

The page features a decorative graphic on the right side consisting of three blue circles of varying sizes, each with a lighter blue ring around its center. Two thin blue lines extend from the top left towards the circles, and a larger blue shape is partially visible at the bottom right.

Zadaci sa vežbi iz Programskog jezika „C“

Tehnički Fakultet Čačak, školska 2007/2008 godina

Vežbe održao: Vlade Maksimović

Zadatak 1. Napisati program koji ispisuje sve trocifrene Armstrongove brojeve, Armstrongovi brojevi su oni brojevi koji su jednaki zbiru kubova svojih cifara.

```
#include<stdio.h>

int main(int argc, char *argv[])
{
    int c1,c2,c3,n;
    for(n=100;n<=999;n++){
        c1=n%10;
        c2=n/10%10;
        c3=n/100;
        if(c1*c1*c1+c2*c2*c2+c3*c3*c3==n)
            printf("%d\n",n);
    }
    return 0;
}
```

Zadatak 2. Objašnjenje rada switch-a, šta ispisuje sledeci program.

```
#include<stdio.h>

int main(int argc, char *argv[])
{
    int x = 0,i;
    for(i=0;i<8;i++)
    {
        switch(i)
        {
            case 1: x++;
            case 2: x+=2; break;
            case 3: x+=1; break;
            default: x-=1;
        }
    }
    printf("x=%d",x);
    return 0;
}
```

Zadatak 3. Napisati program koji za uneti mesec ispisuje broj dana u mesecu.

```
#include<stdio.h>

int main(int argc, char *argv[])
{
    int mesec;
    char ch;
```

```

printf("Uneti redni broj meseca\n");
scanf("%d",&mesec);

switch(mesec)
{
case 1:case 3:case 5:case 7:case 8:case 10: case 12: printf("31 dan\n"); break;
case 4: case 6: case 9: case 11: printf("30 dana\n"); break;
case 2: printf("Da li je godina prestupna (d/n)?\n"); scanf("%c",&ch); scanf("%c",&ch);
        if((ch=='d')||(ch=='D'))
            printf("29 dana\n");
        else
            printf("28 dana\n"); break;
default: printf("Nekorektan broj meseca\n");
}
return 0;
}

```

Zadatak 4. Napisati program koji za uneti datum ispisuje datum sledećeg dana.

```

#include<stdio.h>

int main(int argc, char *argv[])
{
int d, m, g, p=0, brdana;
printf("Unesi datum(d m g): ");
scanf("%d%d%d",&d,&m,&g);

    if (((g%100 !=0 && g%4==0) || (g%400 == 0))
        p++;

    switch (m) {
case 1: case 3: case 5: case 7: case 8: case 10: case 12: brdana=31; break;
case 4: case 6: case 9: case 11: brdana=30; break;
case 2: brdana=28+p; break;
    }

    if ( d < brdana )
        d++;
    else if (m!=12) {
        d=1;
        m++;
    }
    else {
        d=1;
        m=1;
        g++;
    }
    printf("Datum sledeceg dana je: %d/%d/%d\n",d,m,g);
}

```

```
return 0;
}
```

Zadatak 5. Nalaženje korena kvadratne jednačine.

```
#include<stdio.h>
#include<math.h>

int main(int argc, char *argv[])
{
    double a,b,c,d,x1,x2,y1,y2;
    typedef enum {Realni,Kompleksni,Dvostruki,Linearna,Pogresna} Vrsta;
    Vrsta vrsta;

    printf("Unesi koeficijente kvadratne jednacine:");
    scanf("%lf%lf%lf",&a,&b,&c);

    if(a){
        d=b*b-4*a*c;
        if(d>0){
            vrsta=Realni;
            x1=(-b+sqrt(d))/(2*a);
            x2=(-b-sqrt(d))/(2*a);
        }
        else if (d==0){
            vrsta=Dvostruki;
            x1=(-b/(2*a));
        }
        else {
            vrsta=Kompleksni;
            x1=(-b/(2*a));
            x2=x1;
            y1=sqrt(-d)/(2*a);
            y2=-y1;
        }
    }
    else if(b){
        vrsta=Linearna;
        x1=-c/b;
    }
    else
        vrsta = Pogresna;

    switch(vrsta){
    case Realni: printf("Realni koreni su %lf i %lf\n",x1,x2);break;
    case Dvostruki: printf("Dvostruki realni koren je: %lf",x1);break;
    case Kompleksni: printf("Kompleksni koreni su:(%lf + %lf) i (%lf - %lf)",x1,y1,x2,y2);
    break;
    }
```

```

case Linearna: printf("Resenje linearne jednacine je: %lf",x1);break;
case Pogresna: printf("Podaci nemaju smisla!!!");break;
                }
printf("\n");
return 0;
}

```

Zadatak 6. Mnozenje broja sa 2 bez upotrebe operacija sabiranja i mnozenja.

```

#include<stdio.h>

int main(int argc, char *argv[])
{
    int d;
    printf("Unesi broj: "); scanf("%d",&d);
    d= d<<1;
    printf("Novi broj je: %d\n",d);
return 0;
}

```

Zadatak 7. Predsavljanje brojeva u binarnom brojnom sistemu.

```

#include<stdio.h>

int main(int argc, char *argv[])
{
    int broj,i;
    printf("Unesi decimalni broj: ");
    scanf("%d",&broj);
    for (i=1;i<=8;i++)
    {
        if ((broj & 0x80) != 0) {
            broj <<= 1;
            printf("1");
        }
        else {
            broj <<= 1;
            printf("0");
        }
        if (i%4 == 0)
            printf(" ");
    }
    printf("\n");
return 0;
}

```

Zadatak 8. Permutovanje cifara broja.

```
#include<stdio.h>

int main(int argc, char *argv[])
{
    int broj;
    printf("Ukucajte ceo broj?\n");
    scanf("%d",&broj);
    printf("Permutovan broj je\n");
    do
    {
        printf("%d",broj%10);
        broj/=10;
    }
    while(broj);
    printf("\n");
return 0;
}
```

Zadatak 9. Ispisivanje slučajnih brojeva pritiskom na taster SPACE.

```
#include <stdio.h>
#include <conio.h>
#include <ctype.h>
#include <time.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    int ch;
    ch = -1;

    srand( (unsigned)time(NULL) );

    while ( ch != 0x1B )
    {
        if ( kbhit() )
        {
            ch = getch();
            switch ( ch )
            {
                case 0x20: printf( "%d\n",rand()%100 ); break;
                default: printf( "Greska!\n" ); break;
            }
        }
    }
}
```

```

return 0;
}

```

Zadatak 10. Napisati program u kome se definiše funkcija `prost()` koja ispituje da li je broj `prost` i funkcija `main()` koja njenim korišćenjem štampa sve proste brojeve do datog prirodnog broja `n`.

```

#include<stdio.h>
#include<math.h>

int prost(int n)
{
int i;
for (i=2;i<=sqrt(n);i=i+1)
if(n%i==0)
return 0;
return i;
}

int main(int argc, char *argv[])
{
int i,n;
printf("Uneti broj\n");
scanf("%d",&n);
printf("Prosti brojevi su\n");
for(i=2;i<=n;i=i+1)
if(prost(i)) printf("%d\n",i);
return 0;
}

```

Zadatak 11. Napisati program kojim se štampaju svi trocifreni brojevi (ako ih ima) koji su jednaki sumi faktoriijela svojih cifara.

```

#include<stdio.h>

int fakt(int n)
{
int p=1,i;
for(i=2;i<=n;i++)
p*=i;
return p;
}

int main(int argc, char *argv[])
{
int n,a,b,c;

```

```

printf("Trocifreni brojevi jednaki sumi faktorijela svojih cifara su:\n");
for(n=100;n<1000;n++)
{
    a=n%10;
    b=n/10%10;
    c=n/100;
    if(n==fakt(a)+fakt(b)+fakt(c)) printf("%d\n",n);
}
return 0;
}

```

Zadatak 12. Pomeranje elemenata niza za jedno mesto u levo.

```

#include<stdio.h>

void upis(int a[], int n)
{
    int i;

    for(i=0;i<n;i++)
    {
        printf("a[%d]= ",i+1);
        scanf("%d",&a[i]);
    }
}

void ispisi(int a[], int n)
{
    int i;

    for(i=0;i<n;i++)

        printf(" a[%d]= %d",i+1,a[i]);
}

void pomerilevo(int a[], int n)
{
    int i,pom;
    pom = a[0];
    for(i=1;i<n;i++)
    a[i-1]=a[i];
    a[n-1]=pom;
}

int main(int argc, char *argv[])
{
    int a[25],n;
    n=5;
    upis(a,n);
}

```



```

pomerilevo(a,n);
ispis(a,n);
printf("\n");
return 0;
}

```

Zadatak 13. Izdvajanje niza iz k-te paralele dijagonale kvadratne matrice, i njegovo sortiranje.

```
#include<stdio.h>
```

```

void upis(int a[][25],int n)
{
int i,j;
for(i=0;i<n;i++)
for(j=0;j<n;j++)
{
printf("a[%d][%d]=",i+1,j+1);
scanf("%d",&a[i][j]);
}
}

```

```

void ispis(int a[],int n)
{
int i;

for(i=0;i<n;i++)
printf("%d\n",a[i]);
}

```

```

int izdvoj (int a[][25],int b[],int n,int k)
{
int i,j,p=0;
/*za k-tu paralelu iznad sporedne dijagonale */
for(i=0;i<=n-1-k;i++)
for(j=0;j<=n-1-k;j++)
if(i+j==n-1-k)
b[p++]=a[i][j];

/*za k-tu paralelu ispod sporedne dijagonale
for(i=k;i<=n-1;i++)
for(j=k;j<=n-1;j++)
if(i+j==n-1+k)
b[p++]=a[i][j]; */

/*za k-tu paralelu iznad glavne dijagonale
for(i=0;i<=n-1-k;i++)
for(j=k;j<=n-1;j++)

```

```

if(j-i == k)
b[p++] = a[i][j];*/

/*za k-tu paralelu ispod glavne dijagonale
for(i=k;i<=n-1;i++)
for(j=0;j<=n-1-k;j++)
if(i-j == k)
b[p++] = a[i][j];*/
return p;
}

```

```

void sort(int b[],int n)
{
int i,j,pom;
for(i=0;i<n-1;i++)
for(j=i+1;j<n;j++)
if(b[i]>b[j])
{
pom=b[i];
b[i]=b[j];
b[j]=pom;
}
}

```

```

int main(int argc, char *argv[])
{
int a[25][25],b[25],n,k,br;
n=4;
k=1;
upis(a,n);
br=izdvoj(a,b,n,k);
sort(b,br);
ispis(b,br);
return 0;
}

```

Zadatak 14. Napisati program koji od unete matrice formira niz na taj način što od maksimalnog elementa oduzme minimalan u svakoj vrsti matrice.

```
#include<stdio.h>
```

```

void unos(int a[][25], int v, int k)
{
int i,j;
for(i=0;i<v;i++)
for(j=0;j<k;j++)
{
printf("a[%d,%d]= ",i+1,j+1);
}
}

```

```

        scanf("%d",&a[i][j]);
    }
}

void ispis(int a[], int v)
{
    int i;
    for(i=0;i<v;i++)
        printf("%d ",a[i]);
}

void razlika(int a[][25], int v, int k, int b[])
{
    int i,j, max[25], min[25];
    for(i=0;i<v;i++)
    {
        max[i]=a[i][0];

        for(j=0;j<k;j++)
            if(a[i][j]>max[i])
                max[i]=a[i][j];
    }

    for(i=0;i<v;i++)
    {
        min[i]=a[i][0];
        for(j=0;j<k;j++)
            if(a[i][j]<min[i])
                min[i]=a[i][j];
    }

    for(i=0;i<v;i++)
        b[i]= max[i]-min[i];
}

int main(int argc, char *argv[])
{
    int a[25][25],k,v,b[25];
    printf("Unesi dimenziju matrice(npr: 3x4): ");
    scanf("%d%d",&v,&k);
    unos(a,v,k);
    razlika(a,v,k,b);
    ispis(b,v);
    return 0;
}

```

Zadatak 15. Napisati program koji menja elemente iz dve vrste matrice.

```
#include<stdio.h>

void unos(int a[][25], int v, int k)
{
    int i,j;
    for(i=0;i<v;i++)
    for(j=0;j<k;j++)
    {
        printf("a[%d,%d]= ",i+1,j+1);
        scanf("%d",&a[i][j]);
    }
}

void ispis(int a[][25], int v, int k)
{
    int i,j;
    for(i=0;i<v;i++)
    {
        for(j=0;j<k;j++)
        printf("%d ",a[i][j]);
        printf("\n");
    }
}

void zamena(int a[][25], int k, int m, int n)
{
    int i,pom;
    for(i=0;i<k;i++)
    {
        pom = a[m][i];
        a[m][i] = a[n][i];
        a[n][i] = pom;
    }
}

int main(int argc, char *argv[])
{
    int a[25][25],k,m,n,v;
    printf("Unesi dimenziju matrice(npr: 3x4): ");
    scanf("%d%d",&v,&k);
    unos(a,v,k);
    printf("Unesi dve vrste koje menjas: ");
    scanf("%d%d",&m,&n);
    zamena(a,k,m,n);
    ispis(a,v,k);
    return 0;
}
```

Zadatak 16. Napisati program koji za uneti niz brojeva određuje koliko ih ima do prvog negativnog.

```
#include<stdio.h>

void upis(int a[], int n)
{
    int i;
    for(i=0;i<n;i++)
        scanf("%d",&a[i]);
}

void ispis(int a[], int n)
{
    int i;
    for(i=0;i<n;i++)
        printf("%d",a[i]);
}

int brojpozitivnih(int a[], int n)
{
    int k=0,i;
    for(i=0;i<n && a[i]>=0;i++)
        if(a[i]>0)
            k++;
    return k;
}

int main(int argc, char *argv[])
{
    int a[100],n;
    printf("Unesite dimenziju niza: ");
    scanf("%d",&n);
    printf("Unesite članove niza: ");
    upis(a,n);
    printf("Broj pozitivnih do prvog negativnog je: %d", brojpozitivnih(a,n));
    return 0;
}
```

Zadatak 17. Napisati program koji sve negativne elemente niza stavlja na kraj tog niza.

```
#include <stdio.h>

void citaj(float a[],int n)
{
    int i;
    printf("Uneti niz \n");
    for(i=0;i<n;i++)
```

```
{
printf("a[%d]=",i);
scanf("%f",&a[i]);
}
}

void uredi(float a[],int n)
{
float pom;
int i,j;
i=0; j=n-1;
while(i<j)
if(a[i]>0) i++;
else
{
if(a[j]>0)
{
pom=a[i];
a[i]=a[j];
a[j]=pom; }
j--;
}
}

void pisi(float a[],int n)
{
int i;
printf("Formirani niz\n");
for(i=0;i<n;i++)
printf("a[%d]=%5.2f\n",i,a[i]);
}

int main(int argc, char *argv[])
{
float a[100];
int n;
printf("Uneti broj elemenata niza\n");
scanf("%d",&n);
citaj(a,n);
uredi(a,n);
pisi(a,n);
}
```

Zadatak 18. Napisati program koji za slučajno uneti niz brojeva, sažima niz na taj način što sve iste elemente koji su jedan za drugim sabira i postavlja na prvo mesto pojavljivanja tog broja u nizu.

```
#include<stdio.h>
#include<stdlib.h>
#include<time.h>

void upis (int a[], int n)
{
    int i;
    srand((unsigned)time(NULL));
    for(i=0;i<n;i++)
    {
        a[i]=rand()%10;
    }
}

void ispis (int a[], int n)
{
    int i;
    for(i=0;i<n;i++)
    {
        printf("%d ",a[i]);
    }
}

int sazimanje(int a[], int b[], int n)
{
    int i,j,k,p=0,pom;

    for(i=0;i<n;i++)
    {
        k=i;
        j=i+1;
        pom=a[k];

        while(a[k] == a[j])
        {
            pom=pom+a[j];
            j++;
            i++;
        }
        b[p++]=pom;
    }
    return p;
}

int main(int argc, char *argv[])
{
    int n,a[50],b[50],br;
    printf("Unesi dimenziju niza: ");
    scanf("%d",&n);
```

```
printf("\nUnesi elemente niza");
upis(a,n);
br = sazimanje(a,b,n);
printf("\n");
ispis(a,n);
printf("\n");
ispis(b,br);
printf("\n");
}
```

Zadatak 19. Napisati program koji na početak niza stavlja one elemente koji se pojavljuju samo jedanput u nizu, a zatim ostale elemente po redosledu po kome su učitani u niz.

```
#include<stdio.h>
#include<stdlib.h>
#include<time.h>

void upis (int a[], int n)
{
    int i;
    srand((unsigned)time(NULL));
    for(i=0;i<n;i++)
    {
        a[i]=rand()%10;
    }
}

void ispis (int a[], int n)
{
    int i;
    for(i=0;i<n;i++)
    {
        printf("%d ",a[i]);
    }
}

int brponavljanja(int a[], int n, int w)
{
    int i,k=0;
    for(i=0;i<n;i++)
    {
        if (a[i] == w)
            k++;
    }

    return k;
}
```



```

int main(int argc, char *argv[])
{
int n,a[50],b[50],c[50],k=0,e=0,i;

printf("Unesi dimenziju niza: ");
scanf("%d",&n);
printf("\nUnesi elemente niza");
upis(a,n);

for(i=0;i<n;i++)
{
if (brponavljanja(a,n,a[i]) == 1)
b[k++]=a[i];
else
c[e++]=a[i];
}

printf("\n");
ispis(a,n);
printf("\n");
ispis(b,k);
printf(" ");
ispis(c,e);
}

```

Zadatak 20. Šta nedostaje u sledećem opisu, da bi predstavljao opis stringa?

```
char ime[]={F, A, L, S, E};
```

Rešenje: Nedostaje null znak na kraju stringa.

Zadatak 21. Šta ispisuje sledći program?

```

#include<stdio.h>
int main(int argc, char *argv[])
{
char film[]="Tango argentino";
char *pok;
pok=film;
puts(pok);
puts(++pok);
film[5]='\0';
puts(film);
puts(++pok);
}

```

Rešenje:
Tango argentino
ango argentino

Tango
ngo

Zadatak 22. Šta ispisuje sledeći program?

```
#include<stdio.h>
#include<string.h>
int main(int argc, char *argv[])
{
char ime[]="Bingo";
char *pok;
pok=ime+strlen(ime);
while(--pok >= ime)
    puts(pok);
}
```

Rešenje:

o
go
ngo
ingo
Bingo

Zadatak 23. Šta ispisuje sledeći program?

```
#include<stdio.h>
#include<string.h>
int main(int argc, char *argv[])
{
char s1[28]="ani ceo";
char s2[40]="Ko rano r";
char *s3="dan zeva.";
strcat(s1,s3);
strcat(s2,s1);
puts(s2);
}
```

Rešenje: Ko rano rani ceo dan zeva.

Zadatak 24. Napisati funkciju koja:

- kopira string s1 u s2;
- proverava da li je string s2 palindrom;
- vraća pokazivač na prvo pojavljivanje znaka c u stringu s
- brise svako pojavljivanje znaka c;
- vraća poziciju prvog pojavljivanja stringa s2 u stringu s1, ako string s2 nije prisutan u stringu s1 funkcija treba da vrati -1.

```

a)
void kopi(char *s1, char *s2)
{
while (*s2==*s1)
{
s1++;
s2++;
}
}

b)
int palindrom(char *s)
{
char *p=s+strlen(s)-1;
while(s<p && *s==*p)
    {s++; p--;}
return (s>=p);
}

c)
char *nadj_i_prvi(char *s, char c)
{
while(*s && *s!=c) s++;
if(*s) return s
}

d)
void brisi_znak(char *s, char c)
{
char *p=s;
while(*p)
{
if(*p!=c) *s++=*p;
p++;
}
*s='\0';
}

e)
int pos(char *s1, char *s2)
{
if(strstr(s1,s2)!=NULL) return strstr(s1,s2) - s1;
else return -1;
}

```

Zadatak 25. Napisati funkciju kojom se iz stringa s briše d znakova počev od pozicije p, pri čemu počev od pozicije p u stringu s ima barem d znakova.

```

#include< stdio.h >
#include<string.h>
void brisi(char *s, int d, int p)
{
char *q;
for(q=s+p;*(q+d);q++)    *q=*(q+d);
*q='\0';
}

```

Zadatak 26. Napisati program koji za uneti string i neki broj ispisuje samo one karaktere sa pozicija gde se u broju u njegovom binarnom zapisu javlja broj 1.

```

#include <stdio.h>

int main(int argc, char *argv[])
{
short int M, maska;
char S1[17], *s1, S2[17], *s2;
printf("%Unesi string S1 i broj M:\n");
scanf("%s%d", &S1, &M);
s1 = S1;
s2 = S2;
maska = 1;
while (maska)
{
if (M & maska) {*s2 = *s1; s2++; };
s1++; maska<<=1;
}
*s2 = '\0';
printf("Novi string S2: %s\n", S2);
}

```

Zadatak 27. Objašnjenje nizovia i pokazivača.

```

/*Nizovi i pokazivaci*/
#include<stdio.h>
int main(int argc, char *argv[])
{
float a,b,c,fniz[5]={0.01,0.1,0.5,1.0,10.0};
float *p_fniz;
char tekst[]={"Ovo je znakovni niz\n"};
char *p_tekst;
int i;

```

```

p_fniz=fniz;
a=*p_fniz; // a=fniz[0]
b=(p_fniz+2); // b=fniz[2]
p_fniz=&fniz[2];
c=(p_fniz+2); // c=fniz[4]
printf("a=%f b=%f c=%f\n",a,b,c);

//ispis stinga
for(i=0;tekst[i]!='\0';++i)putchar(tekst[i]);
// ispis stringa na drugi nacin
for(p_tekst=tekst;*p_tekst!='\0';++p_tekst)putchar(*p_tekst);

return 0;
}

```

Zadatak 28. Stranice a i b pravougaonika su prirodni brojevi. Odrediti, na koliko se kvadrata maksimalne površine može iseći dati pravougaonik. Ispisati dimenzije kvadrata. Na primer za $a=12$ i $b=7$ se ispisuje:

```

1 kvadrata stranice: 7
1 kvadrata stranice: 5
2 kvadrata stranice: 2
2 kvadrata stranice: 1

```

```
#include<stdio.h>
```

```

void razmeni(int *a, int *b)
{
int t;
t=*a;
*a=*b;
*b=t;
}

```

```

int main(int argc, char *argv[])
{
int a,b;
printf("Uneti stranice pravougaonika\n");
scanf("%d%d",&a,&b);
if(a==b)
printf("Jedan kvadrat stranice %d\n",a);
else
while(a)
{
if(a<b) razmeni(&a,&b);
printf("%d kvadrata stranice: %d\n",a/b,b);
a=a%b;
}
return 0;
}

```

Zadatak 29. Napisati program kojim se iz stringa s briše svaki znak koji odgovara bilo kom znaku stringa t.

```
#include <stdio.h>
#include <string.h>

int main(int argc, char *argv[])
{
    char s[50], t[50];
    int i,j;
    gets(s);
    gets(t);
    for(i=j=0;*(s+i);i++)
    if(!strchr(t,*(s+i)))
    {
        *(s+j)=*(s+i);
        j++;
    }
    *(s+j)='\0';
    puts(s);
}
```

Zadatak 30. Program za određivanje dužine stringa koji se učitava sa komandne linije.

```
#include<stdio.h>

int duzina(char *p_string);

int main(int argc,char *argv[])
{
    int i;
    for(i=1;i<argc;i++)
    printf("argument%d %s\n",i,argv[i]);
    i=duzina(argv[2]);
    printf("duzina stringa je %d\n",i);
    return 0;
}

// funkcija koja kao rezultat vraca duzinu stringa
int duzina(char *p_string)
{
    int l=0;
    char *p;
    p=p_string;
    while(*p!='\0')
    {
```

```
++l;  
++p;  
}  
return l;  
}
```

Zadatak 31. Napisati program koji iz datog stringa s izbacuje sve komentare, s tim što u komentaru nema komentara.

```
#include <stdio.h>  
#include <string.h>  
  
int main(int argc, char *argv[])  
{  
    char s[100], *p,*q;  
    gets(s);  
    while(p=strstr(s,"/*"))  
    {  
        q=strstr(s,"/*");  
        strcpy(p,q+2);  
    }  
    puts(s);  
}
```

Zadatak 32. Napisati program koji određuje broj reči u stringu s i dužinu najduže reči, reci su razdvojene blanko simbolima.

```
#include<stdio.h>  
  
int main(int argc, char *argv[])  
{  
    char s[80];  
    int i,brojreci,brojslovaureci,maksimalnarec;  
    gets(s);  
    maksimalnarec=brojslovaureci=brojreci=0;  
  
    for(i=0;*(s+i);i++)  
        if(*(s+i)==' ')  
        {  
            if(brojslovaureci>maksimalnarec) maksimalnarec=brojslovaureci;  
            brojslovaureci=0;  
        }  
}
```

```

else
{
if(!brojslovaureci)brojreci++;
brojslovaureci++;
}

if(brojslovaureci>maksimalnarec) maksimalnarec=brojslovaureci;
printf("%d %d\n",brojreci,maksimalnarec);
return 0;
}

```

Zadatak 33. Napisati program koji određuje prosečan broj slova u rečima parne dužine.

```

#include<stdio.h>

int main(int argc, char *argv[])
{
char s[80];
// br broj reci parne duzine,
// q broj simbola u recima parne duzine
int k,br=0,q=0,i;
gets(s);
k=0;
for(i=0;*(s+i);i++)
if(*(s+i)!=' ')
k++;
else
if(k>0)
{
if(k%2==0)
{
br++;
q+=k;
}
k=0;
}

if(k>0)
if(k%2==0)
{
br++;
q+=k;
}

printf("Prosecan broj simbola u recima parne duzine je %.0f\n",(float)q/br);
return 0;
}

```


Zadatak 34. Nalaženje unije dva skupa.

```

#include<stdio.h>
#include<stdlib.h>

int main(int argc, char *argv[])
{

int n1,n2,n3,i,j;
double *s1, *s2, *s3;

printf("Unesi broj elemenata skupova: ");
scanf("%d%d",&n1,&n2);

s1=malloc(n1*sizeof(double));
printf("Elementi prvog skupa su: ");
for(i=0;i<n1;scanf("%lf",&s1[i++]));
if (n1==0) putchar('\n');

s2=malloc(n2*sizeof(double));
printf("Elementi drugog skupa: ");
for(i=0;i<n2;scanf("%lf",&s2[i++]));
if (n2==0) putchar('\n');

s3=malloc((n1+n2)*sizeof(double));
for(n3=0;n3<n1;n3++)
s3[n3]=s1[n3];
for(j=0;j<n2;j++){
for(i=0;i<n1;i++)
if(s2[j]==s1[i]) break;
if(i==n1) s3[n3++]=s2[j];
}

s3=realloc(s3,n3*sizeof(double));

printf("Unija pocetnih skupova: ");
for(i=0;i<n3;i++)
printf("%lf ",s3[i]);

free(s1);
free(s2);
free(s3);
}

```

Zadatak 35. Sortiranje imena smeštenih u dinamičku matricu.

```

#include<stdio.h>

```

```

#include<string.h>
#include<stdlib.h>

#define MAX_IME 100
#define MAX_DUZ 30

int main(int argc, char *argv[])
{

char **imena, *ime;
int br_imena=0,duz,i;

printf("\nNeuredjeni niz imena: \n");
imena = malloc(MAX_IME*sizeof(char *));

do {

ime=malloc(MAX_DUZ*sizeof(char));
gets(ime);

if((duz=strlen(ime))==0)
{
free(ime);
break;
}

for(i=br_imena-1;i>=0;i--)
if( strcmp(imena[i],ime) > 0 )
imena[i+1]=imena[i];
else break;
imena[i+1]=ime;

}
while(++br_imena<MAX_IME);

imena = realloc(imena, br_imena*sizeof(char *));
printf("Uredjeni niz imena:\n");
for(i=0;i<br_imena;i++)
puts(imena[i]);
free(imena);
}

```

Zadatak 36. Odrediti broj linija u tekstualnom fajlu sa imenom knjiga.txt.

```

#include<stdio.h>

int main(int argc, char *argv[])
{
int k=0;

```

```

FILE *fp;
char s[256];
if((fp=fopen("knjiga.txt","r"))==NULL)
{
perror("Greska");
return;
}
while(fgets(s,256,fp)!=NULL)k++;
printf("Fajl ima %d linija\n",k);
fclose(fp);
}

```

Zadatak 37. Napisati program kojim se sadržaj fajla test.dat formiran od velikih slova alfabeta šifriran šalje u fajl sifra.dat. Znak se šifrira tako što se zamenjuje sledećim ASCII znakom, a znak Z zamenjuje sa A.

```

#include<stdio.h>

int main(int argc, char *argv[])
{
int c;
FILE *inf,*outf;
inf=fopen("test.dat","r");
outf=fopen("sifra.dat","w");
while((c=fgetc(inf))!=EOF)
{
if('A'<=c&&c<'Z')
c++;
else c='A';
putc(c,outf);
}
fclose(inf);
fclose(outf);
}

```

Zadatak 38. U tekstualnom fajlu odrediti najdužu liniju. Ako ima više linija najveće dužine odrediti prvu među njima.

```

#include<stdio.h>
#include<string.h>

int main(int argc, char *argv[])
{
unsigned k=0,maxl;
42
char s[256],maxs[256];
FILE *fp;

```

```

if((fp=fopen("test.txt","r"))==NULL)
{
printf("Greska\n ");
return;
}
maxl=0;
while(fgets(s,256,fp)!=NULL)
if(strlen(s)>maxl)
{
maxl=strlen(s);
strcpy(maxs,s);
}
printf("Najduza linija\n%s\n ima duzinu %d\n",maxs,maxl);
fclose(fp);
}

```

Zadatak 39. Napisati program za učitavanje datoteke pod imenom ulaz.dat i formiranje datoteke sa nazivom izlaz.dat koja sadrži samo one redove ulazne datoteke ulaz.dat u kojima se nalazi reč “fakultet”.

```

#include<stdio.h>
#include<string.h>

int main(int argc, char *argv[])
{
FILE *ulaz;
FILE *izlaz;
char buffer[80];
char trazenaRec[]="fakultet";
if((ulaz=fopen("ulaz.dat","r"))==NULL)
printf("Datoteka ulaz.dat nije otvorena.");
if((izlaz=fopen("izlaz.dat","w"))==NULL)
printf("Datoteka izlaz.dat nije otvorena.");
while(!feof(ulaz)){
fgets(buffer,80,ulaz);
if(strstr(buffer,trazenaRec)!=NULL){
printf("%s",buffer); /*stamanje na ekranu*/
fprintf(izlaz,"%s",buffer);
}
}
fclose(ulaz);
fclose(izlaz);
}

```

Zadatak 40. Napisati program koji čita podatke iz datoteke ULAZ.TXT (ime studenta, broj bodova na prijemnom ispitu), broji sve studente koji imaju broj bodova veći od proseka i upisuje podatke o takvim studentima u datoteku IZLAZ.TXT.

```
#include<stdio.h>

#define DIM 10
typedef struct student{
char ime[20];
int brBodova;
};

int main(int argc, char *argv[])
{
struct student nizStudenata[DIM];
int i;
int brojStudenata=0;
int suma=0;
float prosek;
FILE *ulaz;
FILE *izlaz;
ulaz=fopen("ulaz.txt", "r");
if(ulaz==NULL){
printf("Ulazna datoteka nije otvorena!\n");
return 1;
}
izlaz=fopen("izlaz.txt", "w");
if(izlaz==NULL){
printf("Izlazna datoteka nije kreirana!\n");
return 1;
}
while(!feof(ulaz)){
fscanf(ulaz, "%s%d", &nizStudenata[brojStudenata].ime,
&nizStudenata[brojStudenata].brBodova);
suma+=nizStudenata[brojStudenata].brBodova;
brojStudenata++;
}
prosek=(float)suma/brojStudenata;
for(i=0; i<brojStudenata; i++)
if(nizStudenata[i].brBodova>prosek)
fprintf(izlaz, "%s\t%d\n", nizStudenata[i].ime, nizStudenata[i].brBodova);
fclose(ulaz);
fclose(izlaz);
return 0;
}
```

Zadatak 41. U datoteci IMENIK.TXT nalaze se sledeci podaci: ime, prezime i broj telefona telefonskog pretplatnika. Napisati program koji vrsi sortiranje imenika prema prezimenu pretplatnika i za zadato prezime prikazuje telefonski broj pretplatnika na standardnom izlazu.

```
#include<stdio.h>
#include<string.h>
#define DIM 10

typedef struct telefonski_imenik{
char ime[10];
char prezime[20];
char broj[10];
};
void razmeni(struct telefonski_imenik *a, struct telefonski_imenik *b){
struct telefonski_imenik tmp;
tmp=*a;
*a=*b;
*b=tmp;
}

void sort(struct telefonski_imenik niz[], int n){
int i, j;
for(i=0; i<n-1; i++)
for(j=i+1; j<n; j++)
if(niz[i].prezime<niz[j].prezime)
razmeni(&niz[i], &niz[j]);
}

int main(int argc, char *argv[])
{
struct telefonski_imenik nizImenik[DIM];
int n, i;
FILE *in;
char trazi[20];
in=fopen("imenik.txt", "r");
if(in==NULL){
printf("Ulazni fajl nije otvoren!");
return 1;
}
i=0;
while(!feof(in)){
fscanf(in, "%s%s%s", &nizImenik[i].ime, &nizImenik[i].prezime, &nizImenik[i].broj);
i++;
}
fflush(in);
n=i+1;
sort(nizImenik, n);
printf("\nTelefonski imenik sortiran po prezimenu pretplatnika je:\n\n");
for(i=0; i<n; i++)
```

```
printf("%s\t%s\t%s\n", nizImenik[i].prezime, nizImenik[i].ime, nizImenik[i].broj);
printf("Ukupan broj telefonskih pretplatnika je: %d\n",n);
printf("Zadati prezime pretplatnika: ");
scanf("%s", &trazi);
for(i=0; i<n; i++){
if(strcmp(nizImenik[i].prezime,trazi)==0){
printf("Broj telefona trazenog pretplatnika je:\n");
printf("%s\t%s\t%s\n", nizImenik[i].prezime, nizImenik[i].ime, nizImenik[i].broj);
}
}
fclose(in);
return 0;
}
```

REŠENI ZADACI SA PRVOG KOLOKVIJUMA IZ PROGRAMSKIH JEZIKA

Zadatak 1. Napisati program koji za uneti broj ispisuje koliko ima jedinica u njegovom binarnom zapisu.

```
#include<stdio.h>

int brjedinica(int broj)
{
    int i,br=0;
    for(i=0;i<8;i++)
    {
        if(broj & 128)
            br++;
        broj<<=1;
    }
    return br;
}

int main(int argc, char *argv[])
{
    int broj;
    printf("Unesi broj: ");
    scanf("%d",&broj);
    printf("Broj jedinica je %d\n",brjedinica(broj));
    return 0;
}
```

Zadatak 2. Napisati program koji uneti niz pomera za n cifara u desno. Zadatak rešiti preko funkcija.

```
#include<stdio.h>

void unosniza(int a[],int n)
{
    int i;
    for(i=0;i<n;i++)
    {
        printf("a[%d]=",i+1);
        scanf("%d",&a[i]);
    }
}

void ispishiza(int a[],int n)
{
    int i;
    for(i=0;i<n;i++)
        printf("%d ",a[i]);
    printf("\n");
}

void pomeri(int a[], int n, int koliko)
```



```

{
int i,pom,j;
for(j=0;j<=koliko;j++)
{
pom=a[n-1];
for(i=n-1;i>=0;i--)
a[i]=a[i-1];
a[0]=pom;
}
}
int main(int argc, char *argv[])
{
int a[25],n,k;
printf("Unesi broj elemenata niza: ");
scanf("%d",&n);
unosniza(a,n);
printf("Unesi broj za koji pomeras niz u desno: ");
scanf("%d",&k);
printf("Niz posle pomeranja u desno je:\n");
pomeri(a,n,k);
ispisniza(a,n);
return 0;
}

```

Zadatak 3. Napisati program koji za unetu matricu vrši razmenu elemenata po vrstama na taj način što negativne elemente stavlja na kraj svake vrste.

```

#include<stdio.h>
void unosmatrice(int a[][20],int v,int k)
{
int i,j;
for(i=0;i<v;i++)
for(j=0;j<k;j++)
{
printf("a[%d][%d]=",i+1,j+1);
scanf("%d",&a[i][j]);
}
}
void ispismatrice(int a[][20],int v,int k)
{
int i,j;
for(i=0;i<v;i++)
{
for(j=0;j<k;j++)
printf("%d ",a[i][j]);
printf("\n");
}
}

```

```

}
void uredimaticu(int a[][20], int v, int k)
{
int i,j,pom,vrsta;
for (vrsta=0;vrsta<v;vrsta++)
{
i=0;
j=k-1;
while(i<j)
{
if(a[vrsta][i]>0)
i++;
else
if(a[vrsta][j]>0)
{
pom=a[vrsta][i];
a[vrsta][i]=a[vrsta][j];
a[vrsta][j]=pom;
j--;
}
}
}
}
int main(int argc, char *argv[])
{
int a[20][20],v,k;
printf("Unesi broj vrsta i kolona matrice: ");
scanf("%d %d",&v,&k);
unosmatrice(a,v,k);
uredimaticu(a,v,k);
ispismatrice(a,v,k);
return 0;
}

```

Zadatak 4. Napisati program koji za uneti broj ispisuje koliko ima nula u njegovom binarnom zapisu.

```

#include<stdio.h>
int brnula(int broj)
{
int i,br=0;
for(i=0;i<8;i++)
{
if((broj & 128)==0)
br++;
broj<<=1;
}
return br;
}

```

```

int main(int argc, char *argv[])
{
int broj;
printf("Unesi broj: ");
scanf("%d",&broj);
printf("Broj nula je %d\n",brnula(broj));
return 0;
}

```

Zadatak 5. Napisati program koji uneti niz pomera za 2. cifre u desno, pa za 3. cifre u levo. Zadatak raditi prko funkcija.

```

#include<stdio.h>
void unosniza(int a[],int n)
{
int i;
for(i=0;i<n;i++)
{
printf("a[%d]=",i+1);
scanf("%d",&a[i]);
}
}
void ispisniza(int a[],int n)
{
int i;
for(i=0;i<n;i++)
printf("%d ",a[i]);
printf("\n");
}
void pomeri(int a[], int n, int smer, int koliko)
{
int i,pom,j;
switch(smer)
{
/*pomeranje u levo */
case 1: {
for(j=0;j<koliko;j++)
{
pom=a[0];
for(i=0;i<n-1;i++)
a[i]=a[i+1];
a[n-1]=pom;
}
} break;
/*pomeranje u desno */
case 2: {
for(j=0;j<koliko;j++)
{
pom=a[n-1];

```

```

for(i=n-1;i>=0;i--)
a[i]=a[i-1];
a[0]=pom;
}
} break;
}
}
int main(int argc, char *argv[])
{
int a[25],n;
printf("Unesi broj elemenata niza: ");
scanf("%d",&n);
unosniza(a,n);
pomeri(a,n,2,2);
printf("Niz posle pomeranja u desno za dve pozicije:\n");
ispisniza(a,n);
pomeri(a,n,1,3);
printf("Niz posle pomeranja u levo za tri pozicije:\n");
ispisniza(a,n);
return 0;
}

```

Zadatak 6. Napisati program koji za unetu matricu vrši razmenu elemenata po kolonama na taj način što negativne elemente stavlja na kraj svake kolone.

```

#include<stdio.h>
void unosmatrice(int a[][20],int v,int k)
{
int i,j;
for(i=0;i<v;i++)
for(j=0;j<k;j++)
{
printf("a[%d][%d]=",i+1,j+1);
scanf("%d",&a[i][j]);
}
}
void ispismatrice(int a[][20],int v,int k)
{
int i,j;
for(i=0;i<v;i++)
{
for(j=0;j<k;j++)
printf("%d ",a[i][j]);
printf("\n");
}
}
void uredimaticu(int a[][20], int v, int k)
{

```

```

int i,j,pom,kolona;
for (kolona=0;kolona<k;kolona++)
{
i=0;
j=v-1;
while(i<j)
{
if(a[i][kolona]>0)
i++;
else
if(a[j][kolona]>0)
{
pom=a[i][kolona];
a[i][kolona]=a[j][kolona];
a[j][kolona]=pom;
j--;
}
}
}
}
}

```

```

int main(int argc, char *argv[])
{
int a[20][20],v,k;
printf("Unesi broj vrsta i kolona matrice: ");
scanf("%d %d",&v,&k);
unosmatrice(a,v,k);
uredimatricu(a,v,k);
ispismatrice(a,v,k);
return 0;
}

```

Zadatak 7. Марку је једне преступне године на часу из програмских језика било досадно. Насумице је окренуо роковник који је показивао датум(d.m.g.). Не знајући шта да ради, Марко је почео редом да исписује датуме почев од показаног датума, без водећих нула, са тачком иза дана, месеца и године. После исписаних **n** знакова, оловка је престала да пише. Написати програм који, за уčitани датум исписује код ког датума је Марку престала оловка да пише.

```

#include<stdio.h>

int broj_cifara(int n)
{
    int s=0;

    while(n)

```

```
{
    s++;
    n=n/10;
}
return s;
}
int main(int argc, char *argv[])
{
    int dan,mesec,godina,n,s,brdana;
    printf("Unesite dan, mesec i godinu: ");
    scanf("%d%d%d",&dan,&mesec,&godina);
    printf("Unesite broj cifara n: ");
    scanf("%d",&n);
    s=broj_cifara(dan)+broj_cifara(mesec)+broj_cifara(godina)+3;
    do
    {
        switch(mesec)
        {
            case 1: case 3: case 5: case 7: case 8: case 10: case 12: brdana=31;break;
            case 4: case 6: case 9: case 11: brdana=30; break;
            case 2: brdana=29; break;
        }
        if ( dan < brdana )
            dan++;
    }
    else if (mesec!=12) {
        dan=1;
        mesec++;
    }
}
```

```
    }  
    else {  
        dan=1;  
        mesec=1;  
        godina++;  
    }  
    s = s+ broj_cifara(dan) + broj_cifara(mesec) + broj_cifara(godina) + 3;  
    }  
    while( s < n );  
    printf("Olovka je prestala da pise kod datuma: %d %d %d\n",dan,mesec,godina);  
return 0;  
}
```

REŠEI ZADACI SA DRUGOG KOLOKVIJUMA IZ PROGRAMSKIH JEZIKA

Zadatak 1. На пријему је n особа, нумерисаних бројевима $0, 1, 2, \dots, n-1$. Свака особа има највише два непријатеља. Дана је матрица $A_{n \times n}$ тако да $a[i][j]=1$ ако је особи i непријатељ особа j , иначе $a[i][j]=0$. Матрица A је симетрична у односу на главну дијагоналу ($a[i][j]=a[j][i]$). Написати функцију којом се особе распоређују у три групе тако да су у свакој групи само пријатељи.

```
#include <stdio.h>
#include <stdlib.h>

int provera(int m[][6],int a[],int x[],int k,int i, int n)
{
    int brneprijatelja=0,s,t;
    for(s=0;s<k;s++){
        for(t=0;t<n;t++)
            if(a[s]==x[t]
                if(m[t][i]==1)
                    brneprijatelja++;
            }
    }
    return brneprijatelja;
}

int main(int argc, char*argv[])
{
    int x[6]={2,8,4,7,5,10};
    int m[6][6]= { 0,1,1,0,0,1,
                  1,0,1,0,1,0,
                  1,1,0,1,0,0,
                  0,0,1,0,0,0,
                  0,1,0,0,0,1,
                  1,0,0,0,1,0};
    int a[6],b[6],c[6];
    int i,j=0,k=0,l=0,n=6;
```



```

for (i=0;i<n;i++) {
    if (provera(m,a,x,j,i,n)==0)
        a[j++]=x[i];
    else
        if (provera(m,b,x,k,i,n)==0)
            b[k++]=x[i];
        else
            c[l++]=x[i];
}

printf("Prvi niz je:\n");
for(i=0;i<j;i++)
printf("%d ",a[i]);
printf("\nDrugi niz je:\n");
for(i=0;i<k;i++)
    printf("%d ",b[i]);
printf("\nTreci niz je:\n");
for(i=0;i<l;i++)
    printf("%d ",c[i]);
printf("\n");
system("PAUSE");
}

```

Zadatak 2. У датотеци **magacin.txt** се налази списак са подацима о **n** роба (шифра, назив, количина и јединична цена). У датотеци може да постоји више записа са истом шифром робе, који се налазе један за другим. Формирати датотеку **sredjeno.txt** у којој су сви записи са истом шифром сажети у један запис у коме је количина робе једнака збиру количина у свим записима са истом шифром.

```

#include<stdio.h>
#include<string.h>

struct Roba{

```

```
int sifra;
char naziv[20];
int kolicina;
float cena;
};

void sredi(struct Roba roba[], int n)
{
int i,j,k,p=0,pom;
FILE *out;
out=fopen("sredjeno.txt","w");

for(i=0;i<n;i++)
{
k=i;
j=i+1;

pom=roba[k].kolicina;

while(roba[k].sifra == roba[j].sifra) {
pom=pom+roba[j].kolicina;
j++;
i++;
}

fprintf(out,"%d %s %d %.2fd\n",roba[k].sifra,roba[k].naziv,pom,roba[k].cena);
}
}

int main(int argc, char *argv[]){

int i,n,p; FILE *in;
struct Roba roba[20];
printf("Unesi broj zapisa koje citas: "); scanf("%d",&n);
```

```

in=fopen("magacin.txt", "r");

for(i=0;i<n;i++){

fscanf(in,"%d%s%d%f",&roba[i].sifra,&roba[i].naziv,&roba[i].kolicina,&roba[i].cena
);

}

sredi(roba,n);
}

```

Zadatak 3. Написати програм на програмском језику **C** који учитава један занковни низ (стринг) **S1** и један цео број **M**, а затим формира нови знаковни низ **S2** само од оних знакова на чијим су позицијама одговарајући битови броја **M** једнаки нули. Сматрати да се цео број представља у 8-битној локацији, а да знаковни низ може имати највише 8 знакова. На крају, програм исписује нови знаковни низ **S2**.

```
#include <stdio.h>
```

```

int main(int argc,char *argv[])
{
    short int M, maska;
    char S1[9], *s1, S2[9], *s2;
    printf("%Unesi string S1 i broj M:\n");
    scanf("%s%d", &S1, &M);
    s1 = S1;
    s2 = S2;
    maska = 1;
    while (maska)
    {
        if (M & maska == 0) {*s2 = *s1; s2++; };
        s1++; maska<<=1;
    }
    *s2 = '\0';
}

```

```
printf("Novi string S2: %s\n", S2);
}
```

Zadatak 4. Izračunati i ispisati vrednost z po formuli: $z = \sqrt{\max^2(x, 16.8) - 2} * \frac{\min^3(x, 19.2)}{1+x^3}$, ako se x menja od x_1 do x_2 sa korakom 1. Ako nije ispunjen uslov definisanosti, uzeti da je vrednost promenljive z jednaka 1. Koristiti funkcije kod kojih se vrši prenošenje parametara po referenci.

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#define min 19.2
#define max 16.8

void min3(float x, float *k)
{
    if (x<min) *k=pow(x,3);
    else *k=pow(min,3);
}

void max2(float x, float *k)
{
    if (x<max) *k=pow(max,2);
    else *k=pow(x,2);
}

void koren(float x, float *k)
{
    float p;
    max2(x,&p);
    *k=sqrt(p-2);
}

void razl(float x, float *k)
{
    float p;
    min3(x,&p);
    *k=p/(1+pow(x,3));
}

void ispis(float a[],int n)
{
    int i;
    for(i=0;i<n;i++)
        printf("%f\n",a[i]);
}

int main(int argc, char *argv[])
{
```

```

float x1,x2,h,d1,d2,z[30];
int j=0;
printf("Unesi granice x1, x2 i korak h: ");
scanf("%f%f%f",&x1,&x2,&h);
while( x1 <= x2 )
{
    if (x1==-1)
        z[j]=1;
    else
    {
        koren(x1,&d1);
        razl(x1,&d2);
        z[j]=d1*d2;
    }
    j++;
    x1=x1+h;
}
ispis(z,j);
system("PAUSE");
}

```

Zadatak 5. Написати програм на програмском језику **C** који учитава један занковни низ (string) **S1** и један цео број **m**, а затим формира нови знаковни низ **S2** тако што од низа **S1** узме све елементе од **p**-тог до **q**-тог, уклони све карактере “**a**” уколико постоје и нађе суму квадрата од броја **k**(број елемената низа **S2**) до **m**. Уколико је та сума већа од $41 * m$ решење исписати на стандардном излазу, у супротном решење запамтити у датотеци. Задатак решити коришћењем функција које своје параметре преносе преко референци.

```

#include<stdio.h>
#include<string.h>

void seci(char *s,int p,int q)
{
    char *t;
    t=s;
    while( p<q )
    {
        *s=*(t+p);
        s++;
        p++;
    }
    *s='\0';
}

```

```
void sredi(char *s)
{
    char *t;
    t=s;
    while(*s && *s!='a')
    {
        *s=*t;
        t++;
        s++;
    }
    *s='\0';
}

void kv(char *s, int m,float *k,int dim)
{
    int i;
    for(i=0;i<dim;i++)
        *k=*k + *(s+i);
}

int main(int argc,char *argv[])
{
    char s1[50];
    int m,p,q,dim;
    float kvadrat=0;
    FILE *out;
    out=fopen("sredjeno.txt","w");

    printf("Unesite string:");
    gets(s1);
    printf("unesite p,q i m:");
    scanf("%d%d%d",&p,&q,&m);
    seci(s1,p,q);
    sredi(s1);
    dim=strlen(s1);
    kv(s1,m,&kvadrat,dim);
    if (kvadrat>41*m)
        fprintf(out,"%f je vece od %d",kvadrat,41*m);
    else
        fprintf(out,"%f nije vece od %d",kvadrat,41*m);
}
```