajuće poruke štampa na ekran.

Zadatak 43 Februar, 2002. Datoteka prica.txt sadrži niz reči (reč je niz karaktera koji ne sadrži blanko, tabulator ili znak za novi red). Sa standardnog ulaza učitava se jedna reč. Nijedna reč, nema više od 20 karaktera. Napisati program koji broji i štampa na ekran koliko se puta data reč pojavila u datoteci, ako se zna da su neke reči pogrešno unete. Smatramo da je neka reč jednaka učitanoj i onda kada:

- je zamenjeno jedno slovo nekim drugim slovom
- ili je izostavljeno jedno slovo u jednoj od te dve reči

Zadatak 44 Februar, 2002. Napisati program koji za dva data pravougaonika R_0 i R_1 sa stranicama paralelnim koordinatnim osama izračunava i na standardni izlaz ispisuje površine njihovih unija ($R_0 \cup R_1$), presjeka ($R_0 \cap R_1$) i razlike ($R_0 \setminus R_1$). Pravougaonici se učitavaju sa standardnog ulaza i zadati su koordinatama donjeg levog, odn. gornjeg desnog tjemena. Ove koordinate su realni brojevi. Za čuvanje podataka koji određuju neki pravougaonik deklarirati odgovarajuću strukturu. Sve operacije nad pravougaonikom (ili pravougaonicima) izdvojiti u posebne funkcije. Primjer: za pravougaonike zadate na sledeći način:

```
10 20 30 40
20 30 40 50
```

program treba da ispiše:

```
Povrsina uniije iznosi 700
Povrsina preseka iznosi 100
Povrsina razlike iznosi 300"
```

Zadatak 45 Februar, 2002. U datoteci *tajna.txt* nalazi se riječ dužine ne veće od 20 karaktera. Riječ se sastoji isključivo od malih slova. Napisati program za pogađanje riječi. Program treba da učitava riječ iz datoteke, a zatim da sa standardnog ulaza čita jedno po jedno slovo koja daje korisnik pogađajući da li ih riječ sadrži. Po učitavanju svakog slova program treba da ispiše ona slova u riječi koja su dotad pogodena. Na mjestima ostalih slova treba da budu karakteri *. Voditi računa o mogućnosti da korisnik greškom unese nešto što nije slovo, takode i neko slovo koje je ranije već unosio. Program ne treba da pravi razliku između malih i velikih slova, tj. ako korisnik unese neko veliko slovo, program treba da ga tretira kao malo slovo. Kada sva slova budu pogodena, program treba da ispiše ukupan broj pokušaja. Primjer sesije za slučaj kada je riječ koja se pogađa **zdravo** bi mogao biti:

```
a
***a**
e
***a**
i
***a**
o
***a*o
r
**ra*o
m
**ra*o
b
**ra*o
d
*dra*o
v
*dravo
z
zdravo
Ukupan broj pokusaja: 10
```

Zadatak 46 Februar, 2002. Napisati program koji učitava kvadratnu matricu sa standardnog ulaza čiji su članovi celi brojevi i proverava da li je matrica ortogonalna. Ne koristiti pomoćne matrice! U prvoj liniji nalaze se dimenzija matrice, a zatim se u svakoj liniji nalaze vrste matrice. Elementi unutar vrste su razdvojeni blanko znakovima. Dimenzija matrice nije unapred poznata. Pretpostaviti da su sve linije sem prve u ispravnom formatu i u slučaju greške izdati poruku na standardnom izlazu za poruke o grešci.

Zadatak 47 Februar, 2002. Parametri komandne linije su imena dve datoteke i ceo broj *n*. Napisati program koji poslednjih *n* linija prve datoteke upisuje u drugu datoteku. Može se pretpostaviti da prva datoteka ne sadrži linije duže od 80 karaktera, ali broj linija u datoteci nije unapred ograničen. U slučaju greške izdati poruku na standardnom izlazu za poruke o grešci.

Program komentarisati i programski kod pisati čitko.

Zadatak 48 April, 2002. . Prvi red standardne ulazne datoteke sadrži 2 cela broja manja od 50 koji predstavljaju redom broj vrsta i broj kolona realne matrice A. Svaki sledeći red sadrži po jednu vrstu matrice. Napisati program koji :

1. nalazi sve elemente matrice A koji su jednaki zbiru svih svojih susednih elemenata i štampa ih u obliku (broj vrste, broj kolone, vrednost elementa)
2. nalazi i štampa sve četvorke oblika
(A(i,j), A(i+1,j), A(i,j+1), A(i+1,j+1)) u kojima su svi elementi međusobno različiti.

Zadatak 49 April, 2002. Parametri komandne linije su nazivi 2 datoteke. Prva datoteka sadrži niz reči čiji broj i čija dužina nije ograničena (mogu biti proizvoljno veliki brojevi) . Reč je bilo kakav niz karaktera koji nije blanko, tabulator ili oznaka za kraj reda. Napisati program koji u drugu datoteku prepisuje samo one reči iz prve datoteke koje su parne dužine i koje i počinju i završavaju se slovom. (napomena: obavezno voditi računa o tome da se dužina reči ne može ograničiti!)

Zadatak 50 April, 2002. Napisati program koji ispisuje kalendar za zadati mesec i godinu XX vijeka. Poznato je da je 1. januar 1901. bio utorak. Program prima dva argumenta u komandnoj liniji: broj u intervalu [1, 12] koji predstavlja mesec i broj u intervalu [1901, 2000] koji predstavlja godinu (obavezno proveriti validnost ovih argumenata). Program treba da ispiše kalendar na standardni izlaz i to tako što će u prvom redu biti ispisani mesec (punim imenom) i godina, u narednom redu dvoslovne skraćenice od imena dana, počev od ponedeljka i sa po jednim blanko znakom između skraćenica, a zatim u narednim redovima datumi, pri čemu se za svaki dan odvajaju po 2 mesta u kojima broj treba da je poravnat udesno, a između dana se ostavlja po jedan blanko znak. Tako npr, ako su argumenti koji su zadati u komandnoj liniji 1 1970, ispis treba da ima sledeći oblik:

```

Januar 1970.
Po Ut Sr Ce Pe Su Ne
      1  2  3  4
  5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31

```

Zadatak 51 April, 2002. Napisati program koji učitava sa standardnog ulaza prvo jednu liniju teksta a zatim još jednu liniju sa karakterima koje treba izbaciti iz prve linije. Program treba da izbaci specificirane karaktere iz prve linije i ispiše ono što preostane od iste. Dužina prve linije nije unapred ograničena, tj. za čuvanje te linije treba koristiti listu pri čemu će po jedan karakter biti smešten u svaki element liste. Primjer: ako je unos imao sledeći oblik:

```

Hello, world!
aeiou,

```

program treba da ispiše:

```
Hll wrld!
```

Zadatak 52 April, 2002. . Sastaviti program koji ispisuje $n < 19$ redova Pascalovog trougla koristeći samo 1-dimenzionalni niz i ne koristiti rekurziju. Broj n se zadaje kao jedini sadržaj linije standardnog ulaza. Izveštaj o eventualnim greškama na ulazu ispisati i na standardnom izlazu za poruke o grešci.

Zadatak 53 April, 2002. Argumenti komandne linije su imena tri datoteke. Preve dve datoteke u svakom redu sadrže do 80 cifara i u obe datoteke sadržaj je sortiran strogo rastuće po numeričkoj vrednosti broja predstavljenog tim ciframa. Napisati program koji će spojiti te dve datoteke u treću čiji će sadržaj takode biti sortiran strogo rastuće po numeričkoj vrednosti brojeva koje sadrži.

Zadatak 54 Jun, 2002. Neka je $P = (p_1, p_2, \dots, p_n)$ permutacija brojeva $1, 2, \dots, n$. Napisati PASCAL program koji za učitani prirodan broj $n < 50$ i za učitane tablice inverzije ispisuje odgovarajuću permutaciju. Pod tablicom inverzije permutacije P se podrazumeva niz $S = (s_1, s_2, \dots, s_n)$ u kom je s_i jednako broju elemenata permutacije P koji (u P) stoje levo od broja i , a veći su od broja i .

Zadatak 55 Jun, 2002. Slika je opisana u kvadratnoj matrici tako da elementi koji određuju sliku popunjeni su cifrom 1, a ostali elementi su popunjeni cifrom 0. kao parametar komandne linije zadaje se ime datoteke u čijoj prvoj liniji se nalazi dimenzija matrice koaj opisuje sliku, a zatim u svakoj liniji nalaze se vrste matrice. Elementi unutar vrste su razdvojeni blankom. napisati program koji u svakoj liniji datoteke REPORT.DAT ispisuje poruke o simetričnosti matrice u odnosu na horizontalnu osu, vertikalnu osu, glavnu dijagonalu, sporednu dijagonalu, centar.

Zadatak 56 Jun, 2002. Sprovedena je anketa o popularnosti televizijskih emisija. Broj emisija za koje se glasalo nije veći od 50. Ispitanici su podeljeni u 4 kategorije: mškarci do 30 godina, žene do 30 godina, muškarci stariji od 30 godina, žene starije od 30 godina. Svi su glasali za 3 emisije. Svaka linija u datoteci čije se ime zadaje kao prvo u komandnoj liniji, sadrži podatke o glasanju jednog ispitanika i to sledećim redom: pol ispitanika (m ili z), broj godina, pa zatim šifre emisija za koje je ta osoba glasala. Sifra emisije je niz od najviše 5 karaktera. Napisati program koji u datoteku čije ime se zadaje kao drugo u komandnoj liniji, ispisuje šifre emisija i odgovarajući broj glasova poredane nerstuće po broju osvojenih glasova i to za svaku od kategorija posebno. Emisije za koje se nije glasalo treba preskočiti u ispisivanju.

Zadatak 57 Jun, 2002. Sa standardnog ulaza unosi se najpre jedna linija teksta čija dužina nije ograničena, pa zatim još jedna linija koja sadrži samo karakter koji iz prve linije treba izbaciti. napisati program koji treba da ispiše rezultat odnosno šta je preostalo od linije, pa zatim unošenjem sledećeg karaktera koji se želi izbaciti da ponovi postpak za novodobijenu liniju, i tako dalje po istom principu sve dok se za karakter koji se želi izbaciti ne unese * ili dok od linije ne ostane ništa. Pošto dužina linije nije ograničena, za njeno čuvanje treba koristiti povezanu listu kod koje svaki element čuva po jedno slovo iz linije. Koristiti modularni pristup. Primer jedne sesije bi mogao biti:

```
programiranje
p
rogramiranje
e
rogramiranj
s
rogramiranj
a
rogrmirnj
*
```

Zadatak 58 Jun, 2002. Data je datoteka u kojoj se nalazi tekst čiji su naslovi obeleženi etiketom h. Maksimalna dubina naslova je n, gde se ceo broj n zadaje kao argument komandne linije. Svaka etiketa naslova je zatvorena. (Npr. `<h3>Zadaci za pismeni</h3>`). Jedini komentari u tekstu sadrže oznake broja strane i oblika su `<!--xxxx-->` gde je xxxx najviše četvorocifreni neoznačen broj. Tekst je kodiran bez ikakvih grešaka! Sastaviti program koji iz komandne linije uzima ime gore opisane datoteke i kreira na izlazu datoteku u kojoj se nalazi sadržaj ulaznog teksta. Sadržaj se formira kao niz redova koji sadrže niske obeležene h-etiketama i odgovarajući broj strane. Npr.

ulaz:	izlaz:
Zadaci za pismeni	2.2.3. Zadaci za pismeni.....228
Pitanja za usmeni	2.2.3.1. Pitanja za usmeni.....235

gde su navedeni naslovi uzastopni. Sadržaj ulazne datoteke se mora formirati pre nego što se u nju upiše.

Zadatak 59 Jun, 2002. U tekstualnoj datoteci nalaze se podaci o prijemnom ispitu učenika jedne osnovne škole tako što je u svakom redu navedeno: ime i prezime učenika (niz znakova ne duži od 50 znakova), broj poena na osnovu uspeha (decimalan broj), broj poena na prijemnom ispitu iz matematike (decimalan broj) i broj poena na prijemnom ispitu iz maternjeg jezika (decimalan broj). Za učenika koji osvoji manje od 10 poena ukupno na oba prijemna smatra se da nije položio prijemni. Napisati program na C-u koji na osnovu podataka iz ove datoteke formira i prikazuje rang listu učenika. Rang lista sadrži: redni broj, ime i prezime učenika, broj poena na osnovu uspeha, broj poena na prijemnom ispitu iz maternjeg jezika, broj poena na prijemnom ispitu iz matematike i ukupan broj poena i sortirana je opadajuće po ukupnom broju poena. U rang listi se navode prvo oni učenici koji su položili prijemni a potom učenici koji nisu položili prijemni. Između ove dve grupe staviti horizontalnu liniju (-----). Ime datoteke navodi se kao argument komandne linije.

Zadatak 60 Jun, 2002. Napisati program u C-u koji sa standardnog ulaza učitava cifre n i k , a na standardnom izlazu prikazuje najmanji prirodan broj koji počinje cifrom n i ima svojstvo da se smanjuje k puta kada se cifra n premesti sa početka na kraj. Primer: za $n=3$ i $k=2$ traženi broj je 315789473684210526

Zadatak 61 Jun, 2002. Svaka linija datoteke čije se ime prosleđuje komandnom linijom sadrži po 6 celih brojeva: x_1 , y_1 , x_2 , y_2 , x_3 , y_3 koji predstavljaju redom koordinate temena jednog trougla. Linija u datoteci nema više od 100. Napisati program koji uzimajući u obzir samo trouglove koji su jednakokranični, ispituje da li se oni svi mogu "upisati" jedan u drugi (ako je jedan trougao upisan u drugi njegova temena mogu i ne moraju pripadati stranicama ovog drugog). Odgovarajuću poruku štampati na ekran.

Zadatak 62 Jun, 2002. Data je datoteka u kojoj se nalazi tekst u kom se nazivi institucija koji se satoje od slova engleske abecede i blanka obeležavaju etiketom name i atributom type.

Npr. <name type='institution'>Palata pravde</name> maksimalna dužina naziva institucije je n , gde se ceo broj n zadaje kao argument komandne linije. Jedini komentari u tekstu sadrže oznake broja strane i oblika su <!-- -xxxx- -> gde je xxxx najviše četvorocifreni neoznačen broj. Tekst je kodiran bez ikakvih greški. Sastaviti program koji iz komandne linije uzima ime gore opisane datoteke i kreira na izlazu datoteku index.dat u kojoj se nalazi indeks ulaza koji se formira kao niz redova koji sadrže naziv institucije i broj prve stranice na kojoj se taj naziv pojavio. nazive isntitucija koji se javljaju često (više od m puta, gde se m zadaje kao argument komandne linije) ne unostit u indeks. Program ne trab da pravi razliku između malih i velikih slova.

Zadatak 63 Septembar, 2002. Svaki red datoteke čije se ime zadaje komandnom linijom, sadrži po 3 cela broja: A , B , C (A i B nisu istovremeno jednaki nuli), koji predstavljaju koeficijente prave u ravni $Ax+By+C=0$. Broj redova u datoteci nije veći od 100. Napisati program koji pronalazi i na standardnom izlazu ispisuje sve parove paralelnih pravih, kao i sve trojke pravih koje se seku u jednoj tački. Način prikaza traženih podataka je proizvoljan, ali treba voditi računa o njihovoj preglednosti.

Zadatak 64 Septembar, 2002. Grupa od n plesača (na čijim kostimima su redom brojevi od 1 do n) uvežbava svoju plesnu tačku tako što formiraju krug iz kog će redom izlaziti plesači na sledeći način:

1. počev od plesača označenog brojem 1, a brojeći udesno (ka plesačima sa većim rednim brojevima), izlazi m-ti plesač
2. nakon isključenja, brojanje otpočinje od sledećeg plesača i to u suprotnom smeru, tj. ako se brojalo udesno, počinje se od desnog suseda isključenog plesača i broji se ulevo
3. izlasci iz kruga se nastavljaju sve dok svi plesači ne budu isključeni

Celi brojevi m , n se zadaju kao argumenti komandne linije. Napisati C program koji ispisuje redne brojeve plesača u redosledu napuštanja kruga.

Zadatak 65 Septembar, 2002. N osoba obeleženih brojevima 1, 2, . . . N stoji u krugu. Počev od osobe sa rednim brojem 1 broji se K osoba i K-ta osoba izlazi iz kruga, a potom se nastavlja brojanje preostalih osoba na isti način, počev od prve osobe koja je izašla. Ovo se nastavlja sve dok u krugu ne ostane samo jedna osoba. Napisati program koji sa standardnog ulaza učitava vrednosti za N i K, a na standardnom izlazu prikazuje redosled izlaska ljudi iz kruga i redni broj osobe koja poslednja ostaje. Primer: za N=4 i K=3 redosled izlazaka je 3, 2, 4 i na kraju ostaje 1.

Zadatak 66 Septembar, 2002. Parametar komandne linije je ime datoteke čiji svaki red (izuzev prvog) je oblika `ime_deteta:ime_roditelja`. Prvi red sadrži samo ime jednog roditelja čija su sva deca navedena u narednim redovima u već opisanom obliku. Nije obavezno da se sva deca istog roditelja pojavljuju u uzastopnim redovima i nije unapred poznat ukupan broj roditelja. Jednostavnosti radi, može se smatrati: da sve osobe imaju imena sastavljena od slova engleske abecede, da su sva imena međusobno različita (ignorirajući razliku malih i velikih slova), da svaki roditelj nema više od četvero dece i da redovi datoteke nemaju više od 40 karaktera. Napisati program koji za svaku osobu X formira datoteku (čiji je naziv ime osobe) i koja u svakom redu sadrži imena najbližih stričeva, tetki, ujaka osobe X (misli se na rođenu braću i sestre roditelja osobe X).

Zadatak 67 Septembar, 2002. Slika je opisana u kvadratnoj matrici tako da elementi koji određuju sliku popunjeni su cifrom 1, odnosno cifrom 0. Kao parametar komandne linije zadaje se ime datoteke u čijoj prvoj liniji se nalazi dimenzija matrice koja opisuje sliku, a zatim se u svakoj liniji nalaze vrste matrice. Elementi unutar vrste su razdvojeni blankom. Napisati C program koji, ne koristeći pomoćne matrice, premešta podsliku (čije koordinate gornjeg levog ugla, dužina i širina se zadaju kao argumenti komandne linije) na novu poziciju čiji položaj gornjeg levog ugla se zadaje sa standardnog ulaza. Original i kopija moraju ostati u okvirima polazne matrice. Poruke o eventualnim greškama štampati na standardni izlaz za poruke o grešci.

Zadatak 68 Septembar, 2002. Napisati program koji sa standardnog ulaza učitava cifre pozitivnog celog broja (kojih nema više od 100, a na ulazu su jedna pored druge tj. između cifara nema praznih mesta) a na standardnom izlazu ispisuje najmanji pozitivan ceo broj zapisan istim ciframa. Rezultat ne sme počinjati cifrom nula.

Zadatak 69 Januar, 2002. Neka su u tekstualnoj datoteci LAVIRINT dati podaci o matrici-lavirintu. Prvi red tekstualne datoteke sadrži broj kolona (80) i broj vrsta (25) a u svakom sledećem redu se nalaze podaci o jednoj vrsti matrice: karakter 'Z' označava da odgovarajuće polje matrice predstavlja zid, a karakter 'P' označava prazan prostor. Napisati program koji na standardnom izlazu prikazuje lavirint učitani iz datoteke ali tako da polja zida prikazuje karakterom 'X' a prazna polja blanko karakterom. Program potom učitava koordinate dve pozicije u lavirintu i utvrđuje da li postoji put kroz lavirint od jedne do druge pozicije (kretanje je moguće samo kroz prazna polja i to u četiri pravca - gore, dole, levo i desno). Ako put postoji program ponovo prikazuje lavirint ali tako da na početnoj poziciji umesto blanko karaktera stoji karakter 'A', na krajnjoj karakter 'B', a na svim ostalim poljima na putu karakter 'O'. Ako put ne postoji dati odgovarajuću poruku.

Zadatak 70 Nepoznati rok Sa standardnog ulaza se unosi ime datoteke čiji prvi red sadrži dimenziju celobrojne kvadratne matrice n ($n > 100$), a ostali redovi elemente matrice (vrstu po vrstu). Formirati niz b dimenzije n čiji je prvi član suma elemenata glavne dijagonale, drugi suma elemenata na prvoj donjoj dijagonalnoj paraleli (nju čine elementi odmah ispod glavne dijagonale), treći element suma druge donje dijagonalne paralele, itd. Ispisati niz na standardni izlaz. Sve greške štampati na standardni izlaz za greške.

Zadatak 71 Nepoznati rok Sa standardnog ulaza se unose veliki, celi, neoznačeni brojevi sa najviše 100 cifara. Ovih brojeva ima manje od 100 ali njihov broj nije unapred poznat. Napisati program koji sabira ovako unete brojeve i na standardni izlaz ispisuje njihov zbir. Napomena: Svaki broj se unosi u posebnom redu a potrebno je voditi računa o korektnosti ulaznih podataka.