Popa Ecaterina

gr. TI-211

Probleme rezolvate – 251

**Probleme propuse pentru examen**

**Tema: Tipuri de date simple și operații de bază asupra lor (p. 14-16)   
– 38 probleme**

1. Să se calculeze valoarea expresiei ax^2 + bx + c:

#include <stdio.h>

int main () {

int x, a, b, c;

printf ("Introduceti numarul a:");

scanf ("%d", &a);

printf ("Introduceti numarul b:");

scanf ("%d", &b);

printf ("Introduceti numarul c:");

scanf ("%d", &c);

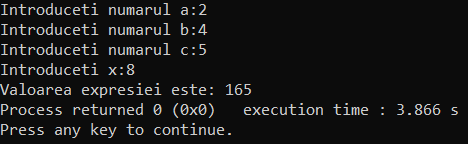
printf ("Introduceti x:");

scanf ("%d", &x);

printf ("Valoarea expresiei este: %d", ((a\*x\*x)+(b\*x)+c));

return 0;

}



1. Se dau două numere întregi. Să se determine suma și diferența lor.

#include <stdio.h>

int main () {

int a, b;

printf ("Introduceti a: ");

scanf ("%d", &a);

printf ("Introduceti b: ");

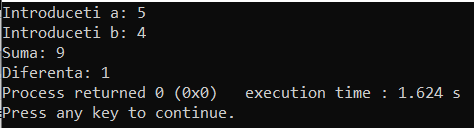
scanf ("%d", &b);

printf("Suma: %d\n", a+b);

printf("Diferenta: %d", a-b);

return 0;

}



1. Se dau două numere întregi. Să se determine produsul și câtul lor.

#include <stdio.h>

int main () {

int a, b;

printf ("Introduceti a: ");

scanf ("%d", &a);

printf ("Introduceti b: ");

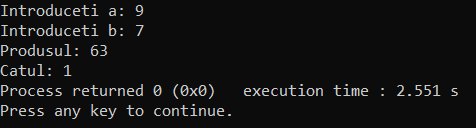
scanf ("%d", &b);

printf("Produsul: %d\n", a\*b);

printf("Catul: %d", a/b);

return 0;

}



1. Se dau două numere naturale. Să se determine media aritmetică și media geometrică a acestora.

#include <stdio.h>

#include <math.h>

int main () {

int a, b;

printf ("Introduceti a: ");

scanf ("%d", &a);

printf ("Introduceti b: ");

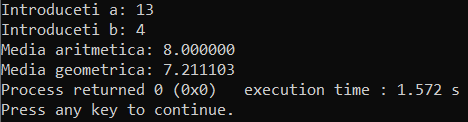
scanf ("%d", &b);

printf("Media aritmetica: %lf\n", (double)((a + b)/2));

printf("Media geometrica: %lf", sqrt(a \* b));

return 0;

}



1. Se dă lungimea laturii unui pătrat. Să se afle perimetrul și aria pătratului.

#include <stdio.h>

#include <math.h>

int main () {

int a, aria;

printf ("Latura = ");

scanf ("%d", &a);

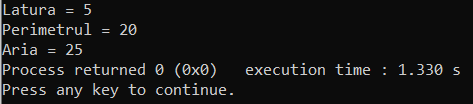
aria = pow(a, 2);

printf ("Perimetrul = %d\n", 4\*a);

printf ("Aria = %d", aria);

return 0;

}



1. Se dă lungimea muchiei unui cub. Să se determine aria și volumul cubului.

#include <stdio.h>

#include <math.h>

int main () {

int m, aria, vol;

printf ("Muchia = ");

scanf ("%d", &m);

aria = 6 \* pow(m, 2);

vol = pow(m, 3);

printf ("Volumul = %d\n", vol);

printf ("Aria = %d", aria);

return 0;

}



1. Se dă raza unui cerc. Să se afle lungimea cercului și aria discului mărginite de acest cerc.

#include <stdio.h>

#include <math.h>

int main () {

int r;

float lung, aria;

const float pi = 3.14;

printf ("Raza = ");

scanf ("%d", &r);

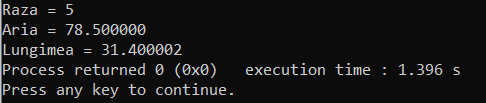
aria = pi \* pow(r, 2);

printf ("Aria = %f\n", aria);

printf ("Lungimea = %f", 2 \* pi \* r);

return 0;

}



1. Se dă lungimea unui cerc. Să se afle raza cercului și aria discului mărginite de acest cerc.

#include <stdio.h>

#include <math.h>

int main () {

float r, lung, aria;

const float pi = 3.14;

printf ("Lungimea = ");

scanf ("%f", &lung);

r = lung / (2 \* pi);

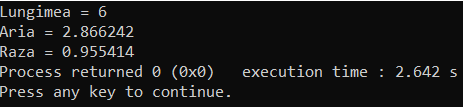
aria = pi \* pow(r, 2);

printf ("Aria = %f\n", aria);

printf ("Raza = %f", r);

return 0;

}



1. Se dă numărul natural n (n < 10000). Să se afișeze:
2. Ultima cifră a acestui număr;
3. Penultima cifră a acestui număr;
4. Restul și câtul împărțirii acestui număr la 9.

#include <stdio.h>

int main () {

int n;

printf ("Introduceti n: ");

scanf ("%d", &n);

if (n<10000) {

printf ("Ultima cifra este %d\n", n%10);

printf ("Penultima cifra este %d \n", (n/10)%10);

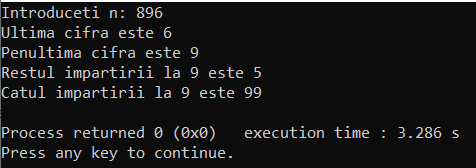
printf ("Restul impartirii la 9 este %d \n", n%9);

printf ("Catul impartirii la 9 este %d \n", n/9);

}

return 0;

}



1. Se dă un număr real n (n<10000). Să se calculeze:
2. Partea întreagă a lui n;
3. Partea fracționară a lui n;
4. Rotunjirea lui n la întregi.

#include <stdio.h>

int main () {

double n;

printf ("Introduceti n: ");

scanf ("%lf", &n);

if (n<10000) {

printf ("Partea intreaga este: %d \n", (int)n);

printf ("Partea fractionara este: %lf \n", (n - (int)n));

if ((n - (int)n)>=0.5) {

printf ("Numarul rotunjit pana la intregi: %d \n", (int)++n);

} else if ((n - (int)n)<0.5) {

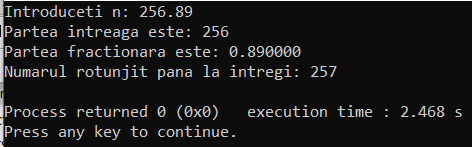
printf ("Numarul rotunjit pana la intregi: %d \n", (int)n);

}

return 0;

}

}



1. Se dau punctele A(x1, x2), B (y1, y2). Să se calculeze distanța AB și coordonatele mijlocului segmentului AB.

#include <stdio.h>

int main () {

int x1, x2, y1, y2, dist, x3, y3;

printf ("Coordonatele lui A: \n");

scanf ("%d %d", &x1, &x2);

printf ("Coordonatele lui B: \n");

scanf ("%d %d", &y1, &y2);

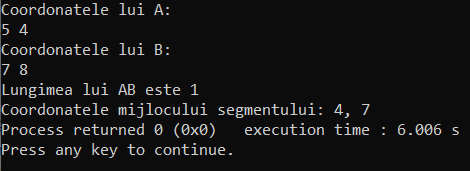
dist = sqrt(pow(x1-x2, 2) + pow(y1-y2, 2));

printf ("Lungimea lui AB este %d \n", dist);

printf ("Coordonatele mijlocului segmentului: %d, %d", (x1+x2)/2, (y1+y2)/2);

return 0;

}



1. Se dau numerele reale x și y. Să se afișeze TRUE dacă x > y, altfel FALSE.

#include <stdio.h>

int main () {

double x, y;

printf ("Introduceti x, y: ");

scanf ("%lf %lf", &x, &y);

if (x>y) {

printf ("TRUE");

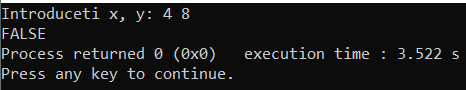
} else {

printf ("FALSE");

}

return 0;

}



1. Se dă nr n, care reprezintă măsura în grade a unghiului orientat format de semidreapta OM cu semiaxă pozitivă Ox în sistemul de axe ortogonale xOy. În ce cadran se află punctul M?

#include <stdio.h>

int main () {

double m;

printf ("Introduceti masura unghiului OM:");

scanf ("%lf", &m);

if ((m>=0) &&(m<90)) {

printf ("Punctul M se afla in cadranul 1");

} else if ((m>=90)&&(m<180)) {

printf ("Punctul M se afla in cadranul 2");

} else if ((m>=180)&&(m<270)) {

printf ("Punctul M se afla in cadranul 3");

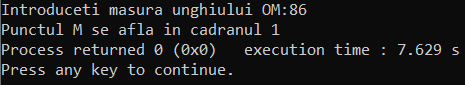
} else {

printf ("Punctul M se afla in cadranul 4");

}

return 0;

}



1. Se dă un număr natural m. Să se transforme măsura m a unui unghi din grade în radiani.

#include <stdio.h>

int main () {

double rad = 0.017453;

int m;

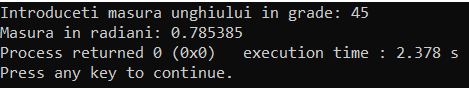
printf ("Introduceti masura unghiului in grade: ");

scanf("%d", &m);

printf ("Masura in radiani: %lf", m\*rad);

return 0;

}



1. Se dă un număr natural r. Să se transforme măsura r a unui unghi din radiani în grade.

#include <stdio.h>

int main () {

double g = 57.2958, r;

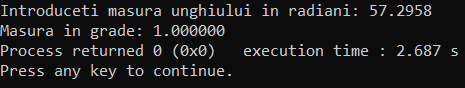
printf ("Introduceti masura unghiului in radiani: ");

scanf("%lf", &r);

printf ("Masura in grade: %lf", r/g);

return 0;

}



1. Se dă numărul natural n. Să se calculeze câte ore, minute, secunde sunt în:
2. N zile;
3. N săptămâni;
4. Luna mai.

#include <stdio.h>

int main() {

int n, zs = 7, mai=31, h = 24, m = 60, s = 60;

printf ("n= ");

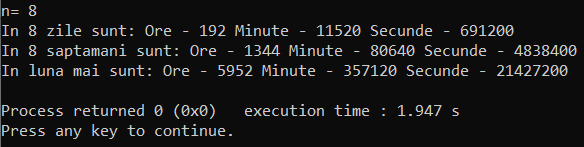
scanf ("%d", &n);

printf ("In %d zile sunt: Ore - %d Minute - %d Secunde - %d \n", n, n\*h, n\*h\*m, n\*h\*m\*s);

printf ("In %d saptamani sunt: Ore - %d Minute - %d Secunde - %d \n", n, n\*zs\*h, n\*zs\*h\*m, n\*zs\*h\*m\*s);

printf ("In luna mai sunt: Ore - %d Minute - %d Secunde - %d \n", n\*mai\*h, n\*mai\*h\*m, n\*mai\*h\*m\*s);

}



1. Se dă numărul natural n.
2. Să se transforme n metri în centimetri;
3. Să se transforme n kilograme în miligrame;
4. Câte tone întregi sunt în n kilograme?
5. Să se transforme n ani în luni, săptămâni, zile.

#include <stdio.h>

int main (){

int n;

printf ("n = ");

scanf ("%d",&n);

printf ("%d m = %.2f cm \n", n,(float)n/100);

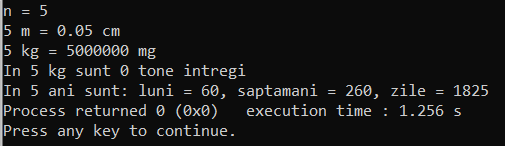
printf ("%d kg = %d mg \n", n, n\*1000000);

printf ("In %d kg sunt %d tone intregi \n", n, n/1000);

printf ("In %d ani sunt: luni = %d, saptamani = %d, zile = %d", n, n\*12, n\*52, n\*365);

return 0;

}



1. Se dau numerele naturale a, p, s. Să se afle cu cât va crește timp de a ani suma de s lei depusă la o bancă, dacă dobânda anuală este de p procente.

#include <stdio.h>

int main () {

int a, p, s;

float sum;

printf ("Introduceti rata dobanzii: ");

scanf ("%d", &p);

printf ("Introduceti numarul de ani: ");

scanf ("%d", &a);

printf ("Introduceti suma depusa :");

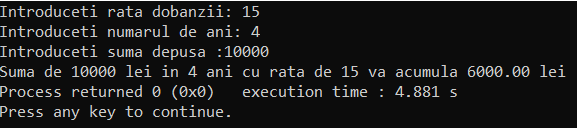
scanf ("%d", &s);

sum = s\*p\*a/100;

printf ("Suma de %d lei in %d ani cu rata de %d % va acumula %.2f lei", s, a, p, sum);

return 0;

}



1. Se dau variabilele numerice a și b. Să se schimbe între ele valorile acestor variabile.

#include <stdio.h>

int main () {

int a, b;

printf ("Variabilele inainte de schimbare: \na = ");

scanf ("%d", &a);

printf ("b = ");

scanf ("%d", &b);

int temp;

temp = a;

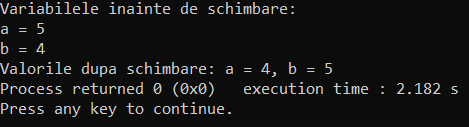
a=b;

b = temp;

printf ("Valorile dupa schimbare: a = %d, b = %d", a, b);

return 0;

}



1. Se dau variabilele numerice a, b, c. Să se schimbe între ele valorile acestor variabile, astfel încât:
2. a să aibă valoarea lui b;
3. b să aibă valoarea lui c;
4. c să aibă valoarea lui a.

#include <stdio.h>

int main () {

int a, b, c;

printf ("Variabilele inainte de schimbare: \na = ");

scanf ("%d", &a);

printf ("b = ");

scanf ("%d", &b);

printf ("c = ");

scanf ("%d", &c);

int temp1, temp2;

temp1 = a;

temp2 = b;

a=b;

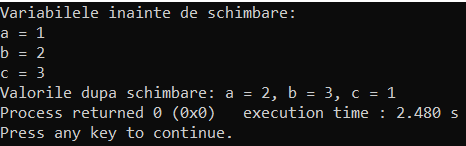
b=c;

c=temp1;

printf ("Valorile dupa schimbare: a = %d, b = %d, c = %d", a, b, c);

return 0;

}



1. Se citesc de la tastatură 4 numere reale a, b, c, d. Să se interschimbe circular de la dreapta la stânga valorile lor.

#include <stdio.h>

#include <math.h>

void main ()

{

double a, b, c, d;

int temp;

printf ("a = ");

scanf ("%lf", &a);

printf ("b = ");

scanf ("%lf", &b);

printf ("c = ");

scanf ("%lf", &c);

printf ("d = ");

scanf ("%lf", &d);

temp = a;

a = b;

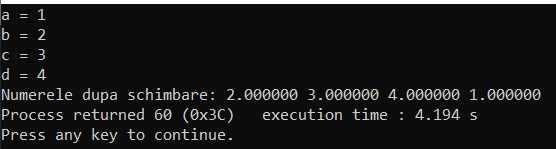
b = c;

c = d;

d = temp;

printf ("Numerele dupa schimbare: %lf %lf %lf %lf", a,b,c,d);

}



1. Se dau numerele naturale n și d. Să se calculeze:
2. 1 + 2 + ... + n;
3. 4 + 4 + d + 4 + 2d + 4 + 3d + ... + 4 + (n -1)d;
4. 5 + 5d + 5d^2 + 5d^3 + ... + 5d^(n-1).

#include <stdio.h>

#include <math.h>

void main ()

{

int i, n, d;

int sum1=0, sum2=0, sum3=0;

printf ("Introduceti n: ");

scanf("%d", &n);

printf ("Introduceti d: ");

scanf("%d", &d);

for (i=1; i<=n; i++)

{

sum1 = sum1 + i;

sum2 = sum2 + 4 + d\*(i-1);

sum3 = sum3 + 5\*pow(d, i-1);

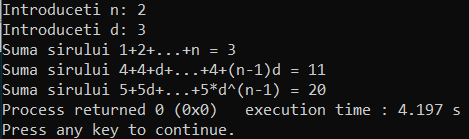
}

printf ("Suma sirului 1+2+...+n = %d \n", sum1);

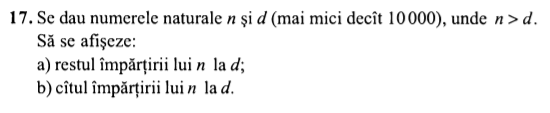
printf ("Suma sirului 4+4+d+...+4+(n-1)d = %d \n", sum2);

printf ("Suma sirului 5+5d+...+5\*d^(n-1) = %d", sum3);

}



23.



#include <stdio.h>

#include <math.h>

void main ()

{

int n, d;

printf ("Introduceti n: ");

scanf ("%d", &n);

printf ("Introduceti d: ");

scanf ("%d", &d);

if (n < d || n > 10000 || d > 10000)

{

printf ("Ati introdus date gresite!");

}

else

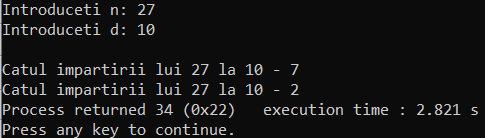
{

printf ("\nCatul impartirii lui %d la %d - %d", n, d, n%d);

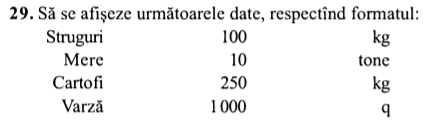
printf ("\nCatul impartirii lui %d la %d - %d", n, d, n/d);

}

}



24.



#include <stdio.h>

#include <math.h>

void main ()

{

char date[4][3][10] =

{

"Struguri",

" 100",

" kg",

" Mere",

" 10",

" tone",

" Cartofi",

" 250",

" kg",

" Varza",

" 1000",

" q"

};

for (int i = 0; i<4; i++)

{

for (int j = 0; j<3; j++)

{

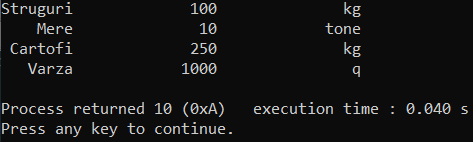
printf("%s\t", date[i][j]);

}

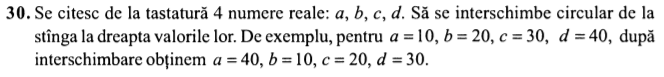
printf("\n");

}

}



25.



#include <stdio.h>

#include <math.h>

void main ()

{

double a, b, c, d;

int temp;

printf ("a = ");

scanf ("%lf", &a);

printf ("b = ");

scanf ("%lf", &b);

printf ("c = ");

scanf ("%lf", &c);

printf ("d = ");

scanf ("%lf", &d);

temp = d;

d = c;

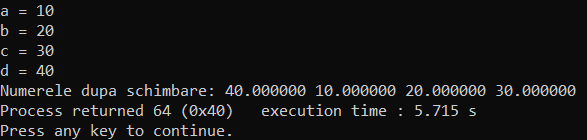
c = b;

b = a;

a = temp;

printf ("Numerele dupa schimbare: %lf %lf %lf %lf", a,b,c,d);

}



26.



#include <stdio.h>

#include <math.h>

void main ()

{

int n, i = 0, sum = 0;

printf ("n = ");

scanf ("%d", &n);

while (n%10!=0)

{

sum = sum + n%10;

n = n/10;

i++;

}

printf ("suma cifrelor = %d", sum);

}

27.



#include <stdio.h>

void main ()

{

int n, i=0, temp;

printf ("n: ");

scanf ("%d", &n);

while (n!=0)

{

temp=n%10;

i=temp+i\*10;

n=n/10;

}

printf ("Rasturnatul - %d", i);

}

28.



#include <stdio.h>

void main ()

{

int n;

printf ("n = ");

scanf ("%d", &n);

if (n>=27)

{

printf ("Ati introdus numar gresit!");

}

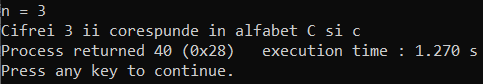
else

{

printf ("Cifrei %d ii corespunde in alfabet %c si %c", n, n+64, n+96);

}

}



29.



#include <stdio.h>

void main ()

{

char n;

printf ("litera = ");

scanf ("%c", &n);

if (n>=65 && n<=90)

{

printf ("Litera %c este pe pozitia %d", n, n-64);

}

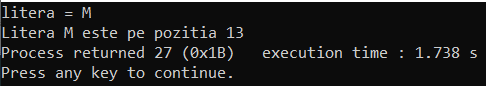
else if (n>=97 && n<=122)

{

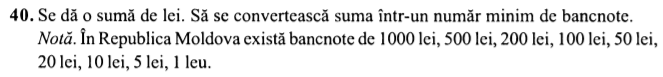
printf ("Litera %c este pe pozitia %d", n, n-96);

}

}



30.



#include <stdio.h>

void main ()

{

int b\_1000, b\_500, b\_200, b\_100, b\_50, b\_20, b\_10, b\_5, b\_1;

int temp;

int sum;

printf ("Introduceti suma Dvs: ");

scanf ("%d", &sum);

temp = sum;

b\_1000 = temp/1000;

temp = temp%1000;

b\_500 = temp/500;

temp = temp%500;

b\_200 = temp/200;

temp = temp%200;

b\_100 = temp/100;

temp = temp%100;

b\_50 = temp/50;

temp = temp%50;

b\_20 = temp/20;

temp = temp%20;

b\_10 = temp/10;

temp = temp%10;

b\_5 = temp/5;

temp = temp%5;

b\_1 = temp;

printf ("Suma de %d se poate diviza astfel: \n", sum);

printf ("%d bancnote de 1000 lei\n", b\_1000);

printf ("%d bancnote de 500 lei\n", b\_500);

printf ("%d bancnote de 200 lei\n", b\_200);

printf ("%d bancnote de 100 lei\n", b\_100);

printf ("%d bancnote de 50 lei\n", b\_50);

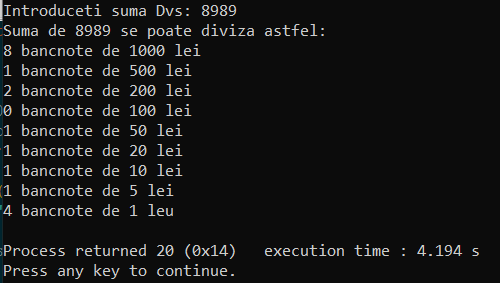
printf ("%d bancnote de 20 lei\n", b\_20);

printf ("%d bancnote de 10 lei\n", b\_10);

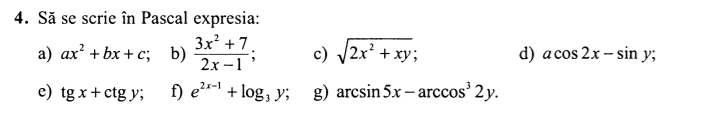
printf ("%d bancnote de 5 lei\n", b\_5);

printf ("%d bancnote de 1 leu\n", b\_1);

}



32//



#include <stdio.h>

#include <math.h>

void main()

{

double a = 1, b = 2, c = 3, x = 1, y = 1;

double exp\_a, exp\_b, exp\_c, exp\_d, exp\_e, exp\_f, exp\_g;

double e = 2.718281;

exp\_a = a \* pow(x, 2) + b \* x + c;

exp\_b = (3\*pow(x, 2)+7)/(2\*x - 1);

exp\_c = sqrt(2\*pow(x, 2) + x\*y);

exp\_d = a \* cos(2\*x) - sin(y);

exp\_e = (sin(x)/cos(x)) + (cos(y)/sin(y));

exp\_f = pow(e, (2\*x - 1)) + log(y);

exp\_g = asin(5\*x) - pow(acos(2\*y),3);

printf ("a) %lf\n", exp\_a);

printf ("b) %lf\n", exp\_b);

printf ("c) %lf\n", exp\_c);

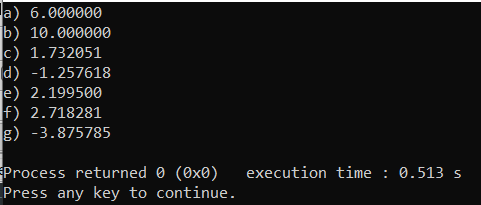
printf ("d) %lf\n", exp\_d);

printf ("e) %lf\n", exp\_e);

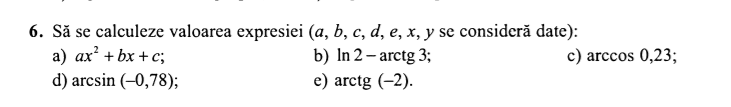
printf ("f) %lf\n", exp\_f);

printf ("g) %lf\n", exp\_g);

}



33//



#include <stdio.h>

#include <math.h>

void main()

{

double a = 1, b = 2, c = 3, x = 1, y = 1;

double exp\_a, exp\_b, exp\_c, exp\_d, exp\_e;

exp\_a = a \* pow(x, 2) + b \* x + c;

exp\_b = log(2) - atan(3);

exp\_c = acos(0.23);

exp\_d = asin(-0.78);

exp\_e = atan(-2);

printf ("a) %lf\n", exp\_a);

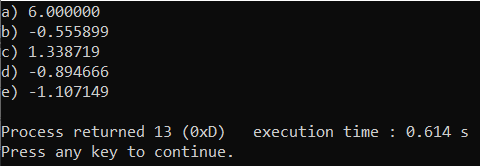
printf ("b) %lf\n", exp\_b);

printf ("c) %lf\n", exp\_c);

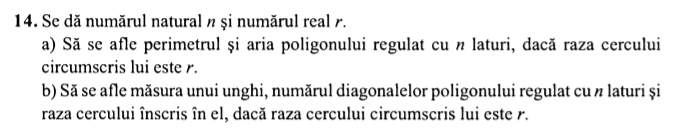
printf ("d) %lf\n", exp\_d);

printf ("e) %lf\n", exp\_e);

}



34//



#include <stdio.h>

#include <math.h>

void main()

{

double raza, aria, l, s, n, per, cot;

const double pi = 3.14;

printf("Introduceti numarul de laturi a poligonului:");

scanf("%lf",&n);

printf("Introduceti raza cercului circumscris poligonului:");

scanf("%lf",&raza);

s = pi/n;

l = (raza\*2\*sin(s));

per = n\*l;

cot = cos(s)/sin(s);

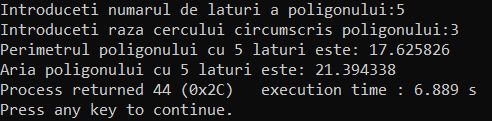
aria = (((n\*pow(l, 2))/4)\*cot);

int nr = (int) n;

printf("Perimetrul poligonului cu %d laturi este: %lf\n", nr, per);

printf("Aria poligonului cu %d laturi este: %lf", nr, aria);

}



35//



#include <stdio.h>

void main()

{

int grad1, min1,sec1;

int grad2, min2,sec2;

int dif1, dif2, dif3;

printf("Introduceti masura unghiului 1: ");

scanf("%d %d' %d''",&grad1,&min1,&sec1);

while(sec1 >= 60)

{

min1 += sec1/60;

sec1 = sec1-60\*(sec1/60);

}

while(min1 >= 60)

{

grad1 += min1/60;

min1 = min1-60\*(min1/60);

}

printf ("\n");

printf("Introduceti masura unghiului 2: ");

scanf("%d %d' %d''",&grad2,&min2,&sec2);

while(sec2 >= 60)

{

min2 += sec2/60;

sec2 = sec2-60\*(sec2/60);

}

while(min2 >= 60)

{

grad2 += min2/60;

min2 = min2-60\*(min2/60);

}

printf ("\n");

printf("Diferenta unghiurilor: ");

if(sec1 < sec2)

{

min1 = min1 - 1;

sec1 = sec1 + 60;

dif3 = sec1 - sec2;

}

else

{

dif3 = sec1 - sec2;

}

if(min1 < min2)

{

grad1 = grad1 - 1;

min1 = min1 + 60;

dif2 = min1 - min2;

}

else

{

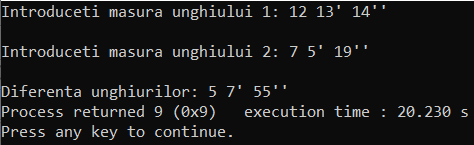
dif2 = min1 - min2;

}

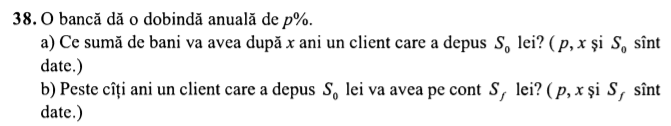
dif1 = grad1 - grad2;

printf("%d %d' %d''",dif1, dif2, dif3);

}



36//



#include <stdio.h>

#include <math.h>

void main()

{

int p, x, s0, sf, dob;

printf ("Suma depusa - ");

scanf ("%d", &s0);

printf ("Perioada depusa (ani) - ");

scanf ("%d", &x);

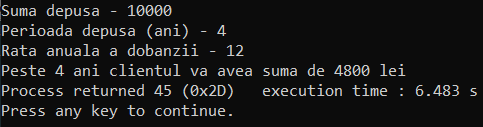
printf ("Rata anuala a dobanzii - ");

scanf ("%d", &p);

dob = (s0 \* p \* x)/100;

printf ("Peste %d ani clientul va avea suma de %d lei", x, dob);

}



#include <stdio.h>

#include <math.h>

void main()

{

int p, x, s0, sf, dob;

printf ("Suma depusa - ");

scanf ("%d", &s0);

printf ("Suma finala - ");

scanf ("%d", &sf);

printf ("Rata anuala a dobanzii - ");

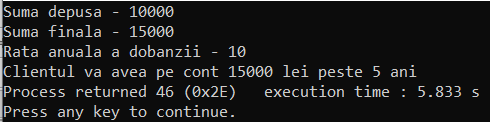
scanf ("%d", &p);

dob = sf - s0;

x = ((sf-s0)\*100)/(s0\*p);

printf ("Clientul va avea pe cont %d lei peste %d ani", sf, x);

}



37//



#include <stdio.h>

void main()

{

int m, n, p;

printf("m = ");

scanf("%d", &m);

printf("n = ");

scanf("%d", &n);

p = m - n;

if (p < 0)

{

printf("Minim este m");

}

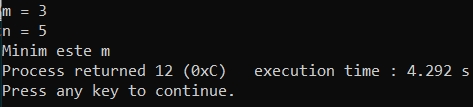
else

{

printf("Minim este n");

}

}



38//



#include <stdio.h>

void main()

{

int m, n, p;

printf("m = ");

scanf("%d", &m);

printf("n = ");

scanf("%d", &n);

p = m - n;

if (p > 0)

{

printf("Maxim este m");

}

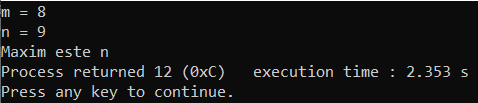
else

{

printf("Maxim este n");

}

}



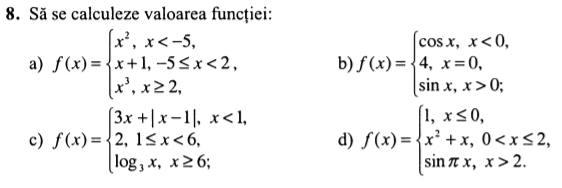
Popa Ecaterina

gr. TI-211

Probleme rezolvate – 31

**Tema: Instrucțiuni de ramificare (p. 21-27) – 36 probleme**

39//



a)

#include <stdio.h>

#include <math.h>

int main () {

double x, f;

printf ("x = ");

scanf ("%lf", &x);

if (x<-5) {

f=pow(x, 2);

printf ("f(x) = %.1lf", f);

} else if ((x>=-5) && (x<2)){

f=(x+1);

printf ("f(x) = %.1lf", f);;

} else {

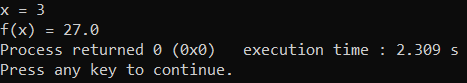
f=pow(x,3);

printf ("f(x) = %.1lf", f);

}

return 0;

}



#include <stdio.h>

#include <math.h>

int main () {

float x, fx;

printf ("x = ");

scanf ("%f", &x);

if (x<0) {

fx=cos(x);

printf ("f(x) = %.2f", fx);

} else if (x==0){

fx=(4);

printf ("f(x) = %.2f", fx);

} else {

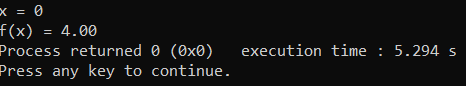
fx=sin(x);

printf ("f(x) = %.2f", fx);

}

return 0;

}



#include <stdio.h>

#include <math.h>

int main () {

int x, fx;

printf ("x = ");

scanf ("%d", &x);

if (x<1) {

fx=3\*x+abs(x-1);

printf ("f(x) = %d", fx);

} else if ((x>=1) && (x<6)){

fx=(2);

printf ("f(x) = %d", fx);

} else {

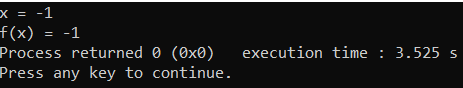
fx=log(x);

printf ("f(x) = %d", fx);

}

return 0;

}



#include <stdio.h>

#include <math.h>

int main () {

double x, fx;

const double pi=3.14;

printf ("x = ");

scanf ("%lf", &x);

if (x<=0) {

fx=1;

printf ("f(x) = %.2lf", fx);

} else if ((x>0) && (x<=2)){

fx=pow(x,2)+ x;

printf ("f(x) = %.2lf", fx);

} else {

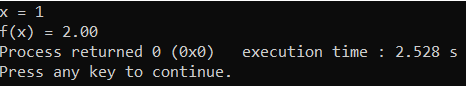
fx=sin(pi\*x);

printf ("f(x) = %.2lf", fx);

}

return 0;

}



40// Se dau numerele întregi m și n. Să se verifice dacă ele sunt consecutive.

#include <stdio.h>

#include <math.h>

int main () {

int m, n;

printf ("m = ");

scanf ("%d", &m);

printf ("n = ");

scanf ("%d", &n);

if ((m==n-1)||(n==m-1)) {

printf ("Numerele %d si %d sunt consecutive", m, n);

}

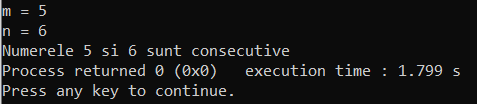
else {

printf ("Numerele %d si %d nu sunt consecutive", m, n);

}

return 0;

}



41// Să se rezolve ecuația: ax+b=0

#include <stdio.h>

int main () {

int a, b;

printf ("a = ");

scanf ("%d", &a);

printf ("b = ");

scanf ("%d", &b);

if (a!=0) {

printf ("Solutia este %d", -b/a);

}

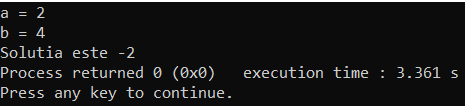
else {

printf ("Solutia este 0");

}

return 0;

}



42// Să se rezolve ecuația: ax^2+bx+c=0

#include <stdio.h>

#include <math.h>

int main () {

double a, b, c, delta, x, x1, x2;

printf ("Introduceti numarul a:");

scanf ("%lf", &a);

printf ("Introduceti numarul b:");

scanf ("%lf", &b);

printf ("Introduceti numarul c:");

scanf ("%lf", &c);

delta = pow(b,2)-4\*a\*c;

if (delta > 0) {

x1 = (- b - sqrt(delta))/(2 \* a);

x2 = (- b + sqrt(delta))/(2 \* a);

printf ("Solutiile sunt %.2f si %.2f", x1, x2);

} else if (delta == 0) {

x = (- b )/ (2 \* a);

printf ("Solutia este %.2f", x);

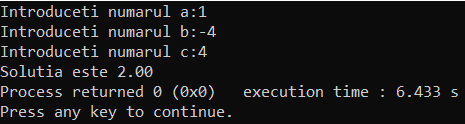
} else {

printf ("Ecuatia nu are solutii");

}

return 0;

}



43// Se dau numerele naturale a, b, c. Să se verifice dacă există un triunghi cu unghiurile de măsurile (în grade) a, b, c. În caz afirmativ, să se determine tipul triunghiului – echilateral, isoscel sau scalen.

#include <stdio.h>

int main () {

int a, b, c;

printf ("Introduceti unghiul a: ");

scanf ("%d", &a);

printf ("Introduceti unghiul b: ");

scanf ("%d", &b);

printf ("Introduceti unghiul c: ");

scanf ("%d", &c);

if (a+b+c==180) {

printf ("Triunghiul este valid\n");

if ((a==b)&&(a==c)&&(b==c)) {

printf ("Triunghiul este echilateral");

}

else if ((a==b)||(b==c)||(a==c)) {

printf ("Triunghiul este isoscel");

}

else {

printf ("Triunghiul este scalen");

}

}

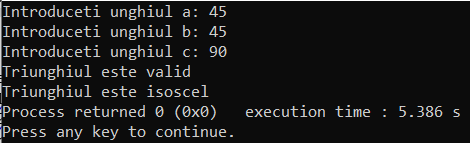
else {

printf ("Triunghiul nu este valid");

}

return 0;

}



44// Se dă numărul natural n, 0 < n < 8. Să se afișeze denumirea zilei corespunzătoare cifrei respective.

#include <stdio.h>

int main () {

int i;

printf ("Introduceti numarul zilei din saptamana: ");

scanf ("%d", &i);

if (i==1) {

printf ("Luni");

}

else if (i==2) {

printf ("Marti");

}

else if (i==3) {

printf ("Miercuri");

}

else if (i==4) {

printf ("Joi");

}

else if (i==5) {

printf ("Vineri");

}

else if (i==6) {

printf ("Sambata");

}

else if (i==7){

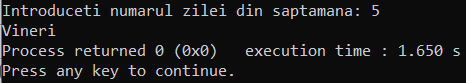
printf ("Duminica");

}

else {

printf ("Expresia nu este valabila")};

return 0;

}

45// Se dă numărul natural n. Să se determine ultima cifră a produsului 1\*2\*3\*…\*n

#include <stdio.h>

int main () {

int i, n, prod=1, u;

printf ("n= ");

scanf ("%d", &n);

for (i=1; i<=n; i++) {

prod=prod\*i;

}

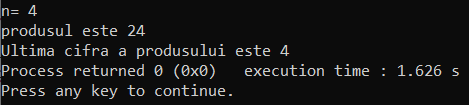
printf ("produsul este %d \n", prod);

u=prod%10;

printf ("Ultima cifra a produsului este %d", u);

return 0;

}



46// Se dă o literă. Să se verifice dacă ea este consoană sau vocală.

#include <stdio.h>

int main () {

char a;

printf ("Introduceti litera ");

scanf ("%c", &a);

if ((a=='a')||(a=='e')||(a=='i')||(a=='o')||(a=='u')||(a=='A')||(a=='E')||(a=='I')||(a=='O')||(a=='U')) {

printf ("vocala"); }

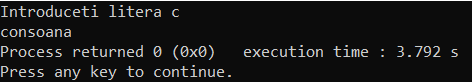
else {

printf ("consoana");

}

return 0;

}



47// Se dă numărul natural n. Să se afișeze denumirile lunilor cu numărul de zile n.

#include <stdio.h>

int main () {

int n;

printf ("n = ");

scanf ("%d", &n);

if (n==30) {

printf ("aprilie, iunie, septembrie, noiembrie");

}

else if (n==31) {

printf ("ianuarie, martie, mai, iulie, august, octombrie, decembrie");

}

else if ((n==28) ||(n==29)){

printf ("februarie");

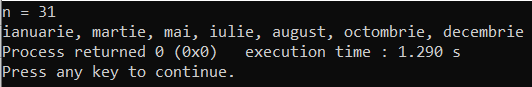
else {

printf ("nu exista");

}

return 0;

}



48// Scriți expresia logică: a mai mare ca 3.

#include <stdio.h>

int main () {

int a, b;

printf ("a = ");

scanf ("%d", &a);

if (a>3) {

printf ("a este mai mare ca 3");

}

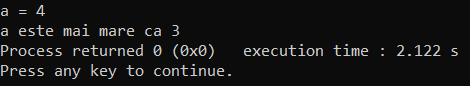
else {

printf ("a este mai mic ca 3");

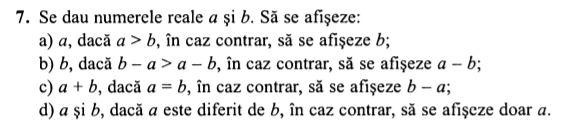
}

return 0;

}



49//



#include <stdio.h>

void main ()

{

double a, b;

printf ("a = ");

scanf ("%lf", &a);

printf ("b = ");

scanf ("%lf", &b);

if (a>b)

{

printf ("%lf", a);

}

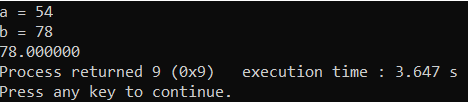
else

{

printf ("%lf", b);

}

}



#include <stdio.h>

void main ()

{

double a, b;

printf ("a = ");

scanf ("%lf", &a);

printf ("b = ");

scanf ("%lf", &b);

if (b-a>a-b)

{

printf ("%lf", b);

}

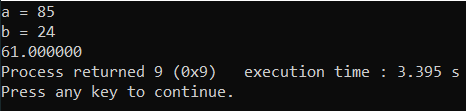
else

{

printf ("%lf", a-b);

}

}



#include <stdio.h>

void main ()

{

double a, b;

printf ("a = ");

scanf ("%lf", &a);

printf ("b = ");

scanf ("%lf", &b);

if (a==b)

{

printf ("%lf", a+b);

}

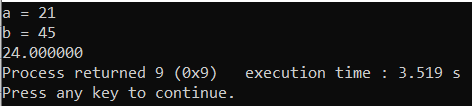
else

{

printf ("%lf", b-a);

}

}



#include <stdio.h>

void main ()

{

double a, b;

printf ("a = ");

scanf ("%lf", &a);

printf ("b = ");

scanf ("%lf", &b);

if (a!=b)

{

printf ("%lf %lf", a, b);

}

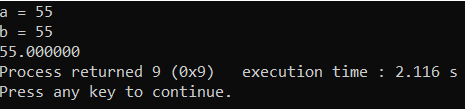
else

{

printf ("%lf", a);

}

}



50//



#include <stdio.h>

#include <math.h>

void main ()

{

double a, b, c;

double a2, b2, c2;

printf ("a = ");

scanf ("%lf", &a);

printf ("b = ");

scanf ("%lf", &b);

printf ("c = ");

scanf ("%lf", &c);

a2 = pow(a, 2);

b2 = pow(b, 2);

c2 = pow(c, 2);

if (a2+b2==c2 || b2+c2==a2 || a2+c2==b2)

{

printf ("Numerele sunt pitagoriene");

}

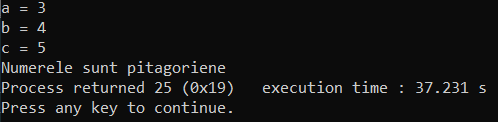
else

{

printf ("Numerele nu sunt pitagoriene");

}

}



51//



#include <stdio.h>

#include <math.h>

void main ()

{

int a, b;

printf ("a = ");

scanf ("%d", &a);

printf ("b = ");

scanf ("%d", &b);

if (b>a)

{

printf ("Ati introdus numere gresite!");

}

else

{

if (a%b==0)

{

printf ("b divide a");

}

else

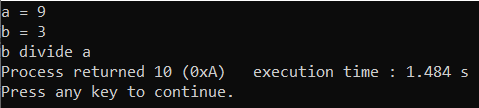
{

printf ("b nu divide a");

}

}

}



52//



#include <stdio.h>

#include <math.h>

void main ()

{

double a, b;

double x;

printf ("a = ");

scanf ("%d", &a);

printf ("b = ");

scanf ("%d", &b);

if (a==0 && b < 0)

{

printf ("x = R");

}

else if (a==0 && b > 0)

{

printf ("inecuatia nu are solutii");

}

else

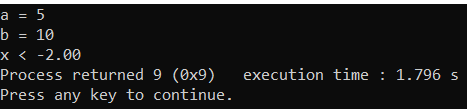
{

x = -b/a;

printf ("x < %.2lf", x);

}

}



#include <stdio.h>

#include <math.h>

void main ()

{

double a, b;

double x;

printf ("a = ");

scanf ("%d", &a);

printf ("b = ");

scanf ("%d", &b);

if (a==0 && b >= 0)

{

printf ("x = R");

}

else if (a==0 && b < 0)

{

printf ("inecuatia nu are solutii");

}

else

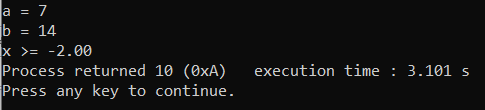
{

x = -b/a;

printf ("x >= %.2lf", x);

}

}



#include <stdio.h>

#include <math.h>

int main ()

{

double a, b, c, delta, x, x1, x2;

printf ("Introduceti numarul a:");

scanf ("%lf", &a);

printf ("Introduceti numarul b:");

scanf ("%lf", &b);

printf ("Introduceti numarul c:");

scanf ("%lf", &c);

delta = pow(b,2)-4\*a\*c;

if (delta > 0)

{

x1 = (- b - sqrt(delta))/(2 \* a);

x2 = (- b + sqrt(delta))/(2 \* a);

if (a>0)

{

printf ("%.2lf <= x <= %.2lf", x1, x2);

}

else

{

printf ("x <= %.2lf, x >= %.2lf", x1, x2);

}

}

else if (delta == 0)

{

x = (- b)/ (2 \* a);

printf ("Solutia este %.2f", x);

}

else

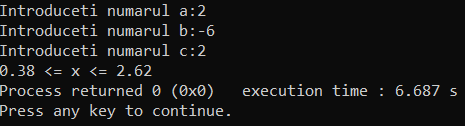
{

printf ("Ecuatia nu are solutii");

}

return 0;

}



#include <stdio.h>

#include <math.h>

int main ()

{

double a, b, c, delta, x, x1, x2;

printf ("Introduceti numarul a:");

scanf ("%lf", &a);

printf ("Introduceti numarul b:");

scanf ("%lf", &b);

printf ("Introduceti numarul c:");

scanf ("%lf", &c);

delta = pow(b,2)-4\*a\*c;

if (delta > 0)

{

x1 = (- b - sqrt(delta))/(2 \* a);

x2 = (- b + sqrt(delta))/(2 \* a);

if (a<0)

{

printf ("%.2lf < x < %.2lf", x1, x2);

}

else

{

printf ("x < %.2lf, x > %.2lf", x1, x2);

}

}

else if (delta == 0)

{

x = (- b)/ (2 \* a);

printf ("Solutia este %.2f", x);

}

else

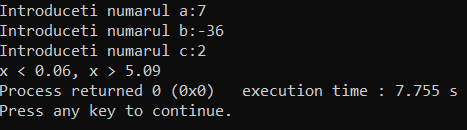
{

printf ("Ecuatia nu are solutii");

}

return 0;

}



53//



#include <stdio.h>

#include <math.h>

void main ()

{

int n;

printf ("n = ");

scanf ("%d", &n);

if (n%2==0)

{

printf ("%d este par\n", n);

}

if (n%2==0 && n%3==0)

{

printf ("%d este divizibil cu 2 si 3\n", n);

}

if (n%3==0 || n%4==0)

{

printf ("%d este divizibil cu 3 sau 4\n", n);

}

if (n%3==0 && n%4!=0)

{

printf ("%d este divizibil cu 3, dar nu cu 4\n", n);

}

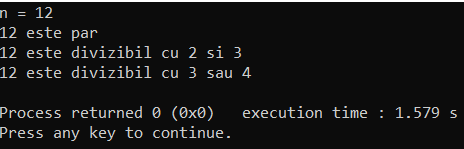
if (n%3!=0 || n%4!=0)

{

printf ("%d nu este divizibil cu 3 sau 4\n", n);

}

}



54//



#include <stdio.h>

#include <math.h>

void main ()

{

int a, b, c;

int temp, i;

printf ("a = ");

scanf ("%d", &a);

printf ("b = ");

scanf ("%d", &b);

printf ("c = ");

scanf ("%d", &c);

if (a<b && c<b && a<b)

{

temp = b;

b = c;

c = temp;

}

else if (c<b && b<a && c<a)

{

temp = a;

a = c;

c = temp;

}

else if (c<a && a<b && c<b)

{

temp = a;

a = c;

c = b;

b = temp;

}

else if (b<a && a<c && b<c)

{

temp = b;

b = a;

a = temp;

}

else if (b<c && c<a && b<a)

{

temp = a;

a = b;

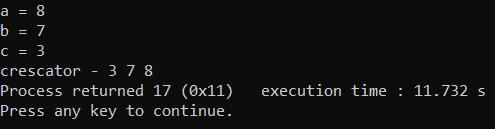
b = c;

c = temp;

}

printf ("crescator - %d %d %d", a, b, c);

}



#include <stdio.h>

#include <math.h>

void main ()

{

int a, b, c;

int temp, i;

printf ("a = ");

scanf ("%d", &a);

printf ("b = ");

scanf ("%d", &b);

printf ("c = ");

scanf ("%d", &c);

if (a<b && c<b && a<b)

{

temp = b;

b = c;

c = temp;

}

else if (c<b && b<a && c<a)

{

temp = a;

a = c;

c = temp;

}

else if (c<a && a<b && c<b)

{

temp = a;

a = c;

c = b;

b = temp;

}

else if (b<a && a<c && b<c)

{

temp = b;

b = a;

a = temp;

}

else if (b<c && c<a && b<a)

{

temp = a;

a = b;

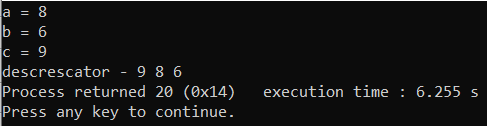
b = c;

c = temp;

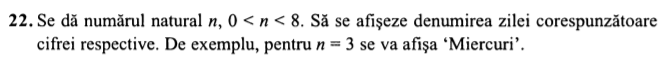
}

printf ("descrescator - %d %d %d", c, b, a);

}



55//



#include <stdio.h>

#include <math.h>

void main ()

{

int n;

printf ("n = ");

scanf ("%d", &n);

switch (n)

{

case 1:

printf ("Luni");

break;

case 2:

printf ("Marti");

break;

case 3:

printf ("Miercuri");

break;

case 4:

printf ("Joi");

break;

case 5:

printf ("Vineri");

break;

case 6:

printf ("Sambata");

break;

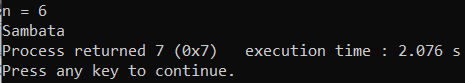
case 7:

printf ("Duminica");

break;

}

}



56//



#include <stdio.h>

#include <math.h>

void main ()

{

int a, b, c;

printf ("a = ");

scanf ("%d", &a);

printf ("b = ");

scanf ("%d", &b);

printf ("c = ");

scanf ("%d", &c);

printf ("Fractia initiala: %d/%d\n", a, b);

if (a%c==0 && b%c==0)

{

printf ("Fractia simplificata: %d/%d\n", a/c, b/c);

}

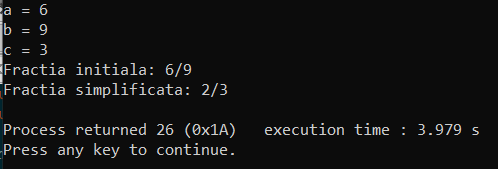
else

{

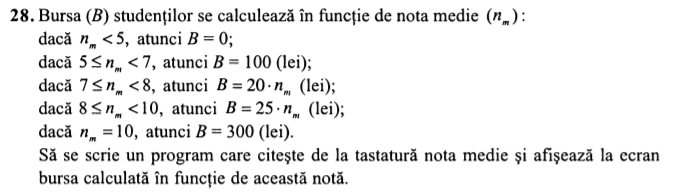
printf ("Fractia nu poate fi simplificata\n");

}

}



57//



#include <stdio.h>

#include <math.h>

void main ()

{

double media;

double bursa;

printf ("Introduceti nota medie a studentului: ");

scanf ("%lf", &media);

if (media<5)

{

bursa = 0;

}

if (media>= 5 && media<7)

{

bursa = 100;

}

if (media>=7 && media<8)

{

bursa = 20\*media;

}

if (media>= 8 && media<10)

{

bursa = 25\*media;

}

if (media == 10)

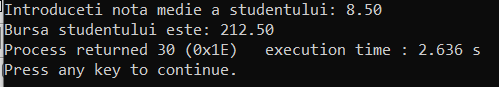
{

bursa = 300;

}

printf ("Bursa studentului este: %.2lf", bursa);

}



58//



#include <stdio.h>

#include <math.h>

void main ()

{

double a, b, c;

printf ("a = ");

scanf ("%lf", &a);

printf ("b = ");

scanf ("%lf", &b);

printf ("c = ");

scanf ("%lf", &c);

if (a!=b && a!=c && b!=c)

{

if (a < b && b < c)

{

printf ("a < b < c\n");

printf ("%.2lf < %.2lf < %.2lf\n", a, b, c);

}

if (c < b && b < a)

{

printf ("c < b < a\n");

printf ("%.2lf < %.2lf < %.2lf\n", c, b, a);

}

if (b < c && c < a)

{

printf ("b < c < a\n");

printf ("%.2lf < %.2lf < %.2lf\n", b, c, a);

}

if (a < c && c < b)

{

printf ("a < c < b\n");

printf ("%.2lf < %.2lf < %.2lf\n", a, c, b);

}

if (b < a && a < c)

{

printf ("b < a < c\n");

printf ("%.2lf < %.2lf < %.2lf\n", b, a, c);

}

if (c < a && a < b)

{

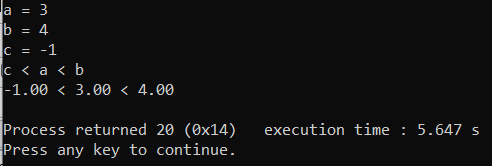
printf ("c < a < b\n");

printf ("%.2lf < %.2lf < %.2lf\n", c, a, b);

}

}

}



59//



#include <stdio.h>

#include <math.h>

void main ()

{

double a, b, c, d;

int flag;

printf ("a = ");

scanf ("%lf", &a);

printf ("b = ");

scanf ("%lf", &b);

printf ("c = ");

scanf ("%lf", &c);

printf ("d = ");

scanf ("%lf", &d);

if (a==b && c==d)

{

flag = 1;

}

else if (a==c && b==d)

{

flag = 1;

}

else if (a==d && c==b)

{

flag = 1;

}

else

{

flag = 0;

}

if (flag == 1)

{

printf ("Numerele pot fi laturile unui paralelogram");

}

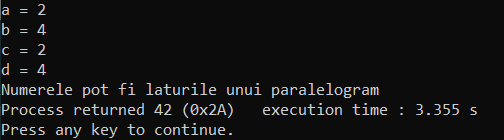
else

{

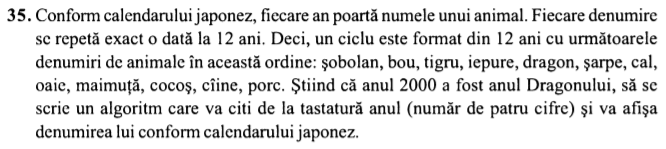
printf ("Numerele nu pot fi laturile unui paralelogram");

}

}



60//



#include <stdio.h>

#include <math.h>

void main ()

{

int an;

printf ("Introduceti anul dorit = ");

scanf ("%d", &an);

switch (an%12)

{

case 0:

printf ("%d este anul maimutei", an);

break;

case 1:

printf ("%d este anul cucosului", an);

break;

case 2:

printf ("%d este anul cainelui", an);

break;

case 3:

printf ("%d este anul porcului", an);

break;

case 4:

printf ("%d este anul sobolanului", an);

break;

case 5:

printf ("%d este anul boului", an);

break;

case 6:

printf ("%d este anul tigrului", an);

break;

case 7:

printf ("%d este anul iepurelui", an);

break;

case 8:

printf ("%d este anul dragonului", an);

break;

case 9:

printf ("%d este anul sarpelui", an);

break;

case 10:

printf ("%d este anul calului", an);

break;

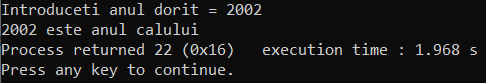
case 11:

printf ("%d este anul oii", an);

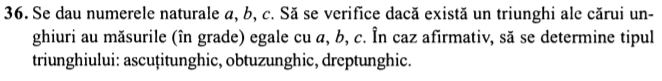
break;

}

}



61//



#include <stdio.h>

#include <math.h>

void main ()

{

int a, b, c;

printf ("a = ");

scanf ("%d", &a);

printf ("b = ");

scanf ("%d", &b);

printf ("c = ");

scanf ("%d", &c);

if (a+b+c!=180)

{

printf ("Triunghiul nu este valid");

}

else

{

if (a<90 && b<90 && c<90)

{

printf ("Triunghiul este ascutitunghic");

}

else if (a==90 || b==90 || c==90)

{

printf ("Triunghiul este dreptunghic");

}

else if (a>90 || b>90 || c>90)

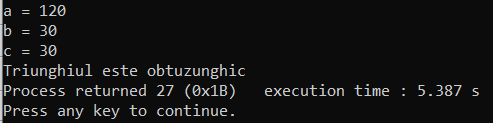
{

printf ("Triunghiul este obtuzunghic");

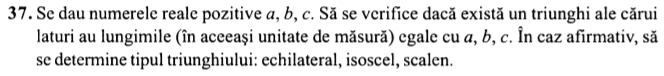
}

}

}



62//



#include <stdio.h>

#include <math.h>

void main ()

{

int a, b, c;

printf ("a = ");

scanf ("%d", &a);

printf ("b = ");

scanf ("%d", &b);

printf ("c = ");

scanf ("%d", &c);

if (a + b > c || a + b > c || a + b > c )

{

if (a==b || a==c || b==c)

{

printf ("Triunghiul este isoscel");

}

else if (a==b || b==c || c==a)

{

printf ("Triunghiul este echilateral");

}

else

{

printf ("Triunghiul este scalen");

}

}

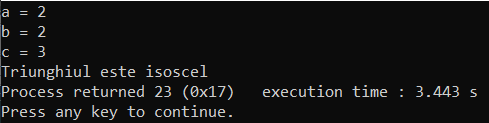
else

{

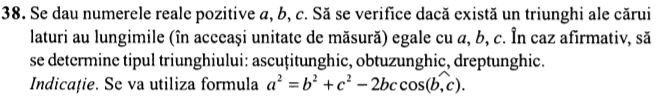
printf ("Triunghiul nu este valid");

}

}



63//



#include <stdio.h>

#include <math.h>

void main ()

{

double a, b, c;

double cos\_ab, cos\_bc, cos\_ac;

printf ("a = ");

scanf ("%lf", &a);

printf ("b = ");

scanf ("%lf", &b);

printf ("c = ");

scanf ("%lf", &c);

cos\_ab = (pow(a,2)+pow(b,2)-pow(c, 2))/(2\*a\*b);

cos\_bc = (pow(b,2)+pow(c,2)-pow(a, 2))/(2\*b\*c);

cos\_ac = (pow(a,2)+pow(c,2)-pow(b, 2))/(2\*a\*c);

if ((a + b) > c || (b + c) > a || (a + c) > b )

{

if ((cos\_ab > -1 && cos\_ab < 0) || (cos\_bc > -1 && cos\_bc < 0) || (cos\_ac > -1 && cos\_ac < 0))

{

printf ("Triunghiul este obtuzunghic");

}

else if (cos\_ab == 0 || cos\_ac == 0 || cos\_bc == 0)

{

printf ("Triunghiul este dreptunghic");

}

else if ((cos\_ab > 0 && cos\_ab < 1) || (cos\_bc > 0 && cos\_bc < 1) || (cos\_ac > 0 && cos\_ac < 1))

{

printf ("Triunghiul este ascutitunghic");

}

}

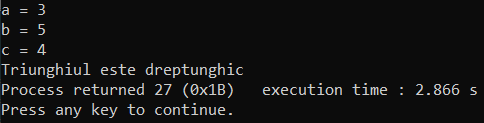
else

{

printf ("Triunghiul nu este valid");

}

}



64//



#include <stdio.h>

#include <math.h>

void main ()

{

int m, n, p;

printf ("m = ");

scanf ("%d", &m);

printf ("n = ");

scanf ("%d", &n);

printf ("p = ");

scanf ("%d", &p);

if ((m == n - 1 && n == p - 1) || (m == p - 1 && p == n - 1) || (n == m - 1 && m == p - 1) || (n == p - 1 && p == m - 1) || (p == m - 1 && m == n - 1) || (p == n - 1 && n == m - 1))

{

printf ("Numerele sunt consecutive");

}

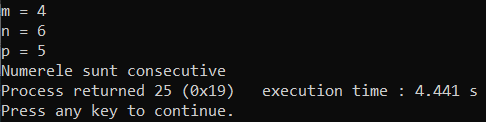
else

{

printf ("Numerele nu sunt consecutive");

}

}



65//



#include <stdio.h>

#include <math.h>

void main ()

{

int m, n, p, q;

int sum;

printf ("m = ");

scanf ("%d", &m);

printf ("n = ");

scanf ("%d", &n);

printf ("p = ");

scanf ("%d", &p);

printf ("q = ");

scanf ("%d", &q);

sum = m + n + p + q;

if (sum == 4\*m + 6 || sum == 4\*n + 6 || sum == 4\*p + 6 || sum == 4\*q + 6)

{

printf ("Numerele sunt consecutive");

}

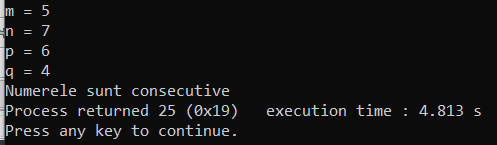
else

{

printf ("Numerele nu sunt consecutive");

}

}



66//



#include <stdio.h>

#include <math.h>

void main ()

{

int n, putere, u;

printf ("n = ");

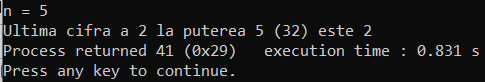
scanf ("%d", &n);

putere = pow(2, n);

u = putere%10;

printf ("Ultima cifra a 2 la puterea %d (%d) este %d", n, putere, u);

}



67//



#include <stdio.h>

#include <math.h>

#include <conio.h>

void main ()

{

char a, b, c;

printf ("Literele = ");

scanf ("%c %c %c", &a, &b, &c);

if(b == a - 1 && b == c + 1 || b == a + 1 && b == c - 1)

{

printf("\nLiterele sunt consecutive");

}

else if(a == b - 1 && a == c + 1 || a == b + 1 && a == c - 1)

{

printf("\nLiterele sunt consecutive");

}

else if(c == a - 1 && c == b + 1 || c == a + 1 && c == b - 1)

{

printf("\nLiterele sunt consecutive");

}

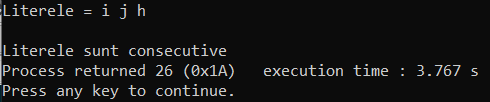
else

{

printf("\nLiterele nu sunt consecutive");

}

}



68//



#include <stdio.h>

void main ()

{

char m, n, p, q;

int sum = 0;

printf ("Introduceti 4 majuscule: ");

scanf ("%c %c %c %c", &m, &n, &p, &q);

sum = m + n + p + q;

if (sum == 4\*m + 6 || sum == 4\*n + 6 || sum == 4\*p + 6 || sum == 4\*q + 6)

{

printf ("Literele sunt consecutive");

}

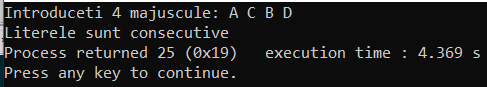
else

{

printf ("Literele nu sunt consecutive");

}

}



69//



#include <stdio.h>

#include <math.h>

#include <conio.h>

void main ()

{

int n, i;

int sum = 0;

printf ("n = ");

scanf ("%d", &n);

for (i = 2; i<=9; i++)

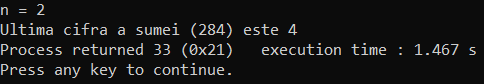
{

sum = sum + pow (i, n);

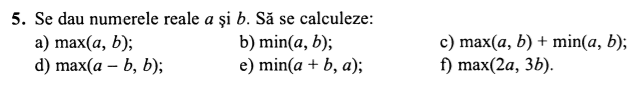
}

printf ("Ultima cifra a sumei (%d) este %d", sum, sum%10);

}



70//



#include <stdio.h>

#include <stdlib.h>

#include <math.h>

void main()

{

int a = rand()%100;

int b = rand()%100;

printf ("a = %d b = %d\n", a, b);

printf ("a) %d\n", max(a, b));

printf ("b) %d\n", min(a, b));

printf ("c) %d\n", max(a, b) + min(a, b));

printf ("d) %d\n", max((a-b), b));

printf ("e) %d\n", min((a+b), b));

printf ("f) %d\n", min(2\*a, 3\*b));

}

int max (int a, int b)

{

if (a > b)

return a;

else

return b;

}

int min (int a, int b)

{

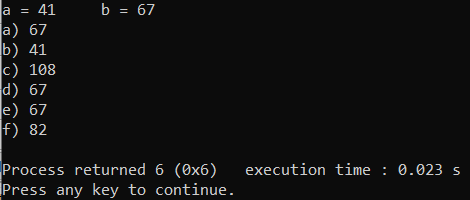
if (a < b)

return a;

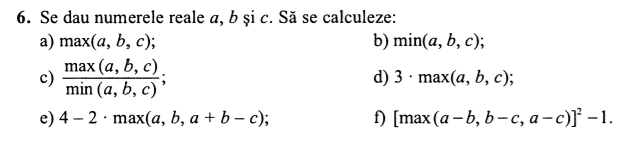
else

return b;

}



71//



#include <stdio.h>

#include <stdlib.h>

#include <math.h>

void main()

{

srand (time(NULL));

int a = rand()%100;

int b = rand()%100;

int c = rand()%100;

printf ("a = %d\t b = %d\t c = %d\n", a, b, c);

printf ("a) %d\n", max(a, b, c));

printf ("b) %d\n", min(a, b, c));

printf ("c) %d\n", max(a, b, c)/min(a, b, c));

printf ("d) %d\n", 3\*max(a, b, c));

printf ("e) %d\n", 4 - (2\*max(a, b, (a + b - c))));

}

int max (int a, int b, int c)

{

if (a > b && a > c)

return a;

else if (b > a && b > c)

return b;

else

return c;

}

int min (int a, int b, int c)

{

if (a < b && a < c)

return a;

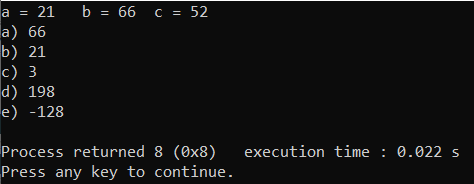
else if (b < a && b < c)

return b;

else

return c;

}



72//



#include <stdio.h>

#include <stdlib.h>

#include <math.h>

void main()

{

int x, y;

printf ("x = ");

scanf ("%d", &x);

printf ("y = ");

scanf ("%d", &y);

if (x > 0 && y > 0)

{

printf ("Punctul (%d, %d) se afla in cadranul 1", x, y);

}

else if (x < 0 && y > 0)

{

printf ("Punctul (%d, %d) se afla in cadranul 2", x, y);

}

else if (x < 0 && y < 0)

{

printf ("Punctul (%d, %d) se afla in cadranul 3", x, y);

}

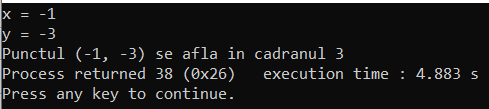
else

{

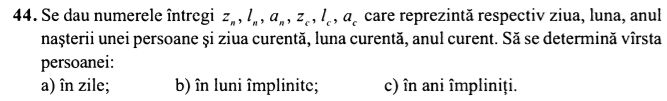
printf ("Punctul (%d, %d) se afla in cadranul 4", x, y);

}

}



73//



#include <stdio.h>

#include <stdlib.h>

#include <math.h>

void main()

{

int z\_n, l\_n, a\_n;

int z\_c, l\_c, a\_c;

int z, l, a;

printf ("Ziua curenta: ");

scanf ("%d/%d/%d", &z\_c, &l\_c, &a\_c);

printf ("Ziua de nastere: ");

scanf ("%d/%d/%d", &z\_n, &l\_n, &a\_n);

if (l\_c < l\_n)

{

if (z\_c < z\_n)

{

z = (a\_c - a\_n)\*365 + (12 - l\_n + l\_c)\*30 + (30 - z\_n + z\_c);

l = (a\_c - a\_n)\*12 + (12 - l\_n + l\_c);

a = a\_c - a\_n - 1;

}

else

{

z = (a\_c - a\_n)\*365 + (12 - l\_n + l\_c)\*30 + (z\_c - z\_n);

l = (a\_c - a\_n)\*12 + (12 - l\_n + l\_c);

a = a\_c - a\_n -1;

}

}

else

{

if (z\_c < z\_n)

{

z = (a\_c - a\_n)\*365 + (l\_c - l\_n - 1)\*30 + (30 - z\_n + z\_c);

l = (a\_c - a\_n)\*12 + (l\_c - l\_n - 1);

a = a\_c - a\_n;

}

else

{

z = (a\_c - a\_n)\*365 + (l\_c - l\_n)\*30 + (z\_c - z\_n);

l = (a\_c - a\_n)\*12 + (l\_c - l\_n);

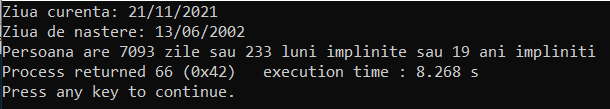
a = a\_c - a\_n;

}

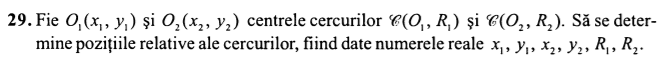
}

printf ("Persoana are %d zile sau %d luni implinite sau %d ani impliniti", z, l, a );

}



74//



#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{

int x1, y1;

int x2, y2;

int R1, R2;

double D;

printf("Primul cerc: \n");

printf ("x = ");

scanf("%d",&x1);

printf ("y = ");

scanf("%d",&y1);

printf ("R = ");

scanf("%d",&R1);

printf("Al doilea cerc: \n");

printf ("x = ");

scanf("%d",&x2);

printf ("y = ");

scanf("%d",&y2);

printf ("R = ");

scanf("%d",&R2);

D = sqrt(pow((x2-x1),2)+pow((y2-y1),2));

printf("Distanta dintre centrele cercurilor = %.2lf\n", D);

if(R1 > R2)

{

if(D > (R1 + R2))

{

printf("Cercuri exterioare");

}

else if(D ==(R1 + R2))

{

printf("Cercuri tangente exterioare");

}

else if(R1 - R2 < D && D < R1 + R2)

{

printf("Cercuri secante");

}

else if(D == R1 - R2)

{

printf("Cercuri tangente interioare");

}

else if(D < R1 - R2)

{

printf("Cercuri interioare");

}

else if(D == 0)

{

printf("Cercuri concentrice");

}

}

else

{

if(D > (R1 + R2))

{

printf("Cercuri exterioare");

}

else if(D == (R1 + R2))

{

printf("Cercuri tangente exterioare");

}

else if(R2 - R1 < D && D < R1 + R2)

{

printf("Cercuri secante");

}

else if(D == R2 - R1)

{

printf("Cercuri tangente interioare");

}

else if(D < R2 - R1)

{

printf("Cercuri interioare");

}

else if(D == 0)

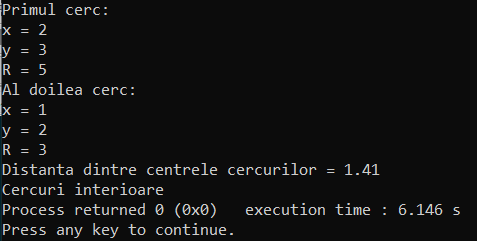
{

printf("Cercuri concentrice");

}

}

}



**Tema: Instrucțiuni ciclice (p. 35-44) – 26 probleme**

75//



#include <stdio.h>

int main ()

{

int a, k, i;

printf ("Introduceti k:");

scanf ("%d", &k);

for (i=1; i<=k; i++)

{

a = a + 4\*i;

printf ("%d ", a);

}

}



76//

#include <stdio.h>

int main ()

{

int k, a, d, prod=1, i;

printf ("Introduceti numarul de termeni: ");

scanf ("%d", &k);

printf ("Introduceti primul termen: ");

scanf ("%d", &a);

printf ("Introduceti ratia: ");

scanf ("%d", &d);

int sum = a;

for (i=1; i<=k; i++)

{

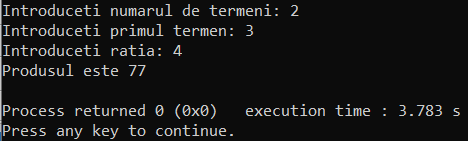
sum = sum+d;

prod = sum\*prod;

}

printf ("Produsul este %d \n", prod);

}



77//



#include <stdio.h>

void main ()

{

int factorial, factorial\_1=1, n, i;

printf ("Introduceti n:");

scanf ("%d", &n);

factorial = 2\*n;

for (i=1; i<=2\*n; i++)

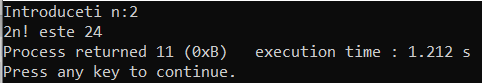
{

factorial\_1 = factorial\_1\*i;

}

printf ("2n! este %d", factorial\_1);

}



78//



#include <stdio.h>

#include <math.h>

int main ()

{

int k, a, i;

printf ("Introduceti numarul de termeni: ");

scanf ("%d", &k);

for (i=1; i<=k; i++)

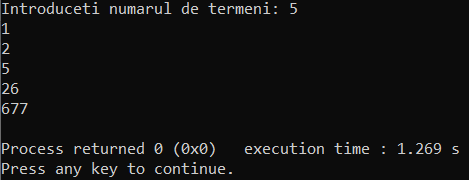
{

a = pow(a, 2) + 1;

printf ("%d \n", a);

}

}



79//



#include <stdio.h>

#include <math.h>

int main ()

{

int m, n, i, prod=1;

printf ("m: ");

scanf ("%d", &m);

printf ("n: ");

scanf ("%d", &n);

if (m<n)

{

for (i=1; i<n; i++)

{

if (i%m==0)

{

prod = prod\*i;

}

}

printf ("Produsul = %d", prod);

}

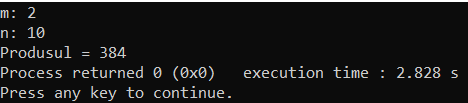
else

{

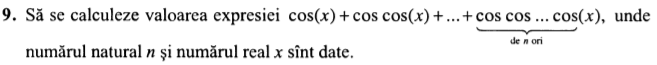
printf ("Ati introdus valori gresite");

}

}



80//



#include <stdio.h>

#include <math.h>

int main ()

{

int n, i;

double x;

printf ("x = ");

scanf ("%lf", &x);

printf ("n = ");

scanf ("%d", &n);

double expresia = x;

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

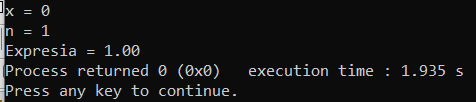
{

expresia = cos (expresia);

}

printf ("Expresia = %.2lf", expresia);

}



81//



#include <stdio.h>

#include <math.h>

int main ()

{

int n, numar, k=0;

printf ("n = ");

scanf ("%d", &n);

numar = n;

while (numar%2==0)

{

numar=numar/2;

k++;

}

if (numar==1)

{

printf ("%d este puterea lui 2", n);

}

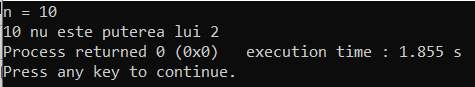
else

{

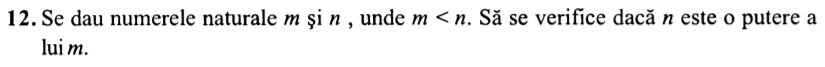
printf ("%d nu este puterea lui 2", n);

}

}



82//



#include <stdio.h>

#include <math.h>

int main ()

{

int m, n, i, k=1, m2, n8;

printf ("m: ");

scanf ("%d", &m);

printf ("n: ");

scanf ("%d", &n);

m2=m;

n8=n;

if (m2<n8)

{

while (n8%m2==0)

{

n8=n8/m2;

}

if (n8==1)

{

printf ("%d este puterea lui %d", n, m);

}

else

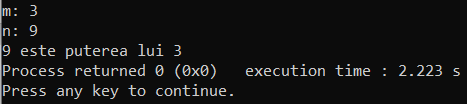
{

printf ("%d nu este puterea lui %d", n, m);

}

}

}



83//



#include <stdio.h>

void main ()

{

int factorial=1, n, i, sum=0;

printf ("Introduceti n:");

scanf ("%d", &n);

for (i=1; i<=n; i++)

{

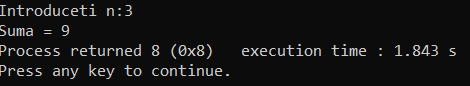
factorial = factorial \*i;

sum = sum+factorial;

}

printf ("Suma = %d", sum);

}



84//



#include<stdio.h>

void main()

{

int n, i, x=0, prim=0;

printf("Introduceti numarul:");

scanf("%d",&n);

x=n/2;

for(i=2; i<=x; i++)

{

if(n%i==0)

{

printf("%d nu este numar prim", n);

prim=1;

break;

}

}

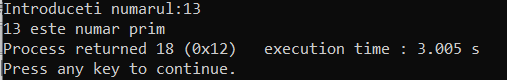
if(prim==0)

{

printf("%d este numar prim", n);

}

}



85//



#include<stdio.h>

void main()

{

int n,i,a,b,c;

printf ("Introduceti n:");

scanf("%d",&n);

a=0;

b=1;

printf ("Primii %d termeni din sirul lui Fibonacci: ", n);

for (i=1; i<=n; i++)

{

printf("%d ",a);

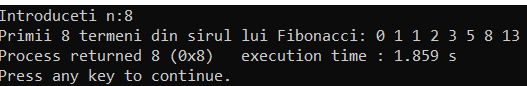
c = a + b;

a = b;

b = c;

}

}



86//



#include <stdio.h>

void main()

{

int n, b, i, m;

printf("n: ");

scanf("%d",&n);

printf ("Numerele prime pana la %d: \n", n);

for(b=2; b<=n; b++)

{

i=2;

while(b%i!=0)

{

i++;

}

if(i==b)

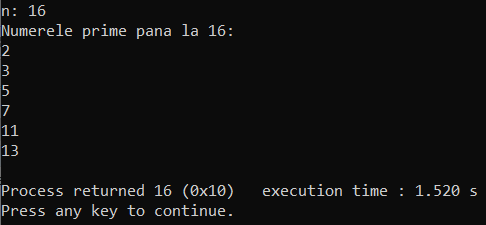
{

printf("%d\n",b) ;

}

}

}



87//



#include <stdio.h>

int main () {

int n, i=0, temp;

printf ("n: ");

scanf ("%d", &n);

while (n!=0) {

temp=n%10;

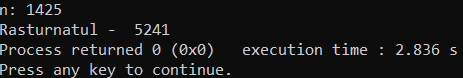
i=temp+i\*10;

n=n/10;

}

printf ("Rasturnatul - %d", i);

return 0;

}

88//



#include <stdio.h>

void main ()

{

int n, i=0, num, temp;

printf ("Introduceti numarul: ");

scanf ("%d", &n);

num=n;

while (n!=0)

{

temp=n%10;

i=temp+i\*10;

n=n/10;

}

printf ("Rasturnatul numarului este %d\n", i);

if (i==num)

{

printf ("Numarul este palindrom");

}

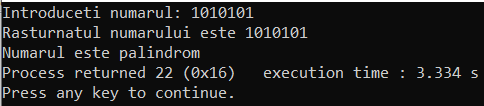
else

{

printf ("Numarul nu este palindrom");

}

}



89//



#include <stdio.h>

#include <math.h>

//media aritmetica = (1+3+4 + ...+)/n, n - numarul de termeni

void main ()

{

int suma=0, i, n; double med;

printf ("Introduceti n: ");

scanf ("%d", &n);

for (i=1; i<=n; i++)

{

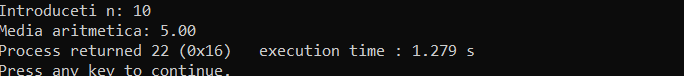
suma=suma+i;

}

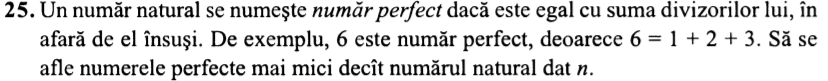
med=suma/n;

printf ("Media aritmetica: %.2lf", med);

}



90//



#include <stdio.h>

int main()

{

int n,perf = 0,i,j;

printf("n = ");

scanf("%d",&n);

for(i = 1; i <= n; i++)

{

perf = 0;

for(j=1; j <= i/2; j++)

{

if(i%j == 0)

{

perf += j;

if(i == perf)

{

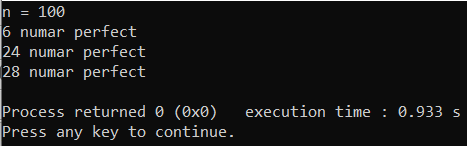
printf("%d numar perfect\n", i);

}

}

}

}

}

91//



#include <stdio.h>

int main()

{

int n,sum=0,temp;

printf("Numarul: ");

scanf("%d",&n);

while(n>0)

{

temp=n%10;

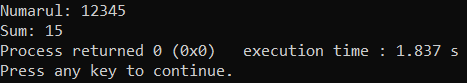
sum+=temp;

n/=10;

}

printf("Sum: %d",sum);

}



92//



#include <stdio.h>

#include <math.h>

int main()

{

int n, m;

double prod=1;

printf("Introduceti n : ");

scanf("%d",&n);

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

{

scanf("%d",&m);

prod=prod\*m;

}

printf("Media geometrica este : %lf",sqrt(prod));

}



93//



#include <stdio.h>

#include <math.h>

int main()

{

int n;

double sum=0,m;

printf("Introduceti n : ");

scanf("%d",&n);

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

{

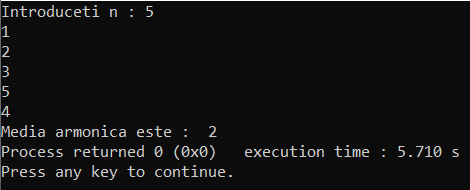
scanf("%lf",&m);

sum=sum+(double)(1/m);

}

printf("Media armonica este : %2.lf",n/sum);

}



94//



#include <stdio.h>

int main()

{

int n;

double a, prod=1;

printf("Introduceti a : ");

scanf("%lf",&a);

printf("Introduceti n : ");

scanf("%d",&n);

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

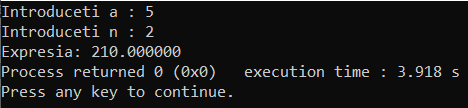
{

prod\*=(a+i);

}

printf("Expresia: %lf",prod);

}



95//



#include <stdlib.h>

#include <stdio.h>

#include <math.h>

int main()

{

int n, i, j;

printf("n: ");

scanf("%d",&n);

for(i=1; i<100; i++)

{

for(j=1; j<100; j++)

{

if(n==((pow(i,2))+pow(j,2)))

{

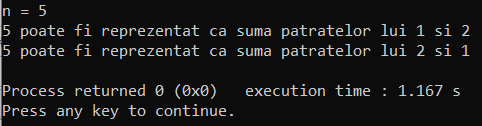
printf("%d poate fi reprezentat ca suma patratelor ,i,j);

}

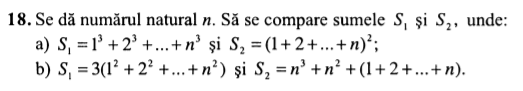
}

}

}



96//



#include <stdio.h>

#include <math.h>

#include <conio.h>

void main ()

{

int n, i;

int sum1 = 0;

int sum2 = 0;

int sum\_temp = 0;

printf ("n = ");

scanf ("%d", &n);

for (i = 1; i <= n; i++)

{

sum1 = sum1 + pow (i, 3);

sum\_temp = sum\_temp + i;

sum2 = pow (sum\_temp, 2);

}

if (sum1 < sum2)

{

printf ("S1 < S2\n");

printf ("%d < %d\n", sum1, sum2);

}

else if (sum1 > sum2)

{

printf ("S1 > S2\n");

printf ("%d > %d\n", sum1, sum2);

}

else

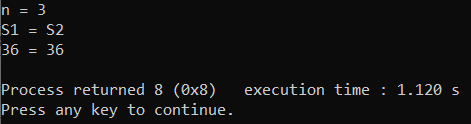
{

printf ("S1 = S2\n");

printf ("%d = %d\n", sum1, sum2);

}

}



#include <stdio.h>

#include <math.h>

#include <conio.h>

void main ()

{

int n, i;

int sum1 = 0;

int sum2 = 0;

int sum\_temp1 = 0;

int sum\_temp2 = 0;

printf ("n = ");

scanf ("%d", &n);

for (i = 1; i <= n; i++)

{

sum\_temp1 = sum\_temp1 + pow(i, 2);

sum1 = sum\_temp1 \* 3;

sum\_temp2 = sum\_temp2 + i;

sum2 = pow(i, 3) + pow(i, 2) + sum\_temp2;

}

if (sum1 < sum2)

{

printf ("S1 < S2\n");

printf ("%d < %d\n", sum1, sum2);

}

else if (sum1 > sum2)

{

printf ("S1 > S2\n");

printf ("%d > %d\n", sum1, sum2);

}

else

{

printf ("S1 = S2\n");

printf ("%d = %d\n", sum1, sum2);

}

}



97//



#include <stdio.h>

void main ()

{

int a, b;

int cmmdc, cmmmc;

printf ("a = ");

scanf ("%d", &a);

printf ("b = ");

scanf ("%d", &b);

for (int i = 1; i <= a && i <= b; i++)

{

if (a % i == 0 && b % i == 0)

{

cmmdc = i;

}

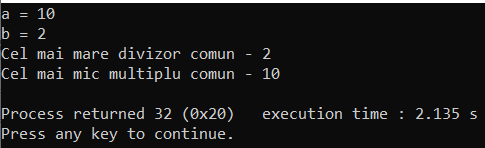
}

cmmmc = (a \* b) / cmmdc;

printf ("Cel mai mare divizor comun - %d\n", cmmdc);

printf ("Cel mai mic multiplu comun - %d\n", cmmmc);

}



98//



#include <stdio.h>

#include <math.h>

int main()

{

int n, num, c, temp;

printf("nr = ");

scanf("%d", &num);

printf("c = ");

scanf("%d", &c);

n = pow(num, 2);

while(n != 0)

{

temp = n%10;

if(temp == c)

{

printf("Cifra %d se contine in patratul lui %d", c, num);

break;

}

n/=10;

}

}



99//



#include <stdio.h>

void main()

{

int a, b, cmmdc;

printf("Fractia - ");

scanf("%d/%d", &a, &b);

int x = a;

int y = b;

for(int i = 1; i <= x && i <= y; i++)

{

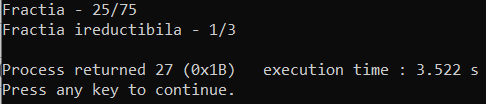
if(x % i==0 && y % i==0)

cmmdc = i;

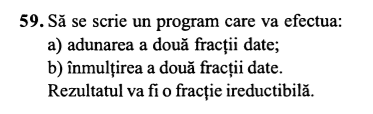
}

printf("Fractia ireductibila - %d/%d\n",a/cmmdc,b/cmmdc);

}



100//



#include <stdio.h>

void main()

{

int a, b, c, d, cmmdc;

printf("Prima fractie - ");

scanf("%d/%d", &a, &b); //

printf("A doua fractie - ");

scanf("%d/%d", &c, &d);

int x = (a\*d) + (c\*b);

int y = b\*d;

for(int i = 1; i <= x && i <= y; i++)

{

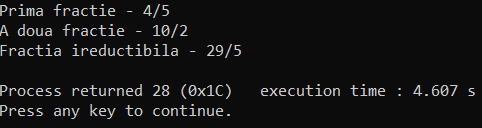
if(x % i==0 && y%i==0)

cmmdc = i;

}

printf("Fractia ireductibila - %d/%d\n",x/cmmdc,y/cmmdc);

}



#include <stdio.h>

void main()

{

int a, b, c, d, cmmdc;

printf("Prima fractie - ");

scanf("%d/%d", &a, &b); //

printf("A doua fractie - ");

scanf("%d/%d", &c, &d);

int x = a\*c;

int y = b\*d;

for(int i = 1; i <= x && i <= y; i++)

{

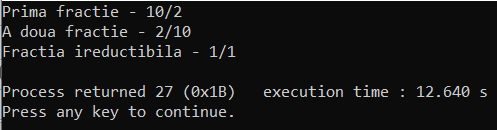
if(x % i==0 && y%i==0)

cmmdc = i;

}

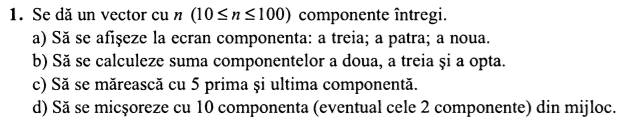
printf("Fractia ireductibila - %d/%d\n",x/cmmdc,y/cmmdc);

}



**Tema: Tablouri unidimensionale (p. 52-57) – 16 probleme**

101//



#include <stdio.h>

void main()

{

int tablou[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}, i;

printf ("Componenta a 3-a : %d\n", tablou [2]);

printf ("Componenta a 4-a : %d\n", tablou [3]);

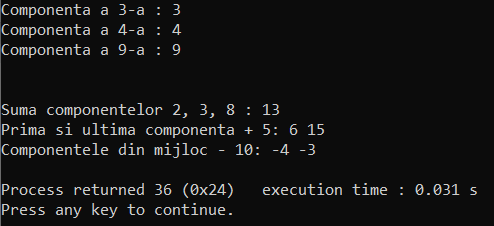
printf ("Componenta a 9-a : %d\n\n\n", tablou [8]);

printf ("Suma componentelor 2, 3, 8 : %d\n", tablou [1]+tablou[2]+tablou [7]);

printf ("Prima si ultima componenta + 5: %d %d\n", tablou[0]+5, tablou[9]+5);

printf ("Componentele din mijloc - 10: %d %d\n", tablou[5]-10, tablou[6]-10);

}



102//



#include <stdio.h>

#define N 100

void main()

{

int vector[N]={[1]=3,[3]=7,[5]=10}, i;

long int prod=1, sum=0;

for (i=0; i<=99; i++)

{

sum=sum+vector[i];

}

for (i=1; i<=99; i++)

{

if (vector[i]!=0)

{

prod=prod\*vector[i];

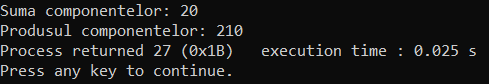
}

}

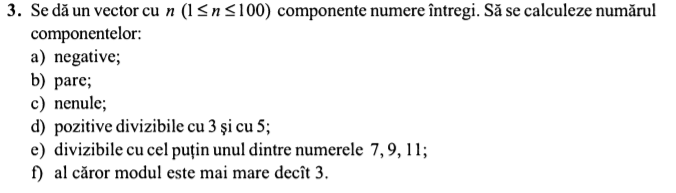
printf ("Suma componentelor: %ld\n", sum);

printf ("Produsul componentelor: %ld", prod);

}



103//



#include <stdio.h>

#include <stdlib.h>

#define N 100

void main()

{

int vector[N]={[1]=3,[3]=-7,[5]=10, [10]=-10, [12]=15}, i;

int negative=0, pare=0, nenule=0, poz35=0, div=0, mod3=0;

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

{

if(vector[i]<0)

{

negative=negative+1;

}

if (vector[i]%2==0)

{

pare=pare+1;

}

if (vector[i]!=0)

{

nenule=nenule+1;

}

if (vector[i]%3==0 && vector[i]%5==0 && vector[i]>0)

{

poz35=poz35+1;

}

if ((vector[i]%7==0) || (vector[i]%9==0) || (vector[i]%11==0))

{

div=div+1;

}

if (abs(vector[i])>3)

{

mod3=mod3+1;

}

}

printf ("Negative: %d \n", negative);

printf ("Pare: %d \n", pare);

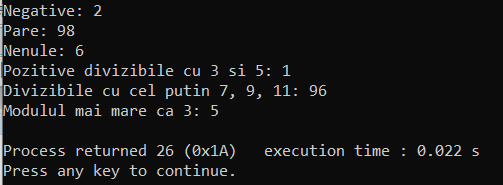
printf ("Nenule: %d \n", nenule);

printf ("Pozitive divizibile cu 3 si 5: %d \n", poz35);

printf ("Divizibile cu cel putin 7, 9, 11: %d \n", div);

printf ("Modulul mai mare ca 3: %d \n", mod3);

}



104//



#include <stdio.h>

void main()

{

int fib[100]={[0]=1, [1]=1}, i;

printf ("%d\n", fib[0]);

printf ("%d\n", fib[1]);

for (i=2; i<=10; i++)

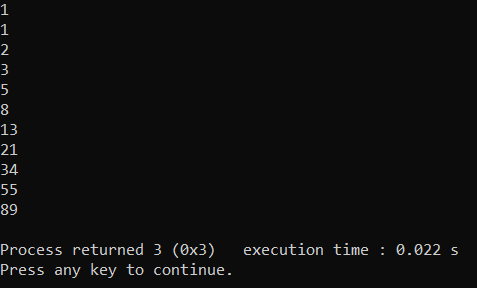
{

fib[i]=fib[i-1]+fib[i-2];

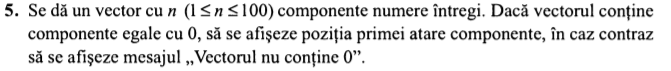
printf ("%d\n", fib[i]);

}

}



105//



#include <stdio.h>

void main()

{

int vector[10]={[0]=5, [2]=2, [7]=8}, i;

for (i=0; i<=9; i++)

{

if (vector[i]!=0)

{

printf("Vectorul nu contine 0\n");

}

else

{

printf ("%d\n", i);

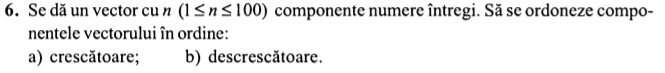
break;

}

}

}

106//



#include <stdio.h>

void main()

{

int i, j, a, n, cresc[30];

printf("Introduceti numarul de elemente \n");

scanf("%d", &n);

printf("Introduceti elementele tabloului\n");

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

{

scanf("%d", &cresc[i]);

}

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

{

for (j = i + 1; j < n; ++j)

{

if (cresc[i] > cresc[j])

{

a = cresc[i];

cresc[i] = cresc[j];

cresc[j] = a;

}

}

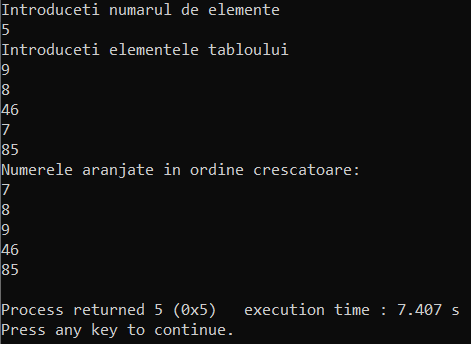
}

printf("Numerele aranjate in ordine crescatoare: \n");

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

printf("%d\n", cresc[i]);

}



#include <stdio.h>

void main()

{

int i, j, a, n, descresc[30];

printf("Introduceti numarul de elemente \n");

scanf("%d", &n);

printf("Introduceti elementele tabloului\n");

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

{

scanf("%d", &descresc[i]);

}

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

{

for (j = i + 1; j < n; ++j)

{

if (descresc[i] < descresc[j])

{

a = descresc[i];

descresc[i] = descresc[j];

descresc[j] = a;

}

}

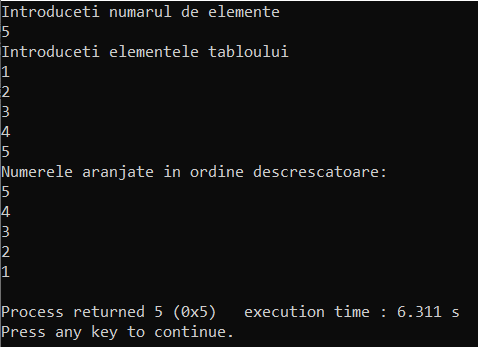
}

printf("Numerele aranjate in ordine descrescatoare: \n");

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

printf("%d\n", descresc[i]);

}



107//



#include <stdio.h>

void main ()

{

int vector[100], n, t, i, j, a, k;

printf ("Introduceti n: ");

scanf ("%d", &n);

printf ("Introduceti t: ");

scanf ("%d", &t);

printf("Introduceti elementele tabloului\n");

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

{

scanf("%d", &vector[i]);

}

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

{

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

{

for (j = i + 1; j < t; ++j)

{

if (vector[i] > vector[j])

{

a = vector[i];

vector[i] = vector[j];

vector[j] = a;

}

}

}

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

{

for (j=i+1; j<n; ++j)

{

if (vector[i] < vector[j])

{

a = vector[i];

vector[i] = vector[j];

vector[j] = a;

}

}

}

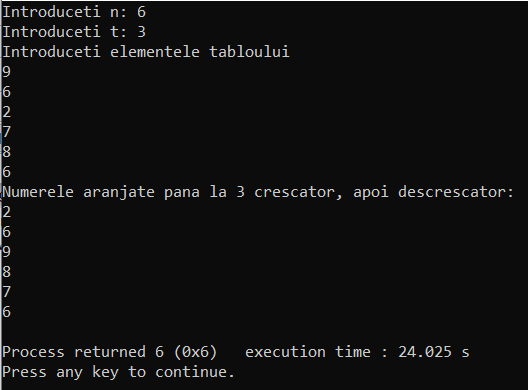
}

printf("Numerele aranjate pana la %d crescator, apoi descrescator: \n", t);

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

printf("%d\n", vector[i]);

}



108//



#include <stdio.h>

int main()

{

int i, j, n, temp=0, prim[100],k=0;

printf("Introduceti numarul de elemente : ");

scanf("%d",&n);

int tablou[n];

printf("Introduceti elementele : \n");

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

{

scanf("%d",&tablou[i]);

}

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

{

for(j=2; j<tablou[i]; j++)

{

if(tablou[i]%j==0)

{

temp=1;

break;

}

}

if(temp==0)

{

prim[k]=tablou[i];

k++;

}

}

printf ("Numerele prime sunt: \n");

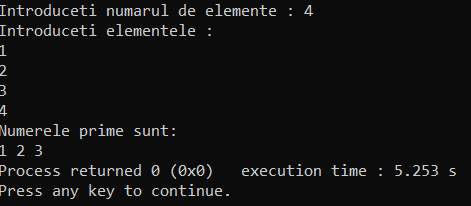
for(int l=0; l<k; l++)

{

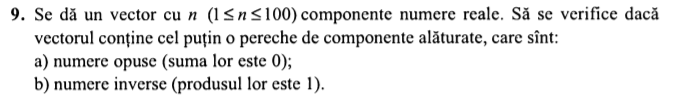
printf("%d ",prim[l]);

}

}



109//



#include <stdio.h>

void main ()

{

int arr[100], i, n, j, opus, invers;

printf ("Introduceti numarul de elemente: ");

scanf ("%d", &n);

printf("Introduceti elementele tabloului\n");

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

{

scanf("%d", &arr[i]);

}

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

{

if(arr[j]+arr[j+1]==0)

{

printf("Vectorul contine numere opuse alaturate\n");

}

if(arr[j+1]\*arr[j]==1)

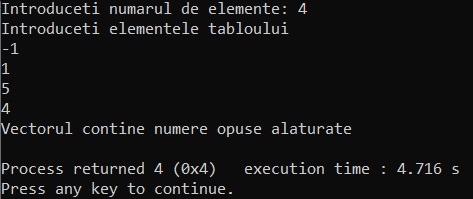
{

printf("Vectorul contine numere inverse alaturate\n");

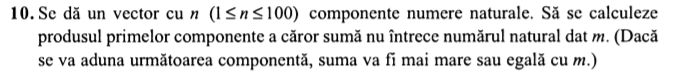
}

}

}



110//



#include <stdio.h>

void main ()

{

int arr[100], i, j, n, m, prod=1, sum=0;

printf ("Introduceti numarul de elemente: ");

scanf ("%d", &n);

printf ("Introduceti m: ");

scanf ("%d", &m);

printf("Introduceti elementele tabloului\n");

for (i=0; i<n-1; i++)

{

scanf ("%d\n", &arr[i]);

}

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

{

sum=sum+arr[i];

prod=prod\*arr[i];

if (sum<=m)

{

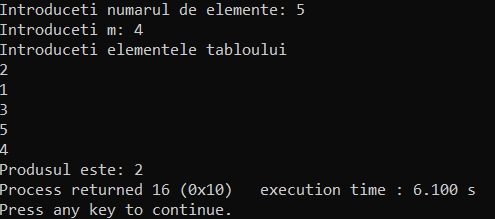
printf("Produsul este: %d", prod);

break;

}

}

}



111//



int main()

{

int i, j, n, max=0;

printf("Introduceti numarul de elemente : ");

scanf("%d",&n);

int tablou[n];

printf("Introduceti elementele : \n");

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

{

scanf("%d",&tablou[i]);

}

for(j=0; j<n-1; j++)

{

if(tablou[j]>tablou[j+1])

{

max=tablou[j];

}

else

{

max=tablou[j+1];

}

}

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

{

if(tablou[j]==max)

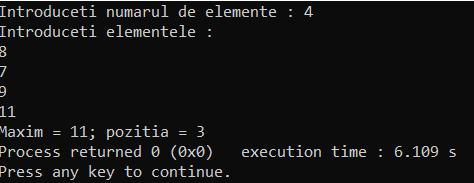
{

printf("Maxim = ", max);

}

}

}



#include <stdio.h>

int main()

{

int i, j, n, k, min=0;

printf("Introduceti numarul de elemente : ");

scanf("%d",&n);

int tablou[n];

printf("Introduceti elementele : \n");

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

{

scanf("%d",&tablou[i]);

}

for(int k=0; k<n-1; k++)

{

if(tablou[k]>tablou[k+1])

{

min=tablou[k+1];

}

else

{

min=tablou[k];

}

}

for(int k=0; k<n; k++)

{

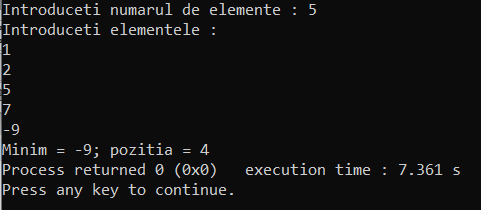
if(tablou[k]==min){

printf("Minim = %d; pozitia = %d", min, k);

}

}

}



#include <stdio.h>

int main()

{

int i, j, n, max=0, min=0, m;

printf("Introduceti numarul de elemente : ");

scanf("%d",&n);

int tablou[n];

printf("Introduceti elementele : \n");

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

{

scanf("%d",&tablou[i]);

}

printf("Introduceti m : ");

scanf("%d",&m);

for(j=0; j<n-1; j++)

{

if(min<m)

{

if(tablou[j]>tablou[j+1] )

{

max=tablou[j];

}

else

{

max=tablou[j+1];

}

}

else

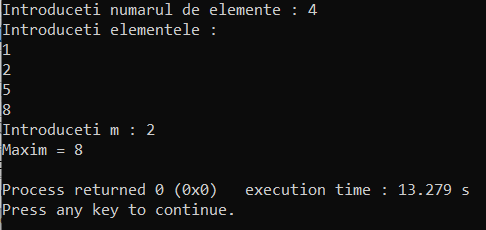
{

break;

}

printf("Maxim = %d\n",max);

}}



#include <stdio.h>

int main()

{

int i, j, n, max=0, min=0, m;

printf("Introduceti numarul de elemente : ");

scanf("%d",&n);

int tablou[n];

printf("Introduceti elementele : \n");

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

{

scanf("%d",&tablou[i]);

}

printf("Introduceti m : ");

scanf("%d",&m);

for(int k=0; k<n-1; k++)

{

if(min<m)

{

if(tablou[k]>tablou[k+1])

{

min=tablou[k+1];

}

else

{

min=tablou[k];

}

}

else

{

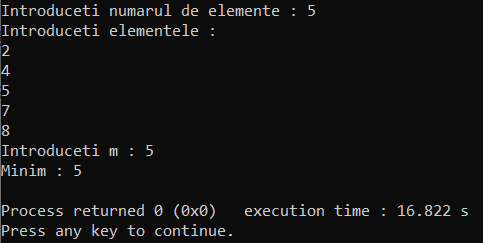
break;

}

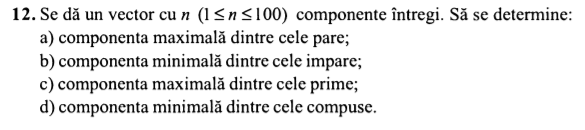
}

printf("Minim : %d\n", min);

}



112//



#include <stdio.h>

int main()

{

int i, j, n, max=0, min=0, k=0;

printf("Introduceti numarul de elemente : ");

scanf("%d",&n);

int tablou[n];

int par[n];

printf("Introduceti elementele : \n");

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

{

scanf("%d",&tablou[i]);

}

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

{

if(tablou[j]%2==0)

{

par[k]=tablou[j];

k++;

}

}

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

{

printf("%d\n",par[i]);

}

for(int l=0; l<k-1; l++)

{

if(par[l]>par[l+1] )

{

max=par[l];

}

else

{

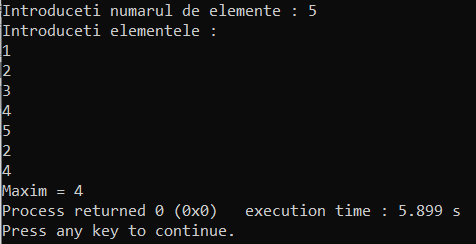
max=par[l+1];

}

}

printf("Maxim = %d",max);

}



#include <stdio.h>

int main()

{

int i, j, n, max=0, min=0, k=0;

printf("Introduceti numarul de elemente : ");

scanf("%d",&n);

int tablou[n];

int impar[n];

printf("Introduceti elementele : \n");

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

{

scanf("%d",&tablou[i]);

}

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

{

if(tablou[j]%2!=0)

{

impar[k]=tablou[j];

k++;

}

}

for(int l=0; l<k-1; l++)

{

if(impar[l]<impar[l+1] )

{

min=impar[l];

}

else

{

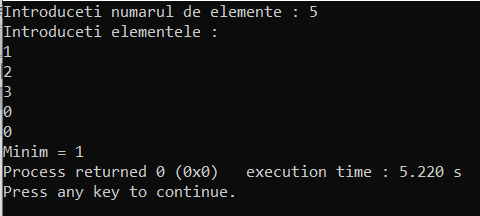
min=impar[l+1];

}

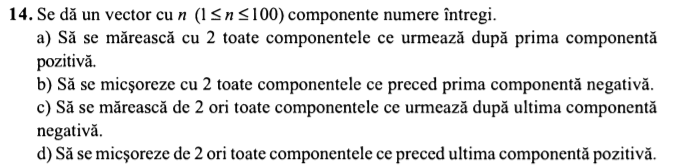
}

printf("Minim = %d",min);

}



113//



#include <stdio.h>

int main()

{

int n;

printf("Introduceti numarul de elemente : ");

scanf("%d",&n);

int tablou[n];

printf("Introduceti elementele : \n");

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

{

scanf("%d",&tablou[i]);

}

for(int i=1; i<n; i++){

tablou[i]=tablou[i]+2;

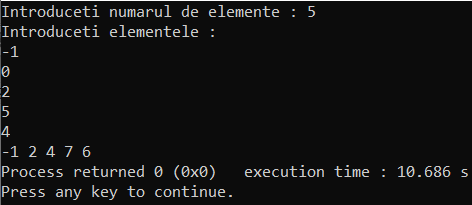
}

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

printf("%d ",tablou[i]);

}

}



114//

#include <stdio.h>

#include <stdlib.h>

int main()

{

int n, dif=0, i;

printf("Introduceti numarul de elemente : ");

scanf("%d",&n);

int tablou[n];

printf("Introduceti elementele : \n");

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

{

scanf("%d",&tablou[i]);

}

for(i=1;i<n;i++){

if(tablou[i]<0){

dif=abs(i-n);

}

}

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

tablou[i]-=2;

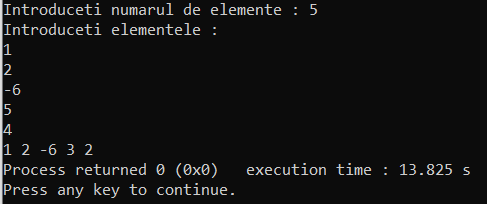
}

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

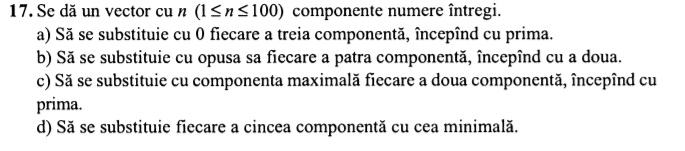
printf("%d",tablou[i]);

}

}



115//



#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{

int n, i;

printf("Introduceti numarul de elemente : ");

scanf("%d",&n);

int tablou[n];

int zero[n];

printf("Introduceti elementele : \n");

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

{

scanf("%d",&tablou[i]);

}

for(i=0; i<n; i+=3)

{

tablou[i]=0;

}

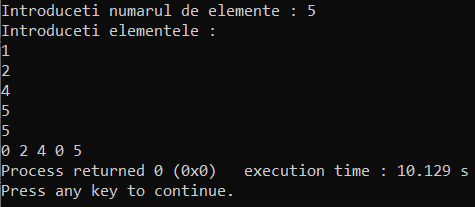
for(i=0; i<n; i++)

{

printf("%d ",tablou[i]);

}

}



116//

#include <stdio.h>

#include <stdlib.h>

int main()

{

int n,i;

printf("Introduceti numarul de elemente : ");

scanf("%d",&n);

int tablou[n];

printf("Introduceti elementele : \n");

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

{

scanf("%d",&tablou[i]);

}

for(int i=1; i<n; i+=4)

{

tablou[i]=-tablou[i];

}

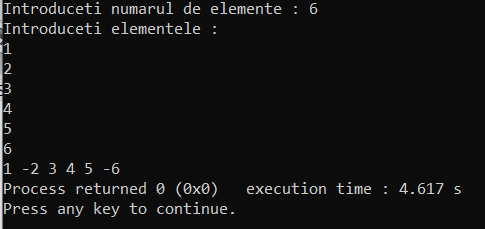
for(int i=0; i<n; i++)

{

printf("%d ",tablou[i]);

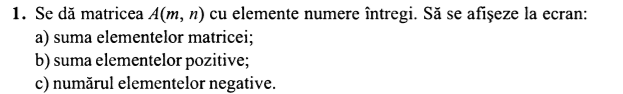
}

}



**Tema: Tablouri bidimensionale (p. 64-67) – 3 probleme**

117//



#include <stdio.h>

#include <stdlib.h>

void main ()

{

int m, n, i, j, sum=0;

printf ("Introduceti numarul de randuri:");

scanf ("%d", &m);

printf ("Introduceti numarul de coloane:");

scanf ("%d", &n);

int matrice [m][n];

printf ("Introduceti elementele matricei:\n");

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

{

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

{

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

}

}

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

{

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

{

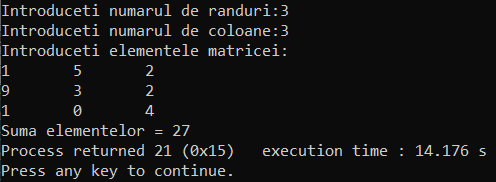
sum=sum+matrice[i][j];

}

}

printf ("Suma elementelor = %d", sum);

}



#include <stdio.h>

#include <stdlib.h>

void main ()

{

int m, n, i, j, sum=0;

printf ("Introduceti numarul de randuri:");

scanf ("%d", &m);

printf ("Introduceti numarul de coloane:");

scanf ("%d", &n);

int matrice [m][n];

printf ("Introduceti elementele matricei:\n");

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

{

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

{

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

}

}

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

{

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

{

if (matrice[i][j]>0)

{

sum=sum+matrice[i][j];

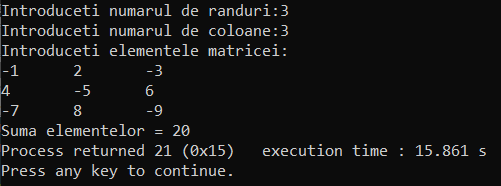
}

}

}

printf ("Suma elementelor = %d", sum);

}



#include <stdio.h>

#include <stdlib.h>

void main ()

{

int m, n, i, j, sum=0;

printf ("Introduceti numarul de randuri:");

scanf ("%d", &m);

printf ("Introduceti numarul de coloane:");

scanf ("%d", &n);

int matrice [m][n];

printf ("Introduceti elementele matricei: ");

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

{

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

{

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

}

}

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

{

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

{

printf ("%d\t", matrice[i][j]);

}

printf ("\n");

}

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

{

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

{

if (matrice[i][j]<0)

{

sum=++sum;

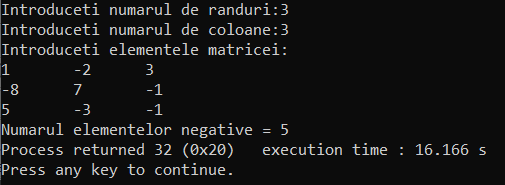
}

}

}

printf ("Numarul elementelor negative = %d", sum);

}



118//



#include <stdio.h>

int main ()

{

int i, j;

int row, col;

int arr[10][10], sum, count=0;

printf ("Introduceti numarul de randuri: ");

scanf ("%d", &row);

printf ("Introduceti numarul de coloane: ");

scanf ("%d", &col);

printf ("\n");

printf ("Introduceti elementele matricei: ");

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

{

for (j=0; j<col; j++)

{

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

}

}

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

{

for (j = 0; j < col; j++)

{

if (arr[i][j]>0)

{

sum=sum+arr[i][j];

count++;

}

}

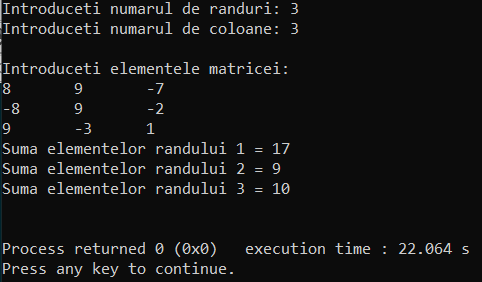
printf("Suma elementelor randului %d = %d\n", i+1, sum);

sum = 0;

}

printf ("\n");

}



119//



#include <stdio.h>

int main ()

{

int i, j;

int row, col;

int arr[10][10], sum=0, count=0;

printf ("Introduceti numarul de randuri: ");

scanf ("%d", &row);

printf ("Introduceti numarul de coloane: ");

scanf ("%d", &col);

printf ("\n");

printf ("Introduceti elementele matricei: ");

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

{

for (j=0; j<col; j++)

{

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

}

}

for (j = 0; j < col; ++j)

{

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

{

if (arr[i][j]<0)

{

sum = sum + arr[i][j];

count++;

}

}

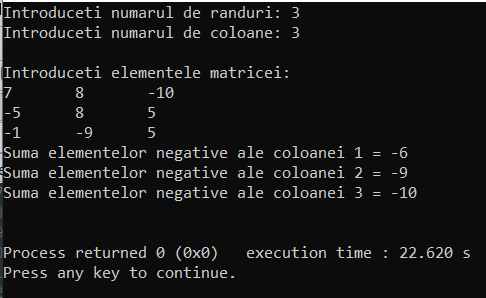
printf("Suma elementelor negative ale coloanei %d = %d\n", j+1, sum);

sum = 0;

}

printf ("\n");

}



120//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int row, col, i, j, max, poz\_row, poz\_col;

printf ("Introduceti numarul de randuri: ");

scanf ("%d", &row);

printf ("Introduceti numarul de coloane: ");

scanf ("%d", &col);

int matrice [row][col];

printf ("\n");

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

{

for (j = 0; j < col; j++)

{

matrice[i][j] = rand()%100;

}

}

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

{

for (j = 0; j < col; j++)

{

printf ("%d\t", matrice[i][j]);

}

printf ("\n");

}

max = matrice[0][0];

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

{

for (j = 0; j < col; j++)

{

if (matrice[i][j] > max)

{

max = matrice[i][j];

poz\_row = i+1;

poz\_col = j+1;

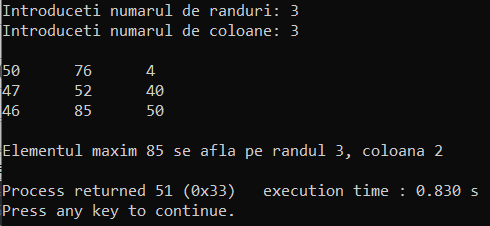
}

}

}

printf ("\nElementul maxim %d se afla pe randul %d, coloana %d\n", max, poz\_row, poz\_col);

}



121//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int row, col, i, j, min, poz\_row, poz\_col;

printf ("Introduceti numarul de randuri: ");

scanf ("%d", &row);

printf ("Introduceti numarul de coloane: ");

scanf ("%d", &col);

int matrice [row][col];

printf ("\n");

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

{

for (j = 0; j < col; j++)

{

matrice[i][j] = rand()%100;

}

}

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

{

for (j = 0; j < col; j++)

{

printf ("%d\t", matrice[i][j]);

}

printf ("\n");

}

min = matrice[0][0];

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

{

for (j = 0; j < col; j++)

{

if (matrice[i][j] < min)

{

min = matrice[i][j];

poz\_row = i+1;

poz\_col = j+1;

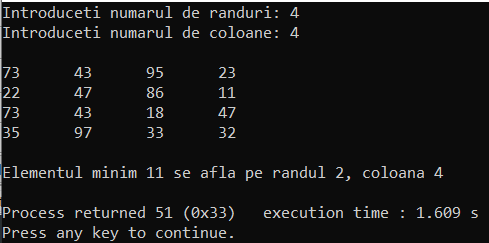
}

}

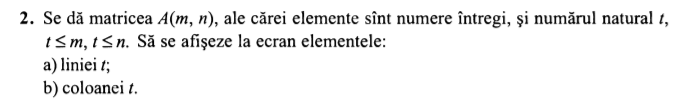
}

printf ("\nElementul minim %d se afla pe randul %d, coloana %d\n", min, poz\_row, poz\_col);

}



122//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int t, row, col;

int a[100][100];

printf ("Introduceti numarul de randuri: ");

scanf ("%d", &row);

printf ("Introduceti numarul de coloane: ");

scanf ("%d", &col);

for (int i = 0; i < row; i++)

{

for (int j = 0; j < col; j++)

{

a[i][j] = rand()%100;

}

}

for (int i = 0; i < row; i++)

{

for (int j = 0; j < col; j++)

{

printf ("%d\t", a[i][j]);

}

printf ("\n");

}

printf ("t = ");

scanf("%d", &t);

printf ("\nElementele liniei t = ");

for (int i = t; i < row; i++)

{

for (int j = 0; j < col; j++)

{

printf ("%d\t", a[i][j]);

if (i == t+1)

{

break;

}

}

}

printf ("\nElementele coloanei t = ");

for (int i = 0; i < row; i++)

{

for (int j = t; j < col; j++)

{

printf ("%d\t", a[i][j]);

if (j == t+1)

{

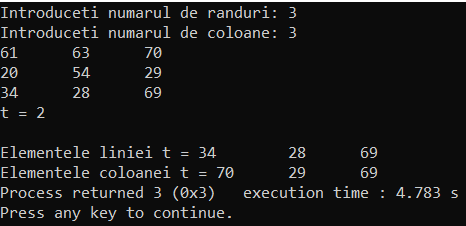
break;

}

}

}

}



**Tema: Șiruri de caractere (p. 80-84) – 30 probleme**

123//



#include <stdio.h>

#include <string.h>

void main ()

{

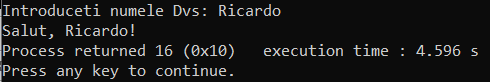
char text[100];

printf ("Introduceti numele Dvs: ");

gets (text);

printf ("Salut, %s! ", text);

}



124//



#include <stdio.h>

#include <string.h>

char \* strup(char \* str)

{

for (int i=0; str[i]; i++)

{

str[i] = toupper(str[i]);

}

}

void main ()

{

char str1[100];

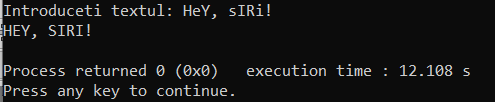
printf ("Introduceti textul: ");

gets(str1);

strupr(str1);

printf ("%s\n", str1);

}



125//



#include <stdio.h>

#include <string.h>

char pozitia\_para (int j, char str[])

{

for (int j=0; str[j]; j++)

{

if (j%2==0)

printf ("Litera de pe pozitia %d este %c\n", j, str[j]);

}

}

char pozitia\_impara (int j, char str[])

{

for (int j=0; str[j]; j++)

{

if (j%2!=0)

printf ("Litera de pe pozitia %d este %c\n", j, str[j]);

}

}

void main ()

{

char str1[100], i;

printf ("Introduceti textul: ");

gets(str1);

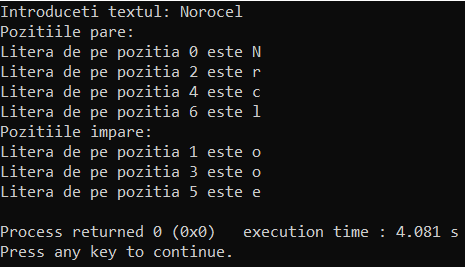
printf("Pozitiile pare: \n");

pozitia\_para(i, str1);

printf("Pozitiile impare: \n");

pozitia\_impara(i, str1);

}



126//



#include <stdio.h>

#include <string.h>

char num(int j, char str[])

{

int count = 0;

for (int j=0; str[j]; j++)

{

if (str[j]=='a'||str[j]=='A')

count++;

}

printf ("Litera 'a' se regaseste de %d ori", count);

}

void main ()

{

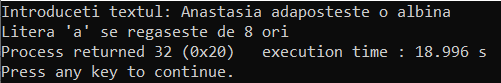
char str1[100], i;

printf ("Introduceti textul: ");

gets(str1);

num(i, str1);

}



127//



#include <stdio.h>

#include <string.h>

char num(int j, char str[])

{

int count = 0;

for (int j=0; str[j]; j++)

{

if (str[j]=='a'||str[j]=='A'||str[j]=='o'||str[j]=='O'||str[j]=='E'||str[j]=='e'||str[j]=='i'||str[j]=='I'||str[j]=='U'||str[j]=='u')

count++;

}

printf ("In text sunt %d vocale", count);

}

void main ()

{

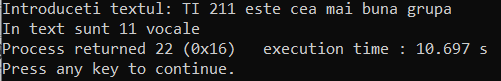
char str1[100], i;

printf ("Introduceti textul: ");

gets(str1);

num(i, str1);

}



128//



#include <stdio.h>

#include <string.h>

#include <ctype.h>

char \*strrev(char \* str)

{

for (int i=0, n = strlen(str)-1; i<n; i++, n--)

{

char temp = str[i];

str[i] = str[n];

str[n] = temp;

}

}

void main ()

{

char str1[100], i;

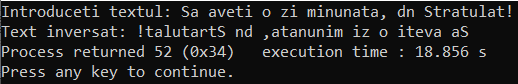
printf ("Introduceti textul: ");

gets(str1);

strrev(str1);

printf ("Text inversat: %s", str1);

}



129//



#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main()

{

char s1[50],s2[50];

printf("Introduceti textul Dvs: ");

gets(s1);

strcpy(s1,s2);

strrev(s2);

if(!strcmp(s1,s2))

{

printf("Textul este palindrom");

}

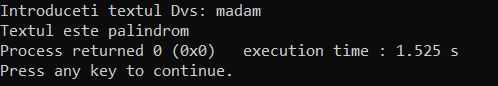
else

{

printf("Textul nu este palindrom");

}

}



130//



#include <stdio.h>

#include <string.h>

char num(int j, char str[])

{

int count = 0;

for (int j=0; str[j]; j++)

{

if (str[j]=='a'||str[j]=='A')

count++;

}

printf ("Silaba 'oa' se regaseste de %d ori", count);

}

void main ()

{

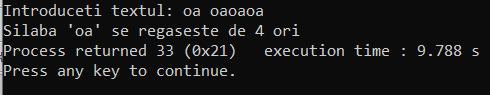
char str1[100], i;

printf ("Introduceti textul: ");

gets(str1);

num(i, str1);

}



131/



#include <stdio.h>

#include <string.h>

char num(int j, char str[])

{

for (int j=0; str[j]; j++)

{

if (str[j]=='o')

{

str[j]='u';

}

else if (str[j]=='O')

{

str[j]='U';

}

}

printf ("Textul modificat: %s", str);

}

void main ()

{

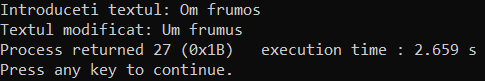
char str1[100], i;

printf ("Introduceti textul: ");

gets(str1);

num(i, str1);

}



132//



#include <stdio.h>

#include <string.h>

char num(int j, char str[])

{

for (int j=0; str[j]; j++)

{

if (str[j]=='o')

{

str[j]='a';

}

else if (str[j]=='a')

{

str[j]='o';

}

}

printf ("Textul modificat: %s", str);

}

void main ()

{

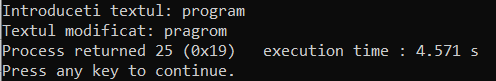
char str1[100], i;

printf ("Introduceti textul: ");

gets(str1);

num(i, str1);

}



133//



#include <stdio.h>

#include <string.h>

void silaba(char text[], char text\_final[])

{

int j=0;

for (int i=0; text[i]!='\0'; i++)

{

text\_final[j]=text[i];

if (text\_final[j]=='a')

{

text\_final[j]='o';

j++;

text\_final[j]='a';

}

j++;

}

text\_final[j]='\0';

puts(text\_final);

}

int main ()

{

char text[100];

char text\_fin[100];

printf ("Introduceti textul: ");

gets(text);

silaba(text, text\_fin);

}



134//



#include <stdio.h>

#include <string.h>

void silaba(char text[], char text\_final[])

{

int j=0;

for (int i=0; i<strlen(text); i++)

{

if (text[i]=='a')

{

text\_final[j]='c';

j++;

text\_final[j]='u';

j++;

}

else if (text[i]=='c'&& text[i+1]=='u')

{

text[i]='a';

text\_final[j]=text[i];

i++;

j++;

}

else

{

text\_final[j]=text[i];

j++;

}

}

text\_final[j]='\0';

puts(text\_final);

}

int main ()

{

char text[100];

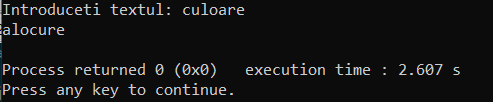
char text\_fin[100];

printf ("Introduceti textul: ");

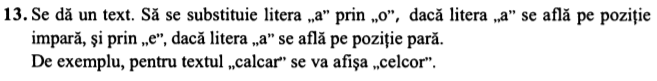
gets(text);

silaba(text, text\_fin);

}



135//



#include <stdio.h>

#include <string.h>

char num(int j, char str[])

{

for (int j=0; str[j]; j++)

{

if (str[j]=='a'&& j%2==0)

{

str[j]='e';

}

else if(str[j]=='a'&& j%2!=0)

{

str[j]='o';

}

}

printf ("Textul modificat: %s", str);

}

void main ()

{

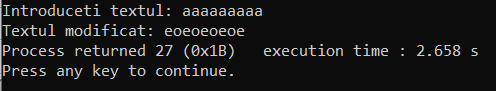
char str1[100], i;

printf ("Introduceti textul: ");

gets(str1);

num(i, str1);

}



136//



#include <stdio.h>

#include <string.h>

void silaba(char text[], char text\_final[])

{

int j=0;

for (int i=0; i<strlen(text); i++)

{

if (text[i]=='c'&& text[i+1]=='s')

{

text[i]='x';

text\_final[j]=text[i];

i++;

j++;

}

else

{

text\_final[j]=text[i];

j++;

}

}

text\_final[j]='\0';

puts(text\_final);

}

int main ()

{

char text[100];

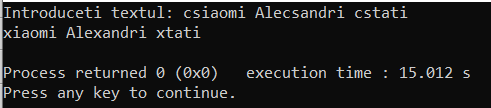
char text\_fin[100];

printf ("Introduceti textul: ");

gets(text);

silaba(text, text\_fin);

}



137//



#include <stdio.h>

#include <string.h>

void silaba(char text[], char text\_final[])

{

int j=0;

for (int i=0; i<strlen(text); i++)

{

if (text[i]=='a')

{

printf(" ");

}

printf ("%c", text[i]);

if (text[i]=='a')

{

printf(" ");

}

}

text\_final[j]='\0';

puts(text\_final);

}

int main ()

{

char text[100];

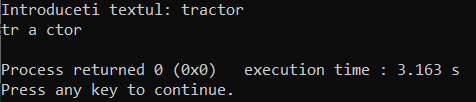
char text\_fin[100];

printf ("Introduceti textul: ");

gets(text);

silaba(text, text\_fin);

}



138//



#include <stdio.h>

#include <string.h>

void silaba(char text[], char text\_final[])

{

int j=0;

for (int i=0; i<strlen(text); i++)

{

text\_final[j]=text[i];

j++;

text\_final[j]=text[i];

j++;

}

text\_final[j]='\0';

printf ("Textul modificat: ");

puts(text\_final);

}

int main ()

{

char text[100];

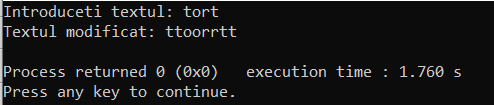
char text\_fin[100];

printf ("Introduceti textul: ");

gets(text);

silaba(text, text\_fin);

}



139//



#include <stdio.h>

#include <string.h>

void silaba(char text[], char text\_final[])

{

int j=0;

for (int i=0; i<strlen(text); i++)

{

if (text[i]=='a')

{

printf ("Prima se intalneste litera 'a'.");

break;

}

else if (text[i]=='o')

{

printf ("Prima se intalneste litera 'o'.");

break;

}

}

}

int main ()

{

char text[100];

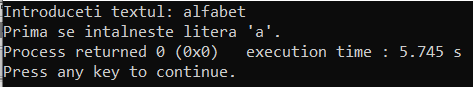
char text\_fin[100];

printf ("Introduceti textul: ");

gets(text);

silaba(text, text\_fin);

}



140//



#include <stdio.h>

#include <string.h>

void silaba(char text[], char litera)

{

for (int i=1; i<=strlen(text); i++)

{

if (text[i]==litera)

{

printf ("Prima pozitie a literei %c in text este %d\n", litera, i);

break;

}

}

printf ("Litera %c nu se regaseste in text\n", litera);

}

int main ()

{

char text[100];

char text\_fin[100];

char lit;

printf ("Introduceti textul: ");

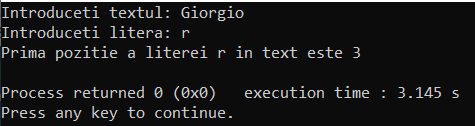
gets(text);

printf ("Introduceti litera: ");

scanf ("%c", &lit);

silaba(text, lit);

}



141//



#include <stdio.h>

#include <string.h>

void main ()

{

char text\_x[100];

char text\_y[100];

printf("Introduceti textul x: ");

gets(text\_x);

printf("Introduceti textul y: ");

gets(text\_y);

str(text\_x, text\_y);

}

void str(char text\_x[], char text\_y[])

{

char \*p = strstr(text\_x, text\_y);

int poz=p-text\_x;

if (strstr(text\_x, text\_y) != NULL)

{

printf ("Textul y se include in textul x incepand cu pozitia %d", poz);

}

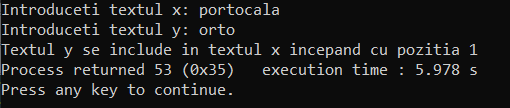
else

{

printf ("Textul y nu se include in textul x");

}

}



142//



#include <stdio.h>

#include <string.h>

void main ()

{

char text\_x[100];

char text\_y[100];

printf("Introduceti textul x: ");

gets(text\_x);

printf("Text 2: ");

gets (text\_y);

str(text\_x, text\_y);

}

void str(char \*text\_x, char \*text\_y)

{

char \*flag = text\_x;

int lenght = strlen(text\_y);

while ((flag = strstr(flag, text\_y)))

{

\*flag = '\0';

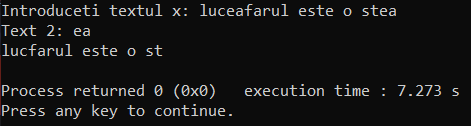
strcat (text\_x, flag+lenght);

flag++;

}

puts (text\_x);

}



143//



#include <stdio.h>

#include <string.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti textul: ");

gets(text);

ins(text, text\_1);

}

void ins (char text[], char text\_1[])

{

int j=0;

for (int i=0; i<strlen(text); i++)

{

text\_1[j] = text[i];

j++;

if (text[i]=='n' && text[i+1]=='a')

{

text\_1[j] = 'o';

j++;

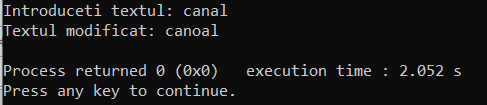
}

}

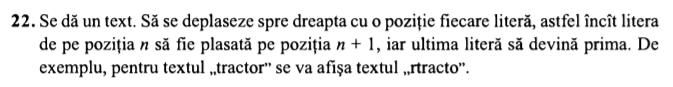
printf ("Textul modificat: ");

puts(text\_1);

}



144//



#include <stdio.h>

#include <string.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti textul: ");

gets(text);

ins(text, text\_1);

}

void ins (char text[], char text\_1[])

{

int size = strlen(text);

char temp = text[size-1];

for (int i=size-1; i>=0; i--)

{

text[i]=text[i-1];

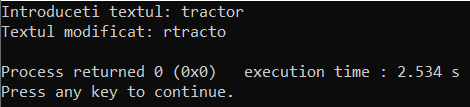
}

text[0]=temp;

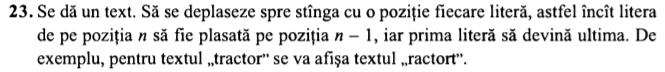
printf ("Textul modificat: ");

puts(text);

}



145//



#include <stdio.h>

#include <string.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti textul: ");

gets(text);

ins(text, text\_1);

}

void ins (char text[], char text\_1[])

{

int size = strlen(text);

char temp = text[0];

for (int i=0; i<size-1; i++)

{

text[i]=text[i+1];

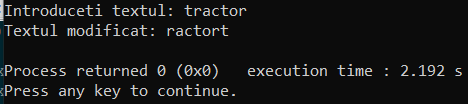
}

text[size-1]=temp;

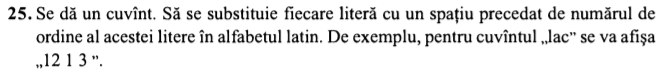
printf ("Textul modificat: ");

puts(text);

}



146//



#include <stdio.h>

#include <string.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti textul: ");

gets(text);

ins(text, text\_1);

}

void ins (char text[], char text\_1[])

{

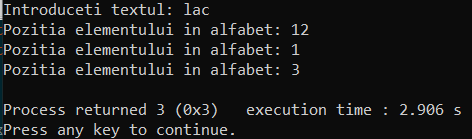
for (int i=0; i<strlen(text); i++)

{

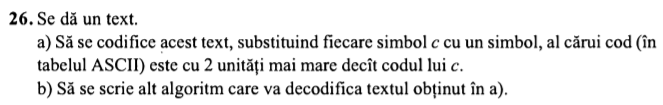
printf ("Pozitia elementului in alfabet: %d\n", text[i]-96);

}

}



147//



#include <stdio.h>

#include <string.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti textul: ");

gets(text);

cod(text, text\_1);

}

void cod (char text[], char text\_1[])

{

printf ("\nTextul codificat: ");

for (int i=0; i<strlen(text); i++)

{

text[i]=text[i]+2;

printf ("%c", text[i]);

}

printf ("\nTextul decodificat: ");

for (int i=0; i<strlen(text); i++)

{

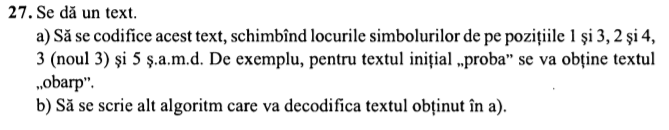
printf ("%c", text[i]-2);

}

}



148//



#include <stdio.h>

#include <string.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti textul: ");

gets(text);

cod(text, text\_1);

}

void cod (char text[], char text\_1[])

{

char temp;

printf ("\nTextul codificat: ");

for (int i=0; i<strlen(text); i++)

{

temp = text[i];

text[i] = text[i+2];

text[i+2] = temp;

if (text[i+3]=='\0')

{

break;

}

}

puts (text);

printf ("\nTextul decodificat: ");

for (int i=strlen(text)-1; i>=0; i--)

{

temp = text[i];

text[i] = text[i-2];

text[i-2] = temp;

if (text[i-3]=='\0')

{

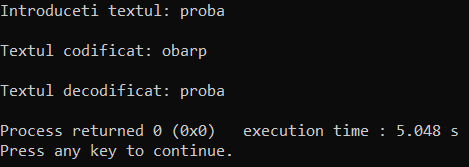
break;

}

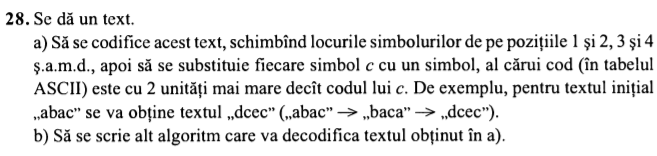
}

puts (text);

}



149//



#include <stdio.h>

#include <string.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti textul: ");

gets(text);

cod(text, text\_1);

}

void cod (char text[], char text\_1[])

{

char temp;

printf ("\nTextul codificat: ");

for (int i=0; i<strlen(text); i++)

{

temp = text[i+1]+2;

text[i+1] = text[i]+2;

text[i] = temp;

i++;

if (text[i+2]=='\0')

{

break;

}

}

puts (text);

printf ("\nTextul decodificat: ");

for (int i=0; i<strlen(text); i++)

{

temp = text[i]-2;

text[i] = text[i+1]-2;

text[i+1] = temp;

i++;

if (text[i+2]=='\0')

{

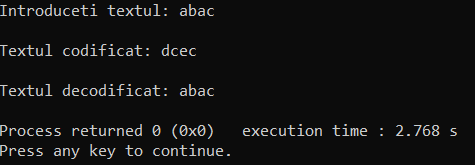
break;

}

}

puts (text);

}



150//



#include <stdio.h>

#include <string.h>

#include <ctype.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti textul: ");

gets(text);

str(text, text\_1);

printf ("Text modificat: ");

puts(text\_1);

}

void str(char text[], char text\_1[])

{

int temp;

int count=0, z=0;

for (int i = 0; i<strlen(text); i++)

{

temp = i;

if (text[i]=='(')

{

for (int j = temp; j < strlen(text); j++)

{

count++;

if (text[j]==')')

{

break;

}

}

i = i + count;

}

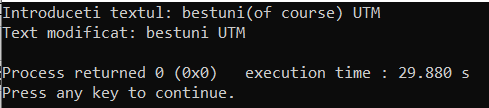
text\_1[z] = text[i];

z++;

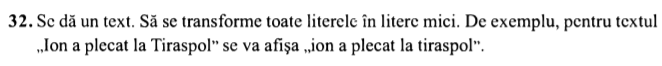
text\_1[z] = '\0';

}

}



151//



#include <stdio.h>

#include <string.h>

#include <ctype.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti textul: ");

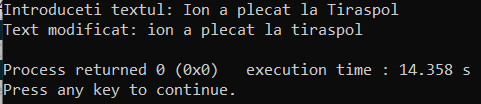
gets(text);

strlwr(text);

printf ("Text modificat: ");

puts(text);

}



152//



#include <stdio.h>

#include <string.h>

#include <ctype.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti textul: ");

gets(text);

str(text);

printf ("Text modificat: ");

puts(text);

}

void str(char text[], char text\_1[])

{

for (int i = 0; i<strlen(text); i++)

{

if (text[i]>=65 && text[i]<=90)

{

text[i] = text[i] + 32;

}

else if (text[i]>=97 && text[i]<=122)

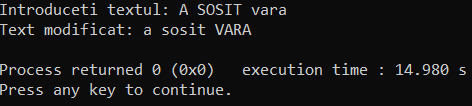
{

text[i] = text[i] - 32;

}

}

}



**Tema: Numere aleatoare (p. 87-89) – 31 probleme**

153//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

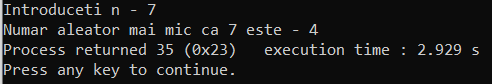
int n;

printf ("Introduceti n - ");

scanf ("%d", &n);

printf ("Numar aleator mai mic ca %d este - %d", n, rand()%n);

}



154//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int n, m;

printf ("Introduceti n - ");

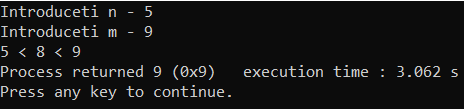
scanf ("%d", &n);

printf ("Introduceti m - ");

scanf ("%d", &m);

printf ("%d < %d < %d", n, n+1+rand()%(m-n-1), m );

}



155//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

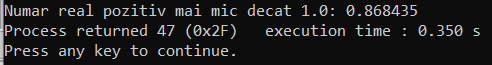
double low = 0;

double up = 1;

double rand\_num = ((double)rand()/(double)RAND\_MAX)\*(up-low)+low;

printf ("Numar real pozitiv mai mic decat %.1lf: %lf ", up, rand\_num);

}



156//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

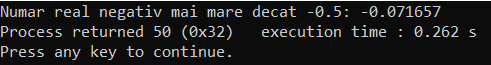
double low = -0.5;

double up = 0;

double rand\_num = ((double)rand()/(double)RAND\_MAX)\*(up-low)+low;

printf ("Numar real negativ mai mare decat %.1lf: %lf ", low, rand\_num);

}



157//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

double low = -2;

double up = 0;

for(int i=0; i<3; i++)

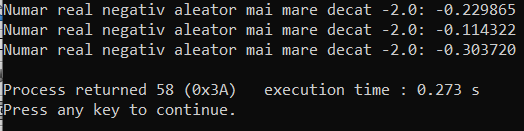
{

double rand\_num = ((double)rand()/(double)RAND\_MAX)\*(up-low)+low;

printf ("Numar real negativ aleator mai mare decat -2.0: %lf\n", rand\_num);

}

}



158//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

double low = 4;

double up = 2;

for(int i=0; i<3; i++)

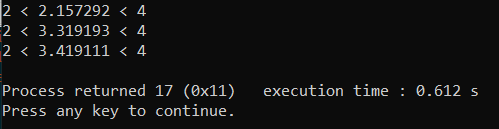
{

double rand\_num = ((double)rand()/(double)RAND\_MAX)\*(up-low)+low;

printf ("%.0lf < %lf < %.0lf\n", up, rand\_num, low);

}

}



159//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

double low = 2;

double up = -3;

for(int i=0; i<5; i++)

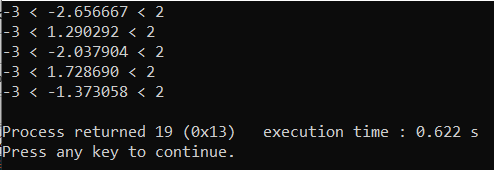
{

double rand\_num = ((double)rand()/(double)RAND\_MAX)\*(up-low)+low;

printf ("%.0lf < %lf < %.0lf\n", up, rand\_num, low);

}

}



160//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int low = 97;

int up = 122;

int rand\_chr = (rand()%(up-low))+low;

printf ("O litera mica a alfabetului latin: %c \n", rand\_chr);

}



161//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

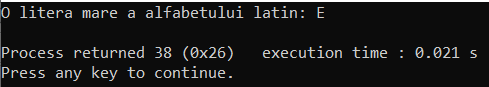
int low = 65;

int up = 90;

int rand\_chr = (rand()%(up-low))+low;

printf ("O litera mare a alfabetului latin: %c \n", rand\_chr);

}



162//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

for (int i=0; i<RAND\_MAX; i++)

{

int num = (rand()%(1000-100))+100;

if (num%6==0 && num%8==0)

{

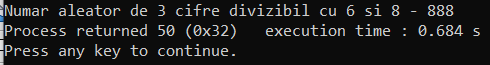
printf ("Numar aleator de 3 cifre divizibil cu 6 si 8 - %d", num);

break;

}

}

}



163//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

for (int i=0; i<RAND\_MAX; i++)

{

int num = rand()%(RAND\_MAX-100)+100;

int prim = num / 2;

int flag = 0;

for (int i = 2; i <= prim; i++)

{

if (num % i == 0)

{

int flag = 1;

break;

}

}

if (flag == 0)

{

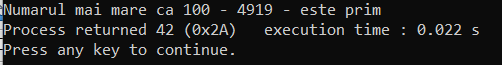
printf ("Numarul mai mare ca 100 - %d - este prim", num);

break;

}

}

}



164//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int array[50];

int n;

int i, j;

printf ("Introduceti n: ");

scanf ("%d", &n);

printf ("Tabloul Dvs este: ");

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

{

array[i] = rand()%100;

if (array[i]==array[i+1])

{

array

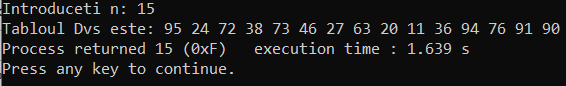
[i+1] = rand()%100;

}

printf ("%d ", array[i]);

}

}



165//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int array[15][15];

int n, m;

int i, j;

printf ("Introduceti nr de randuri: ");

scanf ("%d", &n);

printf ("Introduceti nr de coloane: ");

scanf ("%d", &m);

printf ("Tabloul Dvs este: \n ");

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

{

for (j=0; j < m; j++)

{

array[i][j] = rand()%(300-i-j);

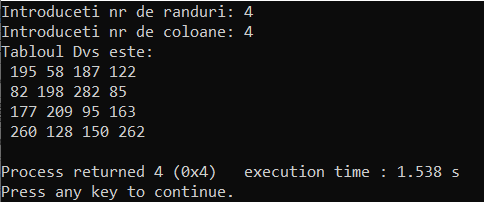
printf ("%d ", array[i][j]);

}

printf ("\n ");

}

}



166//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int i, j, k;

int loto[100];

for (int i=1; i<=36; i++)

{

loto[i] = i;

}

printf ("JOC SUPERLOTO 5 DIN 36! \n");

printf ("\n");

for (int i=1; i<=36; i++)

{

loto[i] = i;

for (int j = 1; j<=10; j++)

{

printf ("Varianta %d: ", j);

for (int k = 1; k <= 5; k++)

{

printf ("%d ", rand()%(loto[35]-loto[0])-loto[0]);

}

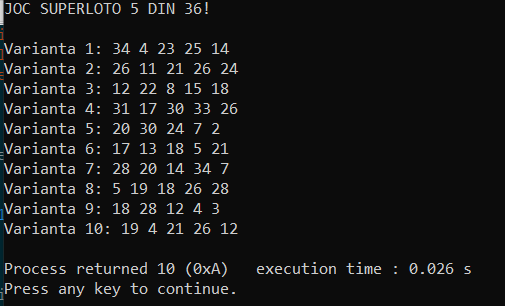
printf ("\n");

}

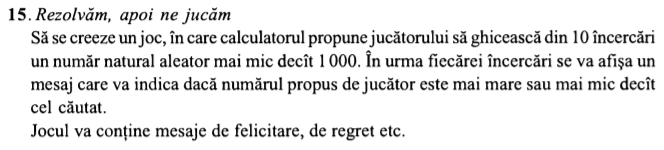
break;

}

}



167//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int random = rand()%1000;

int num;

for (int i = 1; i <= 10; i++)

{

printf ("Introduceti numarul Dvs: ");

scanf ("%d", &num);

if (num < random)

{

printf ("Mai incearca, esti pe aproape! Numarul tau este mai mic decat cel propus\n");

}

else if (num > random)

{

printf ("Stiu ca poti, nu te descuraja! Numarul tau este mai mare decat cel propus\n");

}

if (num == random)

{

printf ("Felicitari, ai castigat!\n");

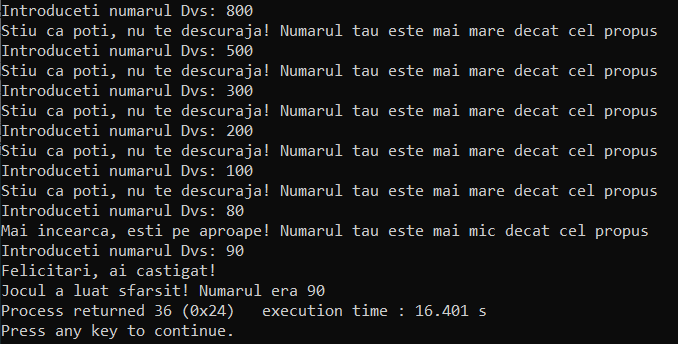
break;

}

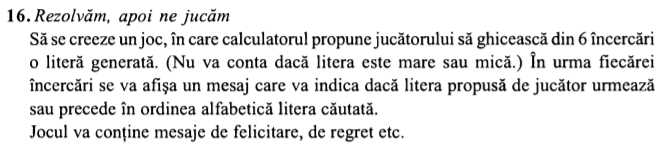
}

printf ("Jocul a luat sfarsit! Numarul era %d", random);

}



168//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int random = rand()%(122-97)+97;

char num;

for (int i = 1; i <= 10; i++)

{

printf ("Introduceti litera Dvs: ");

scanf ("%s", &num);

int check = (int)num;

if (check < random)

{

printf ("Mai incearca, esti pe aproape! Litera ta precede in alfabet litera propusa\n");

}

else if (check > random)

{

printf ("Stiu ca poti, nu te descuraja! Litera ta urmeaza in alfabet litera propusa\n");

}

else if (check == random)

{

printf ("Felicitari, ai castigat!\n");

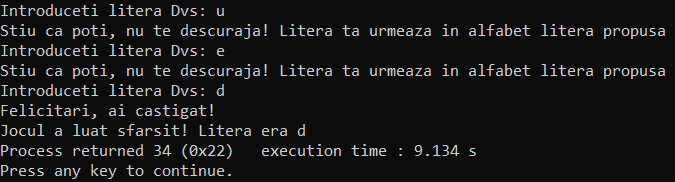
break;

}

}

printf ("Jocul a luat sfarsit! Litera era %c", random);

}



169//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

for(int i=0; i<20; i++)

{

for(int j=0; j<100; j++)

{

int stea = rand()%15;

if(stea==1)

{

printf("\*");

}

else

{

printf(" ");

}

}

printf("\n");

}

}



170//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

char str[] = "program";

int lenght = strlen(str);

int i, j=0;

int count = 0;

while (str[j]!='\0')

{

for (i=0; i<RAND\_MAX; i++)

{

int random = (rand()%(122-97)+97);

if (str[j]!=random)

{

count++;

}

if (str[j]==random)

{

printf ("Pentru generarea literei %c s-a apelat de %d ori la generator \n", str[j], count);

count = 0;

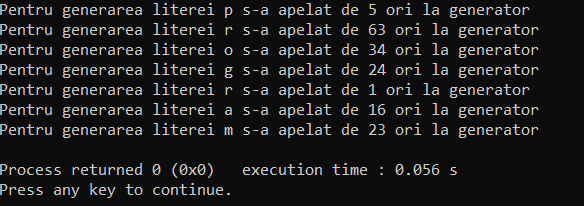
j++;

}

}

}

}



171//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

int main()

{

int i, temp, random;

srand(time(NULL));

char array[26]="abcdefghijklmnopqrstuvwxyz";

int num[26];

for(i=0; i<26; i++)

{

num[i] = i;

}

for(i=0; i<26; i++)

{

temp = num[i];

random = rand()%26;

num[i] = num[random];

num[random] = temp;

}

for (i = 0; i < 26; i++)

{

printf("%3c", array[i]);

}

printf("\n");

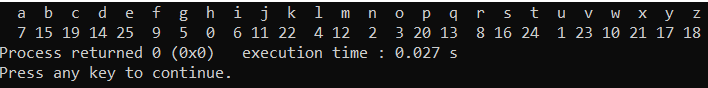
for (i = 0; i < 26; i++)

{

printf("%3d", num[i]);

}

}



172//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int low = 97;

int up = 122;

for (int i = 0; i < 3; i++)

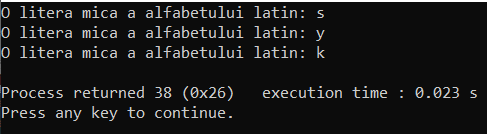
{

int rand\_chr = (rand()%(up-low))+low;

printf ("O litera mica a alfabetului latin: %c \n", rand\_chr);

}

}



173//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int low = 65;

int up = 90;

for (int i = 0; i < 5; i++)

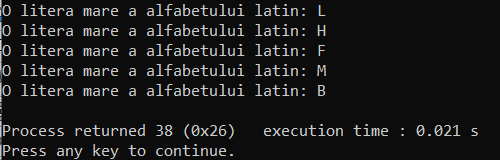
{

int rand\_chr = (rand()%(up-low))+low;

printf ("O litera mare a alfabetului latin: %c \n", rand\_chr);

}

}



174//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int low = 20;

int up = 1000;

int rand\_chr;

int check = rand()%2 + 1;

if (check == 1) // afisam numerele pozitive

{

rand\_chr = rand()%(up-low+1)+low;

}

else

{

rand\_chr = -(rand()%(up-low+1)+low);

}

printf ("Numar aleator, al carui modul este intre 1000 si 20 - %d", rand\_chr);

}



175//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int low = 65;

int up = 90;

int rand\_chr;

int count = 0;

for (int i = 0; i < 10; i++)

{

int check = rand()%2+1;

if (check == 1 && count<3)

{

rand\_chr = rand()%(up-low)+low+32;

count++;

}

else

{

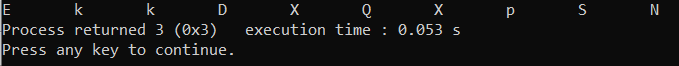
rand\_chr = rand()%(up-low)+low;

}

printf ("%c \t", rand\_chr);

}

}



176//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int low = 65;

int up = 90;

int rand\_chr;

int count = 0;

for (int i = 0; i < 10; i++)

{

int check = rand()%2+1;

if (check == 1)

{

rand\_chr = rand()%(up-low)+low+32;

count++;

}

else

{

rand\_chr = rand()%(up-low)+low;

}

if (count<3 && check == 0)

{

rand\_chr = rand()%(up-low)+low+32;

}

printf ("%c \t", rand\_chr);

}

}



177//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int low = 65;

int up = 90;

int rand\_chr;

int count = 0;

char arr[100];

for (int i = 0; i < 10; i++)

{

int check = rand()%2+1;

if (check == 1 && count<3 )

{

rand\_chr = rand()%(up-low)+low;

count++;

}

else

{

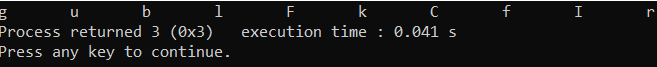
rand\_chr = rand()%(up-low)+low+32;

}

printf ("%c \t", rand\_chr);

}

}



178//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int voc [5] = {97, 101, 105, 111, 117};

int j = rand()%6;

for (int i=0; i<5; i++)

{

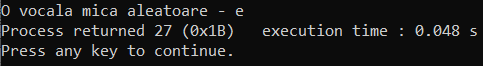
i = j;

printf ("O vocala mica aleatoare - %c", voc[j]);

break;

}

}



179//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int voc [5] = {65, 69, 73, 79, 79};

int j = rand()%6;

for (int i=0; i<5; i++)

{

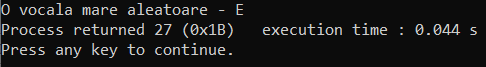
i = j;

printf ("O vocala mare aleatoare - %c", voc[j]);

break;

}

}



180//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int voc [30];

int j = rand()%(122-97)+97;

int i;

for (i = 0; i<RAND\_MAX; i++)

{

if (j !=97 && j != 101 && j != 105 && j != 111 && j != 117)

{

continue;

}

}

printf ("O consoana mica aleatoare - %c", j);

}



181//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int voc [30];

int j = rand()%(90-65)+65;

int i;

for (i = 0; i<RAND\_MAX; i++)

{

if (j !=65 && j != 69 && j != 73 && j != 79 && j != 85)

{

continue;

}

}

printf ("O consoana mare aleatoare - %c", j);

}



182//



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void main()

{

int num, flag, random;

int i, j;

int count = 0;

srand(time(NULL));

for (j=0; j<RAND\_MAX; j++)

{

num = rand()%1000;

if (num < 1000)

{

for(i=1; i<=num; i++)

{

if(num % i==0)

{

flag++;

}

}

if(flag==2)

{

printf("Numar prim mai mic decat 1000 - %d \n", num);

count++;

}

}

flag=0;

if(count==5)

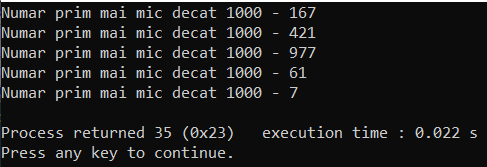
{

break;

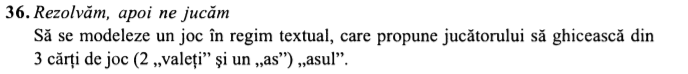
}

}

}



183//



#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <time.h>

int main()

{

srand(time(NULL));

char str[4]="VVV",raspuns[10];

for(int i=0; str[i]; i++)

{

if(rand()%2)

{

str[i]='V';

}

else

{

str[i]='A';

}

}

printf("Ghiceste cartile:\n");

scanf("%3s",raspuns);

if(strcmp(str,raspuns)==0)

{

printf("\nAi castigat!\n");

}

else

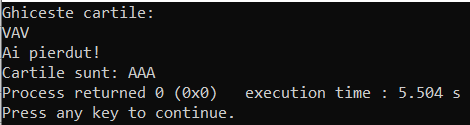
{

printf("Ai pierdut! \nCartile sunt: %s",str);

}

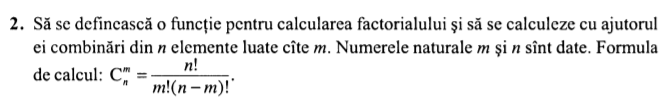
return 0;

}



**Tema: Subprograme (p. 126-130) – 16 probleme**

184//



#include <stdio.h>

int main ()

{

int n, m;

double comb;

printf ("m = ");

scanf ("%d", &m);

printf ("n = ");

scanf ("%d", &n);

if (n<m)

{

printf ("Ati introdus datele gresite!");

}

else

{

comb = (factorial(n))/(factorial(m)\*factorial(n-m));

printf ("Combinari a %d luate cate %d = %.2lf", n, m, comb);

}

}

int factorial (int n)

{

if (n==0)

{

return 1;

}

else

{

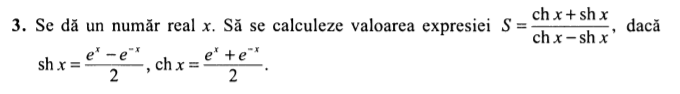
return factorial(n-1)\*n;

}

}



185//



#include <stdio.h>

#include <math.h>

void main ()

{

double x;

printf("x = ");

scanf("%lf",&x);

expresie(x);

}

void expresie(int x)

{

double const e = 2.71828;

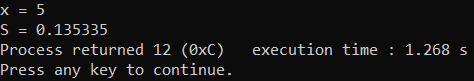
double sh = (pow(e,x)-pow(e,-x))/2;

double ch = (pow(e,x)+pow(e,-x))/2;

double exp = (ch+sh)/(ch-sh);

printf("S = %lf", exp);

}



186//



#include <stdio.h>

void main()

{

int a, b, c;

printf("a = ");

scanf ("%d", &a);

printf("b = ");

scanf ("%d", &b);

printf("c = ");

scanf ("%d", &c);

div(a, b, c);

}

void div(int a, int b, int c)

{

for (int i = 1000; i > 1; i--)

{

if((a%i == 0) && (b%i == 0) && (c%i == 0))

{

printf ("Cel mai mare divizor al lui %d, %d, si %d este %d", a, b, c, i);

break;

}

}

}



187//



#include <stdio.h>

#include <math.h>

void main ()

{

double sum=0;

for (int i=0; i<10; i++)

{

if (i%2==0)

{

sum = sum + put(0.5, i);

}

}

printf ("S = %lf", sum);

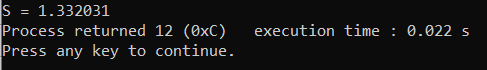
}

int put (int a, int b)

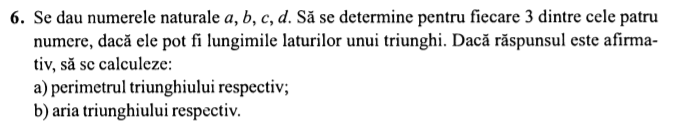
{

return pow(a, b);

}



188//



#include <stdio.h>

#include <math.h>

void main ()

{

int a, b, c, d;

printf ("a = ");

scanf ("%d", &a);

printf ("b = ");

scanf ("%d", &b);

printf ("c = ");

scanf ("%d", &c);

printf ("d = ");

scanf ("%d", &d);

printf("\n");

if (triunghi(a, b, c)==0)

{

printf ("Laturile a, b, c nu pot forma un triunghi.\n");

}

else

{

printf ("Laturile a, b, c pot forma un triunghi.\n");

calcul(a, b, c);

}

if (triunghi(b, c, d)==0)

{

printf ("Laturile b, c, d nu pot forma un triunghi.\n");

}

else

{

printf ("Laturile b, c, d pot forma un triunghi.\n");

calcul(b, c, d);

}

if (triunghi(a, c, d)==0)

{

printf ("Laturile a, c, d nu pot forma un triunghi.\n");

}

else

{

printf ("Laturile a, c, d pot forma un triunghi.\n");

calcul(a, c, d);

}

if (triunghi(a, b, d)==0)

{

printf ("Laturile a, b, d nu pot forma un triunghi.\n");

}

else

{

printf ("Laturile a, b, d pot forma un triunghi.\n");

calcul(a, b, d);

}

}

int triunghi (int a, int b, int c)

{

if (a+b>c || a+c> b || b+c>a)

{

//printf ("Laturile a, b, c pot forma un triunghi.\n");

return 1;

}

}

int calcul (int a, int b, int c)

{

double aria, per, p;

per = a+b+c;

printf ("Perimetrul = %.2lf\n", per);

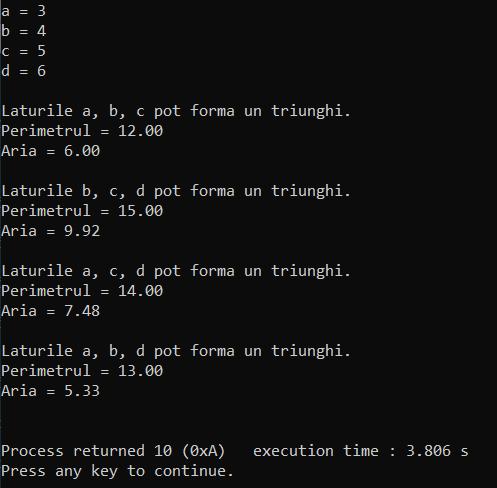
p = per/2;

aria = sqrt(p\*(p-a)\*(p-b)\*(p-c));

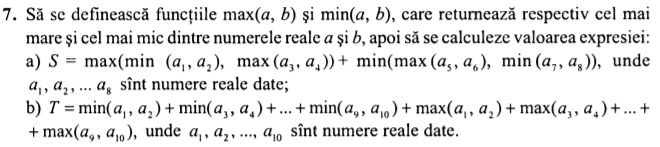
printf ("Aria = %.2lf\n", aria);

printf("\n");

}



189//



#include <stdio.h>

#include <math.h>

#include <time.h>

void main ()

{

srand(time(NULL));

double a[8];

int i;

double sum=0;

printf ("Numerele: ");

for (i = 0; i < 8; i++)

{

a[i]=rand()%50;

printf ("%.lf ", a[i]);

}

printf ("\n");

sum = max(min(a[0],a[1]), max(a[2],a[3]))+ min(max(a[4],a[5]), min(a[6],a[7]));

printf ("S = %.2lf", sum);

}

int max(double x, double y)

{

if (x>y)

{

return x;

}

else

{

return y;

}

}

int min(double x, double y)

{

if (x<y)

{

return x;

}

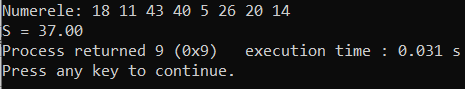
else

{

return y;

}

}



#include <stdio.h>

#include <math.h>

#include <time.h>

void main ()

{

srand(time(NULL));

double a[8];

int i, j;

double sum=0;

printf ("Numerele: ");

for (i = 0; i < 8; i++)

{

a[i]=rand()%50;

printf ("%.lf ", a[i]);

}

printf ("\n");

for (i = 0; i < 8; i++)

{

if (i%2 == 0)

{

sum += min(a[i], a[i+1]) + max(a[i], a[i+1]);

}

}

printf ("T = %.2lf", sum);

}

int max(double x, double y)

{

if (x>y)

{

return x;

}

else

{

return y;

}

}

int min(double x, double y)

{

if (x<y)

{

return x;

}

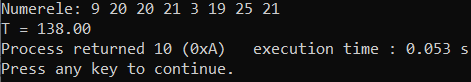
else

{

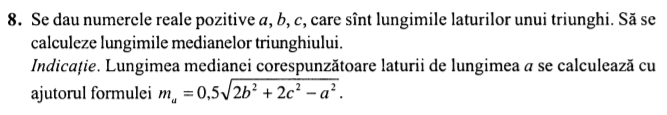
return y;

}

}



190//



#include <stdio.h>

#include <math.h>

#include <time.h>

void main ()

{

srand(time(NULL));

double a = rand()%20;

double b = rand()%20;

double c = rand()%20;

printf ("a = %.lf\n", a);

printf ("b = %.lf\n", b);

printf ("c = %.lf\n", c);

med(a, b, c);

med (b, a, c);

med (c, a, b);

}

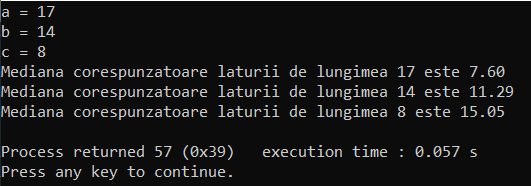
int med(double a, double b, double c)

{

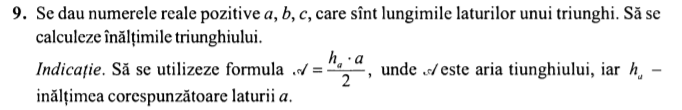
double media = 0.5\*sqrt(2\*pow(b,2)+ 2\*pow(c, 2)-pow(a,2));

printf ("Mediana corespunzatoare laturii de lungimea %.lf este %.2lf\n", a, media);

}



191//



#include <stdio.h>

#include <math.h>

#include <time.h>

void main ()

{

srand(time(NULL));

double a = rand()%20;

double b = rand()%20;

double c = rand()%20;

printf ("a = %.lf\n", a);

printf ("b = %.lf\n", b);

printf ("c = %.lf\n", c);

med(a, b, c);

med(b, a, c);

med(c, a, b);

}

int med(double a, double b, double c)

{

double p = (a+b+c)/2;

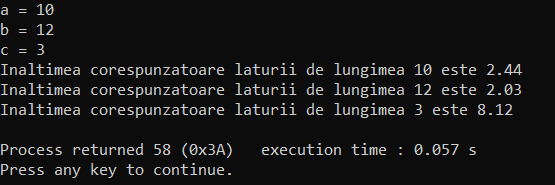
double aria = sqrt(p\*(p-a)\*(p-b)\*(p-c));

double h;

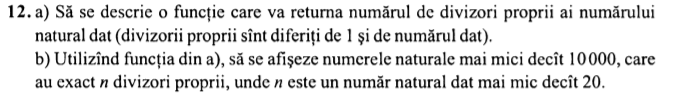
h = (2\*aria)/a;

printf ("Inaltimea corespunzatoare laturii de lungimea %.lf este %.2lf\n", a, h);

}



192//



#include <stdio.h>

void main ()

{

int num;

printf("Introduceti numarul: ");

scanf ("%d", &num);

div(num);

}

void div (int num)

{

int count = 0;

for (int i = 2; i < num; i++)

{

if (num%i == 0)

{

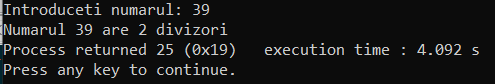
count++;

}

}

printf ("Numarul %d are %d divizori", num, count);

}



#include <stdio.h>

#include <time.h>

int main ()

{

srand(time(NULL));

int n = 10;

int check;

int arr[10000];

printf ("%d\n", n);

for (int k = 0; k < 10; k++)

{

arr[k] = div(k);

printf ("%d ", arr[k]);

}

for (int j = 0; j < 10000; j++)

{

if (arr[j] == n);

{

printf ("%d are exact %d divizori proprii\n", j, check);

}

}

}

void div (int num)

{

int count = 0;

for (int i = 2; i < num; i++)

{

if (num%i == 0)

{

count++;

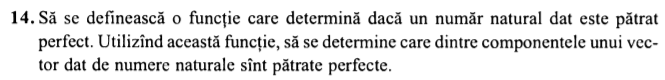
}

}

printf ("%d", count);

}

193//



#include <stdio.h>

#include <math.h>

#include <time.h>

int main ()

{

int a[100];

for (int i=0; i<=100; i++)

{

a[i]=i;

perf(a[i], i);

}

}

int perf(int n, int i)

{

i = sqrt(n);

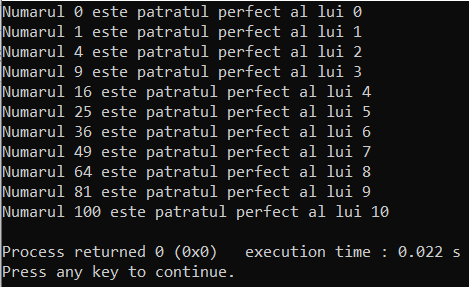
if (i\*i == n)

{

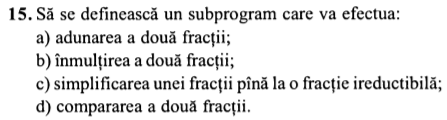
printf ("Numarul %d este patratul perfect al lui %d\n", n, i);

}

}



194//



#include <stdio.h>

#include <math.h>

#include <time.h>

int main ()

{

int a, b, c, d, x, y;

printf ("Introduceti prima fractie: ");

scanf("%d/%d", &a, &b);

printf ("Introduceti a doua fractie: ");

scanf("%d/%d", &c, &d);

fr(a, b, c, d);

}

int fr(int a, int b, int c, int d)

{

int x, y;

if (b==d)

{

x = a + c;

y = b;

}

else

{

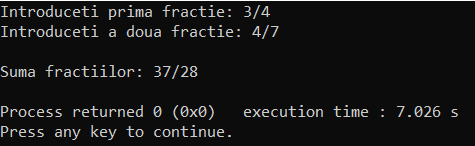
x = (a \* d) + (b \* c);

y = b \* d;

}

printf ("\nSuma fractiilor: %d/%d\n", x, y);

}



#include <stdio.h>

#include <math.h>

#include <time.h>

int main ()

{

int a, b, c, d, x, y;

printf ("Introduceti prima fractie: ");

scanf("%d/%d", &a, &b);

printf ("Introduceti a doua fractie: ");

scanf("%d/%d", &c, &d);

fr(a, b, c, d);

}

int fr(int a, int b, int c, int d)

{

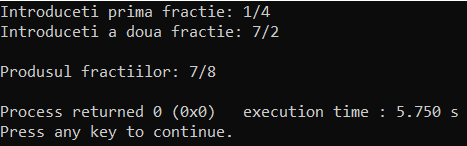
int x, y;

x = a \* c;

y = b \* d;

printf ("\nProdusul fractiilor: %d/%d\n", x, y);

}



#include <stdio.h>

#include <math.h>

#include <time.h>

int main ()

{

int a, b, c, d, x, y;

printf ("Introduceti prima fractie: ");

scanf("%d/%d", &a, &b);

printf ("Introduceti a doua fractie: ");

scanf("%d/%d", &c, &d);

fr(a, b, c, d);

}

int fr(int a, int b, int c, int d)

{

int x, y, z;

if (b==d)

{

if (a < c)

{

printf ("%d/%d < %d/%d", a, b, c, d);

}

else

{

printf ("%d/%d > %d/%d", a, b, c, d);

}

}

else

{

x = a \* d;

z = c \* b;

y = b \* d;

if (x < z)

{

printf ("%d/%d < %d/%d", a, b, c, d);

}

else

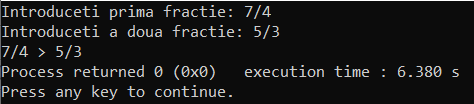
{

printf ("%d/%d > %d/%d", a, b, c, d);

}

}

}



#include <stdio.h>

#include <math.h>

#include <time.h>

int main ()

{

int a, b, c, d, x, y;

printf ("Introduceti fractia: ");

scanf("%d/%d", &a, &b);

fr(a, b);

}

void fr(int a, int b)

{

for (int i=0; i<=100; i++)

{

for (int j = 1; j<=100; j++)

{

if (a%j==0 && b%j==0)

{

a = a/j;

b = b/j;

}

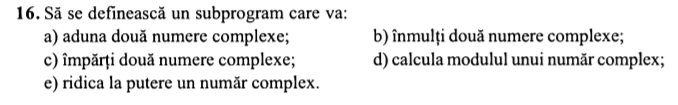
}

}

printf ("Fractia redusa - %d/%d", a, b);

}

195//



#include <stdio.h>

#include <math.h>

#include <time.h>

int main ()

{

double a1, a2, b1, b2;

int n;

printf ("R(z1) = ");

scanf ("%lf", &a1);

printf ("Im(z1) = ");

scanf ("%lf", &b1);

printf ("R(z2) = ");

scanf ("%lf", &a2);

printf ("Im(z1) = ");

scanf ("%lf", &b2);

printf ("Introduceti numarul la ce putere doriti sa fie ridicat numarul complex: ");

scanf ("%d", &n);

complex (a1, b1, a2, b2, n);

}

int complex(double a1, double b1, double a2, double b2, int n)

{

double r = sqrt(pow(a1, 2)+pow(b1, 2));

double cos\_1 = cos(n\*a1/sqrt(pow(a1, 2)+pow(b1, 2)));

double sin\_1 = sin(n\*b1/sqrt(pow(a1, 2)+pow(b1, 2)));

printf ("z1 + z2 = %.2lf + %.2lfi\n", a1+a2, b1+b2);

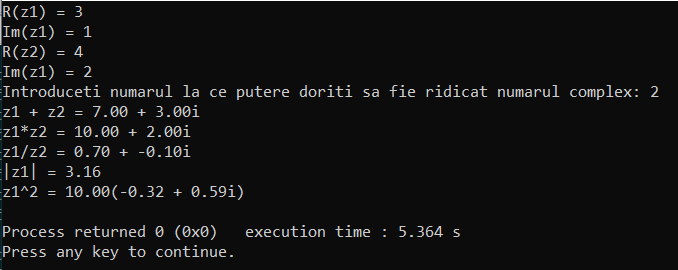
printf ("z1\*z2 = %.2lf + %.2lfi\n", (a1\*a2-b1\*b2), (a1\*b2-b1\*a2));

printf ("z1/z2 = %.2lf + %.2lfi\n", ((a1\*a2 + b1\*b2)/(pow(a2, 2)+pow(b2, 2))), ((a2\*b1 - a1\*b2)/(pow(a2, 2)+pow(b2, 2))));

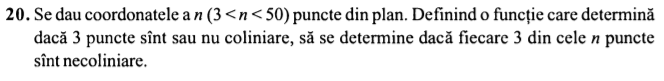
printf ("|z1| = %.2lf\n", r);

printf ("z1^%d = %.2lf(%.2lf + %.2lfi)\n", n, pow(r, n), cos\_1, sin\_1);

}



196//



#include <stdio.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int x1 = rand()%5;

int y1 = rand()%5;

int x2 = rand()%5;

int y2 = rand()%5;

int x3 = rand()%5;

int y3 = rand()%5;

printf ("Punctele sunt: \n");

printf ("A(%d, %d)\n", x1, y1);

printf ("B(%d, %d)\n", x2, y2);

printf ("C(%d, %d)\n", x3, y3);

col(x1, x2, x3, y1, y2, y3);

}

void col (int x1, int x2, int x3, int y1, int y2, int y3)

{

double k1 = (y3 - y1)/(y2 - y1);

double k2 = (x3 - x1)/(x2 - x1);

if (k1 == k2)

{

printf ("Punctele sunt coliniare\n");

}

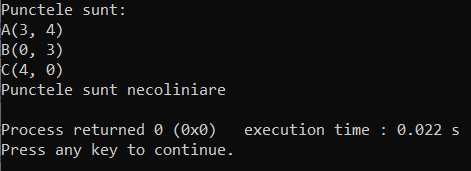
else

{

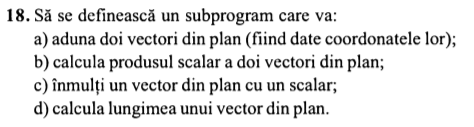
printf ("Punctele sunt necoliniare\n");

}

}



197//



#include <stdio.h>

#include <time.h>

void main ()

{

srand(time(NULL));

int x1 = rand()%10;

int y1 = rand()%10;

int z1 = rand()%10;

int x2 = rand()%10;

int y2 = rand()%10;

int z2 = rand()%10;

int scalar;

printf ("Scalarul - ");

scanf ("%d", &scalar);

printf ("Punctele sunt: \n");

printf ("a = {%d, %d, %d}\n", x1, y1, z1);

printf ("b = {%d, %d, %d}\n", x2, y2, z2);

vector(x1, y1, z1, x2, y2, z2, scalar);

}

void vector(int x1, int y1, int z1, int x2, int y2, int z2, int scalar)

{

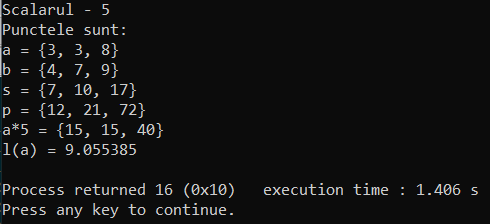
printf ("s = {%d, %d, %d}\n", x1+x2, y1+y2, z1+z2);

printf ("p = {%d, %d, %d}\n", x1\*x2, y1\*y2, z1\*z2);

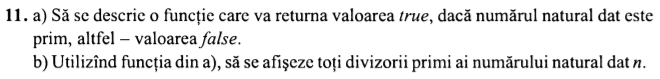
printf ("a\*%d = {%d, %d, %d}\n", scalar, x1\*scalar, y1\*scalar, z1\*scalar);

printf ("l(a) = %lf\n", sqrt(pow(x1, 2)+ pow(y1, 2) + pow(z1, 2)));

}



198//



#include <stdio.h>

void main()

{

int num;

printf("Introduceti numarul: ");

scanf("%d",&num);

prim(num);

}

void prim(int num)

{

int check = 0;

for(int i = 2; i < num; i++)

{

for(int j = 1; j <= i; j++)

{

if(i%j==0)

{

check++;

}

}

if ((check == 2) && (num%i == 0))

{

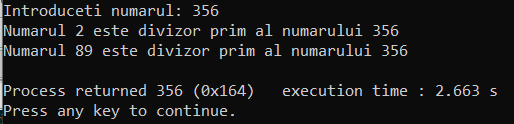
printf ("Numarul %d este divizor prim al numarului %d\n", i, num);

}

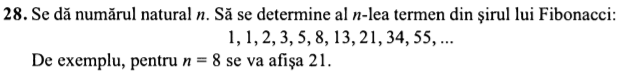
check = 0;

}

}



199//



#include <stdio.h>

#include <time.h>

void main ()

{

int a, b, c, n;

printf ("n = ");

scanf ("%d", &n);

fib(a, b, c, n);

}

void fib(int a, int b, int c, int n)

{

a = 0;

b = 1;

for (int i = 2; i <= n; i++)

{

c = a+b;

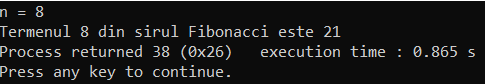
a = b;

b = c;

}

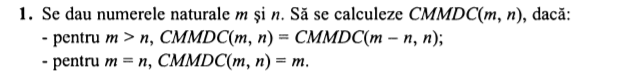
printf ("Termenul %d din sirul Fibonacci este %d", n, c);

}



**Tema: Subprograme recursive (p 137-140) – 8 probleme**

200//



#include <stdio.h>

int main ()

{

int m, n;

printf ("m = ");

scanf ("%d", &m);

printf ("n = ");

scanf ("%d", &n);

printf ("Cel mai mare divizor comun al lui %d si %d - %d", m, n, cmmdc(m, n));

}

int cmmdc(int m, int n)

{

if (n == m)

{

return m;

}

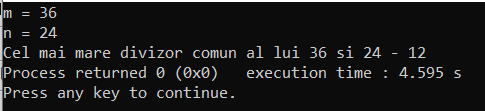
else if (m > n && n > 0)

{

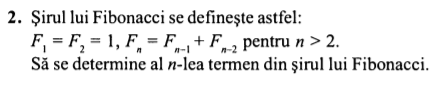
return cmmdc(n, m%n);

}

}



201//



#include <stdio.h>

int main ()

{

int n;

printf ("n = ");

scanf ("%d", &n);

printf ("Termenul al %d-lea din sirul lui Fibonacci - %d", n, fib(n));

}

int fib(int n)

{

int a = 1;

int b = 1;

if (n <= 1)

{

return n;

}

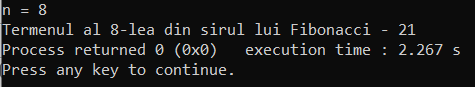
else

{

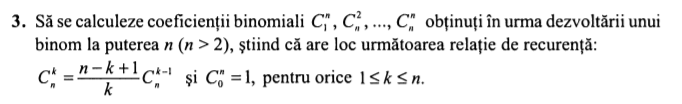
return fib(n-1) + fib (n-2);

}

}



202//



#include <stdio.h>

void main ()

{

int n, k;

printf ("k = ");

scanf ("%d", &k);

printf ("n = ");

scanf ("%d", &n);

printf ("Coeficientul binomial = %d", bin(n, k));

}

int bin(int k, int n)

{

if (k == 0 || n == k)

{

return 1;

}

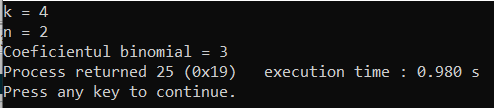
else

{

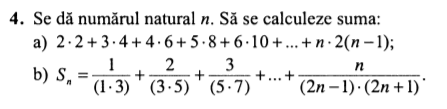
return bin (k-1, n-1)\*((n-k+1)/k);

}

}



203//



#include <stdio.h>

void main ()

{

int n;

printf ("n = ");

scanf ("%d", &n);

printf ("suma = %d", sum(n));

}

int sum(int n)

{

if (n<2)

{

return 0;

}

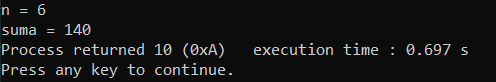
else

{

return sum(n-1)+(n\*2\*(n-1));

}

}



#include <stdio.h>

float exp(float n)

{

if (n==0)

{

return 0;

}

else

{

return ((float)n/((float)(2\*n-1)\*(float)(2\*n+1)))+exp(n-1);

}

}

void main()

{

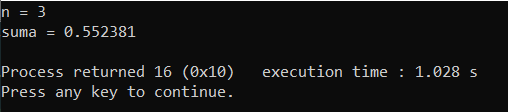
int n;

printf("n = ");

scanf("%d", &n);

printf("suma = %f\n",exp(n));

}



204//



#include <stdio.h>

void main ()

{

int n;

printf ("n = ");

scanf ("%d", &n);

printf ("suma = %d", sum(n));

}

int sum(int n)

{

if (n==0)

{

return 0;

}

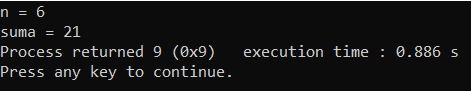
else

{

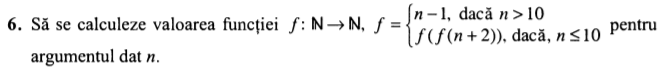
return n + sum(n-1);

}

}



205//



#include <stdio.h>

void main ()

{

int n;

printf ("n = ");

scanf ("%d", &n);

printf ("valoarea functiei pentru n = %d: %d", n, f(n));

}

int f(int n)

{

if (n>10)

{

return n-1;

}

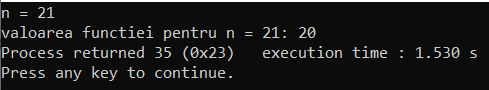
else

{

return f(f(n+2));

}

}



206//



#include <stdio.h>

void main ()

{

int n, inv;

printf ("n = ");

scanf ("%d", &n);

inv = invers(n);

printf ("Inversul lui %d - %d", n, inv);

}

int rest, sum = 0;

int invers(int n)

{

if (n!=0)

{

rest = n%10;

sum = sum\*10 + rest;

return invers(n/10);

}

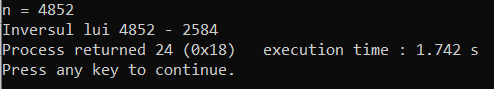
else

{

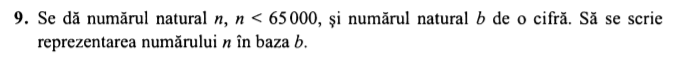
return sum;

}

}



207//



#include <stdio.h>

void main ()

{

int n,b;

printf("n = ");

scanf("%d",&n);

printf("b = ");

scanf("%d",&b);

printf("%d in baza %d = %d", n, b, f(n, b));

}

int f(int n, int b)

{

if (n < b)

{

return n;

}

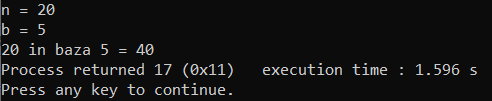
else

{

return 10 \* f(n/b, b) + (n%b);

}

}



**Tema: Tipul de date record (p. 101-103) – 9 probleme**

208//



#include <stdio.h>

#include <stdlib.h>

struct Complex

{

double real1;

double imag1;

double real2;

double imag2;

};

void main ()

{

struct Complex c;

printf ("Introdu partea reala a I-ului nr complex: ");

scanf ("%lf", &c.real1);

printf ("Introdu partea imaginara a I-ului nr complex: ");

scanf ("%lf", &c.imag1);

printf ("Introdu partea reala a celui de-al II-lea nr complex: ");

scanf ("%lf", &c.real2);

printf ("Introdu partea imaginara a celui de-al II-lea nr complex: ");

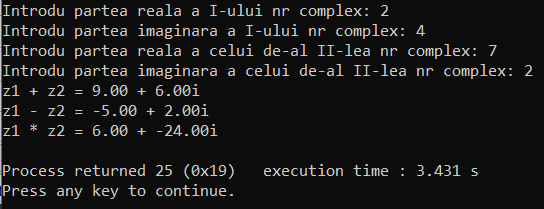
scanf ("%lf", &c.imag2);

printf ("z1 + z2 = %.2lf + %.2lfi\n", c.real1+c.real2, c.imag1+c.imag2);

printf ("z1 - z2 = %.2lf + %.2lfi\n", c.real1-c.real2, c.imag1-c.imag2);

printf ("z1 \* z2 = %.2lf + %.2lfi\n", ((c.real1\*c.real2)-(c.imag1\*c.imag2)), ((c.real1\*c.imag2)-(c.imag1\*c.real2)));

}



209//



#include <stdio.h>

#include <stdlib.h>

struct fractie

{

int numarator1;

int numitor1;

int numarator2;

int numitor2;

};

void operatii (int a, int b, int c, int d)

{

if (b == d)

{

printf ("Suma = %d/%d", a+c, b);

printf ("Diferenta = %d/%d", a-c, b);

printf ("Produs = %d/%d", a\*c, b\*d);

printf ("Impartire = %d/%d", a\*d, b\*c);

}

else

{

printf ("Suma = %d/%d\n", (a\*d)+(c\*d), b\*d);

printf ("Diferenta = %d/%d\n", (a\*d)-(c\*d), b\*d);

printf ("Produs = %d/%d\n", a\*c, b\*d);

printf ("Impartire = %d/%d\n", a\*d, b\*c);

}

}

void main ()

{

struct fractie f;

printf ("Introdu prima fractie: ");

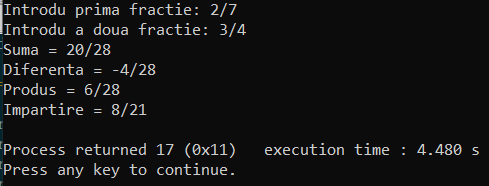
scanf ("%d/%d", &f.numarator1, &f.numitor1);

printf ("Introdu a doua fractie: ");

scanf ("%d/%d", &f.numarator2, &f.numitor2);

operatii(f.numarator1, f.numitor1, f.numarator2, f.numitor2);

}



210//



#include <stdio.h>

#include <stdlib.h>

struct nastere

{

int zi\_S;

int luna\_S;

int an\_S;

int zi\_I;

int luna\_I;

int an\_I;

int zi\_A;

int luna\_A;

int an\_A;

};

int birth (int zi, int luna, int an)

{

zi += (2021-an)\*365 + (12-luna)\*30;

}

void main ()

{

struct nastere n;

printf ("Ziua de nastere a lui Sergiu: ");

scanf ("%d/%d/%d", &n.zi\_S, &n.luna\_S, &n.an\_S);

int n\_S = birth(n.zi\_S, n.luna\_S, n.an\_S);

printf ("Ziua de nastere a lui Ion: ");

scanf ("%d/%d/%d", &n.zi\_I, &n.luna\_I, &n.an\_I);

int n\_I = birth(n.zi\_I, n.luna\_I, n.an\_I);

printf ("Ziua de nastere a lui Andrei: ");

scanf ("%d/%d/%d", &n.zi\_A, &n.luna\_A, &n.an\_A);

int n\_A = birth(n.zi\_A, n.luna\_A, n.an\_A);

if (n\_S > n\_I && n\_S > n\_A)

{

printf ("Cel mai in varsta este Sergiu");

}

else if (n\_A > n\_I && n\_A > n\_S)

{

printf ("Cel mai in varsta este Andrei");

}

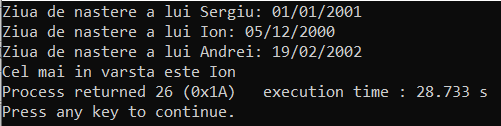
else

{

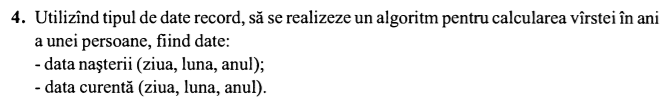
printf ("Cel mai in varsta este Ion");

}

}



211//



#include <stdio.h>

#include <stdlib.h>

struct nastere

{

int zi\_n;

int luna\_n;

int an\_n;

int zi\_c;

int luna\_c;

int an\_c;

int varsta;

};

void main ()

{

struct nastere n;

printf ("Data nasterii persoanei: ");

scanf ("%d/%d/%d", &n.zi\_n, &n.luna\_n, &n.an\_n);

printf ("Data curenta: ");

scanf ("%d/%d/%d", &n.zi\_c, &n.luna\_c, &n.an\_c);

if (n.luna\_c < n.luna\_n)

{

n.varsta = n.an\_c - n.an\_n - 1;

}

else

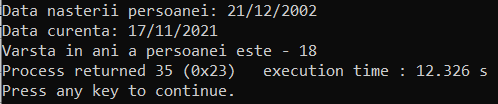
{

n.varsta = n.an\_c - n.an\_n;

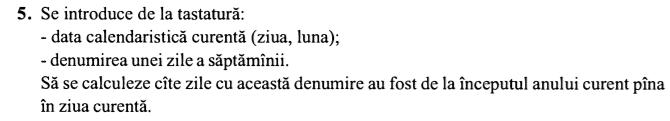
}

printf ("Varsta in ani a persoanei este - %d", n.varsta);

}



212//



#include <stdio.h>

#include <stdlib.h>

struct saptamana

{

int zi\_c;

int luna\_c;

};

void main ()

{

struct saptamana s;

char zi\_sapt[50];

printf ("Data curenta: ");

scanf ("%d/%d", &s.zi\_c, &s.luna\_c); // se considera prima zi a anului - vineri

printf ("Introduceti o zi a saptamanii: ");

scanf ("%s", zi\_sapt);

s.zi\_c += s.luna\_c\*30;

if (s.zi\_c%7 >= 0 && s.zi\_c%7 <= 2)

{

printf ("%s a fost de %d ori\n", zi\_sapt, s.zi\_c/7 - 1);

}

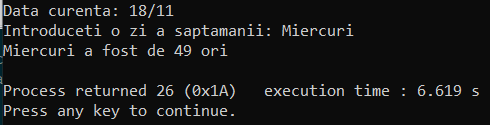
else

{

printf ("%s a fost de %d ori\n", zi\_sapt, s.zi\_c/7);

}

}



213//

  
#include <stdio.h>

#include <stdlib.h>

struct diferenta

{

int zi\_1;

int luna\_1;

int an\_1;

int zi\_2;

int luna\_2;

int an\_2;

};

int zile (int zi, int luna, int an)

{

return luna\*30 + an\*365 + zi;

}

void main ()

{

struct diferenta d;

printf ("Prima data: ");

scanf ("%d/%d/%d", &d.zi\_1, &d.luna\_1, &d.an\_1);

printf ("A doua data: ");

scanf ("%d/%d/%d", &d.zi\_2, &d.luna\_2, &d.an\_2);

int zi\_1, zi\_2;

zi\_1 = zile(d.zi\_1, d.luna\_1, d.an\_1);

zi\_2 = zile(d.zi\_2, d.luna\_2, d.an\_2);

if (zi\_1 < zi\_2)

{

printf ("Diferenta de date = %d zile ", zi\_2-zi\_1);

}

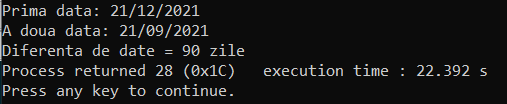
else

{

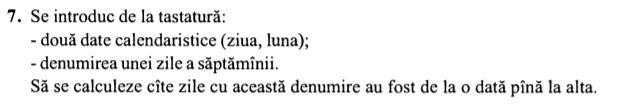
printf ("Diferenta de date = %d zile ", zi\_1-zi\_2);

}

}



214//



#include <stdio.h>

#include <stdlib.h>

struct diferenta

{

int zi\_1;

int luna\_1;

int an\_1;

int zi\_2;

int luna\_2;

int an\_2;

int zi1;

int zi2;

};

int zile (int zi, int luna, int an)

{

return luna\*30 + an\*365 + zi;

}

void main ()

{

struct diferenta d;

char zi\_sapt[50];

printf ("Prima data: ");

scanf ("%d/%d/%d", &d.zi\_1, &d.luna\_1, &d.an\_1);

printf ("A doua data: ");

scanf ("%d/%d/%d", &d.zi\_2, &d.luna\_2, &d.an\_2);

printf ("Introduceti o zi a saptamanii: ");

scanf ("%s", &zi\_sapt);

d.zi1 = zile(d.zi\_1, d.luna\_1, d.an\_1);

d.zi2 = zile(d.zi\_2, d.luna\_2, d.an\_2);

if (d.zi1 > d.zi2)

{

printf ("%s a fost de %d ori", zi\_sapt, (d.zi1 - d.zi2)/7);

}

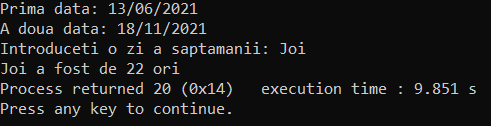
else

{

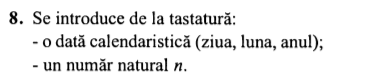
printf ("%s a fost de %d ori", zi\_sapt, (d.zi2 - d.zi1)/7);

}

}



215//





#include <stdio.h>

#include <stdlib.h>

struct diferenta

{

int zi\_1, luna\_1, an\_1;

int zi\_2, luna\_2, an\_2;

int zile\_f, zile\_i;

int n;

};

int zile (int zi, int luna, int an)

{

return luna\*30 + an\*365 + zi;

}

int data (int zi, int luna, int an, int nr)

{

an = nr/365;

luna = (nr - (an\*365))/30;

zi = (nr - (an\*365)-(luna\*30));

}

void main ()

{

struct diferenta d;

int n;

printf ("Prima data: ");

scanf ("%d/%d/%d", &d.zi\_1, &d.luna\_1, &d.an\_1);

printf ("n = ");

scanf ("%d", &n);

d.zile\_i = zile(d.zi\_1, d.luna\_1, d.an\_1);

d.zile\_f = d.zile\_i + n;

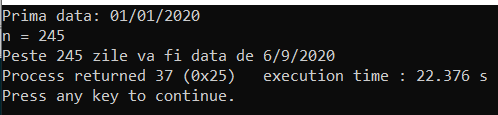
d.an\_2 = d.zile\_f/365;

d.luna\_2 = (d.zile\_f - (d.an\_2\*365))/30;

d.zi\_2 = (d.zile\_f - (d.an\_2\*365)-(d.luna\_2\*30));

printf ("Peste %d zile va fi data de %d/%d/%d", n, d.zi\_2, d.luna\_2, d.an\_2);

}



#include <stdio.h>

#include <stdlib.h>

struct diferenta

{

int zi\_1, luna\_1, an\_1;

int zi\_2, luna\_2, an\_2;

int zile\_f, zile\_i;

int n;

};

int zile (int zi, int luna, int an)

{

return luna\*30 + an\*365 + zi;

}

int data (int zi, int luna, int an, int nr)

{

an = nr/365;

luna = (nr - (an\*365))/30;

zi = (nr - (an\*365)-(luna\*30));

}

void main ()

{

struct diferenta d;

int n;

printf ("Prima data: ");

scanf ("%d/%d/%d", &d.zi\_1, &d.luna\_1, &d.an\_1);

printf ("n = ");

scanf ("%d", &n);

d.zile\_i = zile(d.zi\_1, d.luna\_1, d.an\_1);

d.zile\_f = d.zile\_i - n;

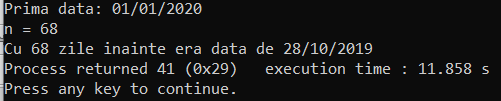
d.an\_2 = d.zile\_f/365;

d.luna\_2 = (d.zile\_f - (d.an\_2\*365))/30;

d.zi\_2 = (d.zile\_f - (d.an\_2\*365)-(d.luna\_2\*30));

printf ("Cu %d zile inainte era data de %d/%d/%d", n, d.zi\_2, d.luna\_2, d.an\_2);

}



216//



#include <stdio.h>

#include <stdlib.h>

struct nastere

{

int zi\_S;

int luna\_S;

int an\_S;

int zi\_I;

int luna\_I;

int an\_I;

int zi\_A;

int luna\_A;

int an\_A;

};

int birth (int zi, int luna, int an)

{

zi += (2021-an)\*365 + (12-luna)\*30;

}

void main ()

{

struct nastere n;

printf ("Ziua de nastere a lui Sergiu: ");

scanf ("%d/%d/%d", &n.zi\_S, &n.luna\_S, &n.an\_S);

int n\_S = birth(n.zi\_S, n.luna\_S, n.an\_S);

printf ("Ziua de nastere a lui Ion: ");

scanf ("%d/%d/%d", &n.zi\_I, &n.luna\_I, &n.an\_I);

int n\_I = birth(n.zi\_I, n.luna\_I, n.an\_I);

printf ("Ziua de nastere a lui Andrei: ");

scanf ("%d/%d/%d", &n.zi\_A, &n.luna\_A, &n.an\_A);

int n\_A = birth(n.zi\_A, n.luna\_A, n.an\_A);

if ((n\_S > n\_A && n\_S > n\_I && n\_A > n\_I) || (n\_I > n\_A && n\_I > n\_S && n\_A > n\_S))

{

printf ("Cel mijlociu este Andrei");

}

else if ((n\_I > n\_S && n\_I > n\_A && n\_S > n\_A) || (n\_A > n\_S && n\_A > n\_I && n\_S > n\_I))

{

printf ("Cel mijlociu este Sergiu");

}

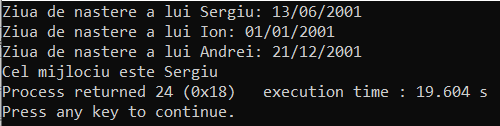
else

{

printf ("Cel mijlociu este Ion");

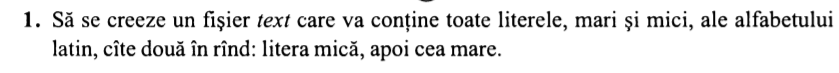
}

}



**Tip de date fișier (p. 114-116) - 6 probleme**

217//



#include <stdio.h>

#include <stdlib.h>

void main ()

{

FILE \* f = fopen("text.txt", "w");

for (int i = 97; i<=122; i++)

{

fprintf(f, "%c%c \n", i, i-32);

}

fclose(f);

char s;

f = fopen("text.txt", "r");

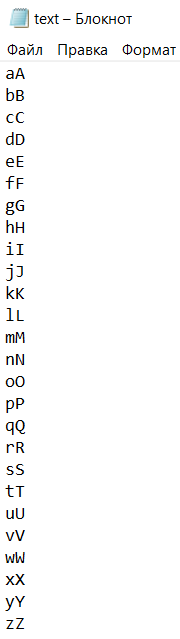
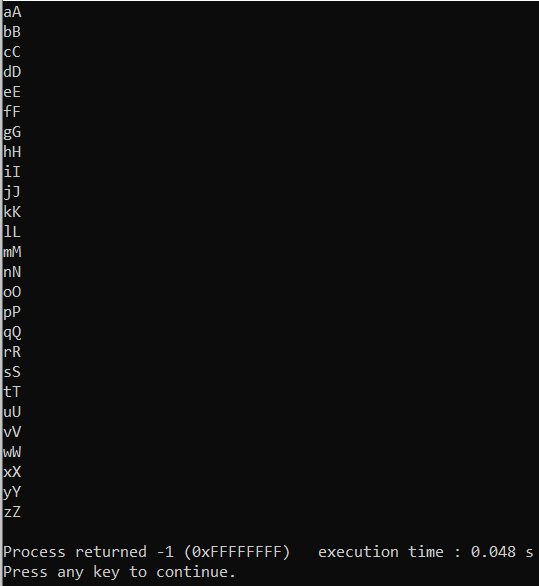
while ((s = fgetc(f))!= EOF)

{

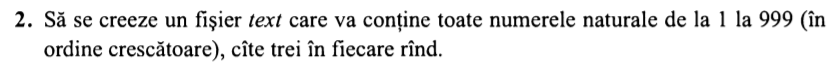
printf ("%c", s);

}

}

218//



#include <stdio.h>

#include <stdlib.h>

void main ()

{

FILE \* f = fopen("text.txt", "w");

int i = 1, count = 0;

while (i <= 999)

{

fprintf(f, "%d ", i);

i++;

count++;

if (count == 3)

{

fprintf (f, "\n");

count = 0;

}

}

fclose(f);

char s;

f = fopen("text.txt", "r");

while ((s = fgetc(f))!= EOF)

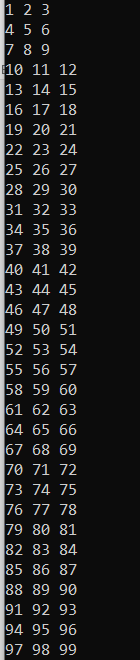
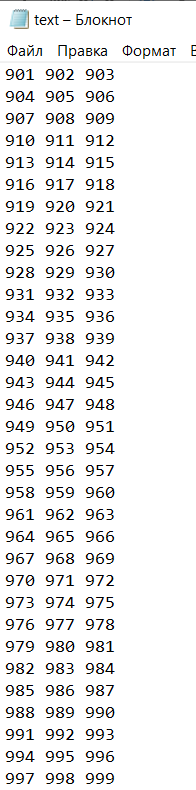
{

printf ("%c", s);

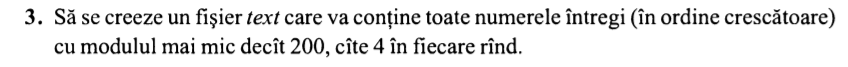
}

fclose(f);

}

219//



#include <stdio.h>

#include <stdlib.h>

void main ()

{

FILE \* f = fopen("text.txt", "w");

int i = -199, count = 0;

while (i < 200)

{

fprintf(f, "%d ", i);

count++;

i++;

if (count == 4)

{

fprintf (f, "\n");

count = 0;

}

}

fclose(f);

char s;

f = fopen("text.txt", "r");

while ((s = fgetc(f))!= EOF)

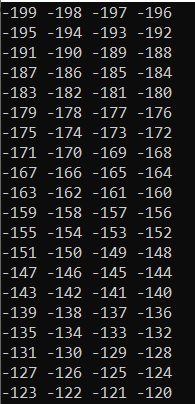
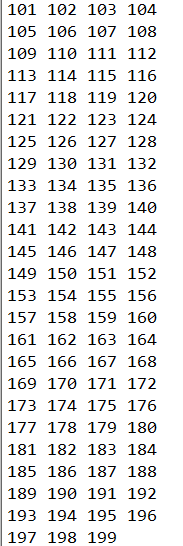
{

printf ("%c", s);

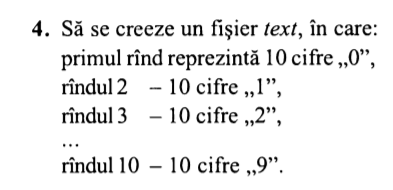
}

fclose(f);

}

220//



#include <stdio.h>

#include <stdlib.h>

void main ()

{

FILE \* f = fopen("text.txt", "w");

int i = 0, count = 0;

while (i < 10)

{

fprintf(f, "%d ", i);

count++;

if (count == 10)

{

fprintf (f, "\n");

count = 0;

i++;

}

}

fclose(f);

char s;

f = fopen("text.txt", "r");

while ((s = fgetc(f))!= EOF)

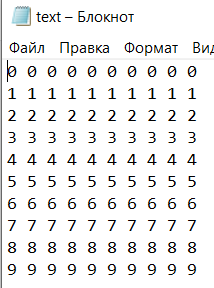
{

printf ("%c", s);

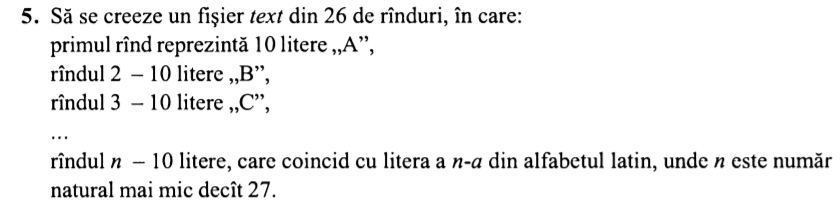
}

fclose(f);

}

221//



#include <stdio.h>

#include <stdlib.h>

void main ()

{

FILE \* f = fopen("text.txt", "w");

int i = 0, count = 0;

while (i < 26)

{

fprintf(f, "%c ", i+65);

count++;

if (count == 10)

{

fprintf (f, "\n");

count = 0;

i++;

}

}

fclose(f);

char s;

f = fopen("text.txt", "r");

while ((s = fgetc(f))!= EOF)

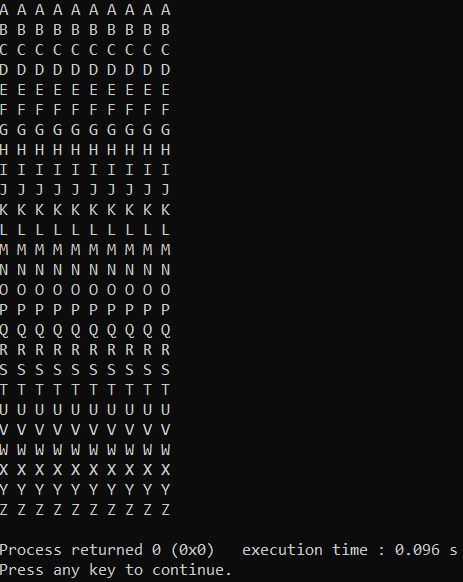
{

printf ("%c", s);

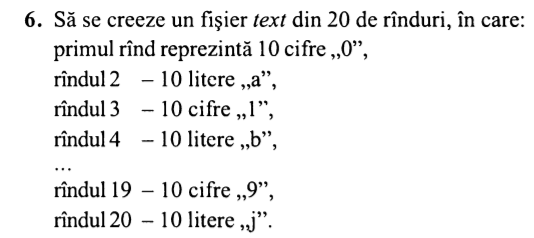
}

fclose(f);

}

222//



#include <stdio.h>

#include <stdlib.h>

void main ()

{

FILE \* f = fopen("text.txt", "w");

int lit = 1, count\_l = 0;

int num = 1, count\_n = 0;

int i = 0;

while(i < 10)

{

for(int j = 0; j < 10; j++)

{

fprintf(f, "%c ", lit+96);

}

fprintf(f, "\n");

for(int k = 0; k < 10; k++)

{

fprintf(f,"%d ", num);

}

fprintf(f,"\n");

i++;

num++;

lit++;

count\_l += 2;

count\_n += 2;

}

fclose(f);

char s;

f = fopen("text.txt", "r");

while ((s = fgetc(f))!= EOF)

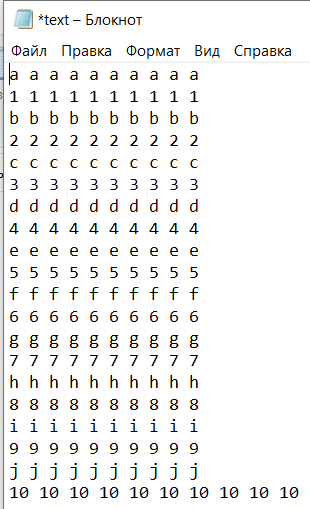
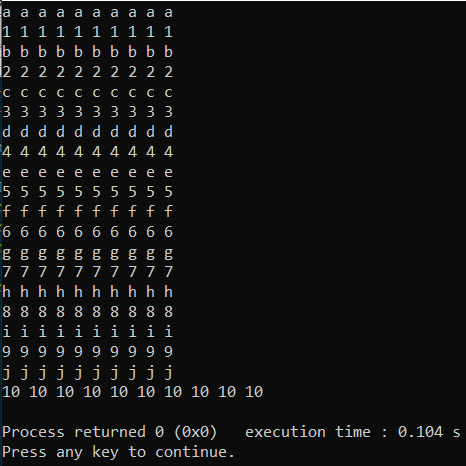
{

printf ("%c", s);

}

fclose(f);

}



**Probleme extrase din laboratoare (28 probleme)**

223//

1. Scrieți un program C pentru a număra numărul total de elemente pare și impare dintr-un tablou.

#include <stdio.h>

int main ()

{

int n, matrice[10], countpare=0, countimpare=0;

printf ("Introduceti numarul de elemente:");

scanf ("%d", &n);

printf ("Introduceti elementele:");

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

{

scanf ("%d", &matrice[i]);

}

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

{

if (matrice[i]%2==0)

{

countpare++;

}

if (matrice[i]%2==1)

{

countimpare++;

}

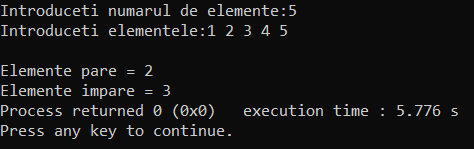
}

printf ("\n");

printf ("Elemente pare = %d\n", countpare);

printf ("Elemente impare = %d", countimpare);

}



224//

1. Scrieți un program C pentru a număra numărul total de elemente negative dintr-un tablou.

#include <stdio.h>

int main ()

{

int n, matrice[10], negative=0;

printf ("Introduceti numarul de elemente:");

scanf ("%d", &n);

printf ("Introduceti elementele:");

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

{

scanf ("%d", &matrice[i]);

}

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

{

if (matrice[i]<0)

{

negative++;

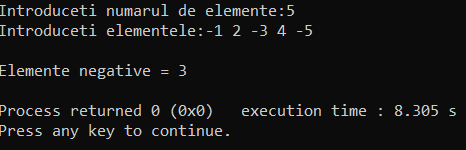
}

}

printf ("\n");

printf ("Elemente negative = %d\n", negative);

}



225//

1. Scrieți un program C pentru a copia toate elementele dintr-un tablou în alt tablou.

#include <stdio.h>

int main ()

{

int n, i, matrice1[10], matrice2[10];

printf ("Introduceti numarul de elemente:");

scanf ("%d", &n);

printf ("Introduceti elementele:");

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

{

scanf ("%d", &matrice1[i]);

}

printf ("Tablou copiat: \n");

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

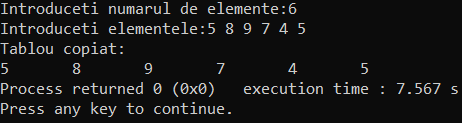
{

matrice2[i]=matrice1[i];

printf ("%d\t", matrice2[i]);

}

}



226//

1. Scrieți un program C pentru a insera un element într-un tablou pe o poziție dată de la tastatură.

#include <stdio.h>

int main ()

{

int n, i;

int poz, num;

int matrice[10]={};

printf ("Introduceti pozitia numarului dorit:");

scanf ("%d", &poz);

printf ("Introduceti numarul dorit:");

scanf ("%d", &num);

matrice[poz]=num;

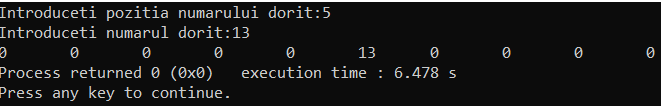
for (int i=0; i<10; i++)

{

printf ("%d\t", matrice[i]);

}

}



227//

1. Scrieți un program C pentru a șterge un element dintr-un tablou în poziția specificată de la tastatură.

#include <stdio.h>

int main ()

{

int n, i;

int poz, num;

int matrice[]={1, 2, 3, 4, 5, 6, 7, 8, 9};

printf ("Introduceti pozitia numarului dorit:");

scanf ("%d", &poz);

matrice[poz-1]=0;

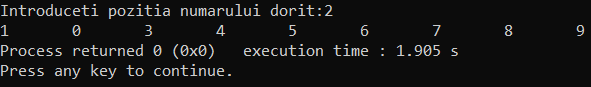
for (int i=0; i<9; i++)

{

printf ("%d\t", matrice[i]);

}

}



228//

1. Scrieți un program C pentru a număra frecvența relativă fiecărui element dintr-un tablou.

#include <stdio.h>

void main ()

{

int n, matrice[100], frecventa[100],k;

printf ("Introduceti numarul de elemente: ");

scanf ("%d", &n);

printf ("Introduceti elementele: ");

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

{

scanf ("%d", &matrice[i]);

frecventa[i]=-1;

}

printf ("\n");

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

{

int count = 1;

for (int j=i+1; j<n; j++)

{

if (matrice[i]==matrice[j])

{

count++;

for(k=j; k<n; k++)

{

matrice[k]=matrice[k+1];

}

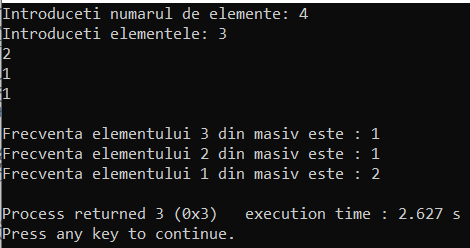
n--;

j--;

}

}

printf("Frecventa elementului %d din masiv este : %d\n",matrice[i],count);}



229//

1. Scrieți un program C pentru a imprima toate elementele unice din tablou.

#include <stdio.h>

void main()

{

int arr[100], n, count=0;

int i, j, k;

printf("Introduceti numarul de elemente: ");

scanf("%d",&n);

printf("Introduceti %d elemente ale vectorului:\n",n);

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

{

printf("Elementul %d: ",i+1);

scanf("%d",&arr[i]);

}

printf("\nElementele unice ale vectorului sunt: ");

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

{

count=0;

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

{

if (i!=j)

{

if(arr[i]==arr[j])

{

count++;

}

}

}

if(count==0)

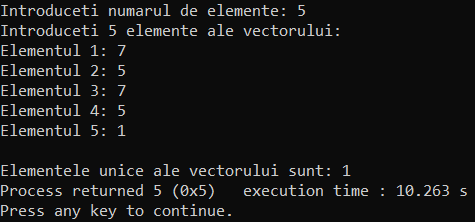
{

printf("%d ",arr[i]);

}

}

}



230//

1. Scrieți un program C pentru a număra numărul total de elemente duplicate dintr-un tablou.

#include <stdio.h>

void main()

{

int arr[100], n, count=0;

int i, j, k;

printf("Introduceti numarul de elemente: ");

scanf("%d",&n);

printf("Introduceti %d elemente ale vectorului:\n",n);

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

{

printf("Elementul %d: ",i+1);

scanf("%d",&arr[i]);

}

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

{

for (j=i+1; j<n; j++)

{

if (arr[i]==arr[j])

{

count++;

}

}

}

printf ("Numarul de elemente duplicate = %d", count);

}



231//

1. Scrieți un program C pentru a îmbina două tablouri în al treilea tablou.

#include <stdio.h>

void main ()

{

int arr1[100], arr2[100], arr\_fin[100];

int size1, size2, sizefin, i, j, size\_fin;

printf ("Introduceti numarul de elemente pentru primul tablou: ");

scanf ("%d", &size1);

printf ("Introduceti elementele primului tablou \n");

for (i=0; i<size1; i++)

{

scanf ("%d", &arr1[i]);

}

printf ("Introduceti numarul de elemente pentru cel de-al doilea tablou: ");

scanf ("%d", &size2);

printf ("Introduceti elementele celui de-al doilea tablou \n");

for (i=0; i<size2; i++)

{

scanf ("%d", &arr2[i]);

}

size\_fin=size1+size2;

for (i=0; i<size1; i++)

{

arr\_fin[i]=arr1[i];

}

for (i=0, j=size1; j<size\_fin && i<size2; i++, j++)

{

arr\_fin[j]=arr2[i];

}

printf ("Tabloul final \n");

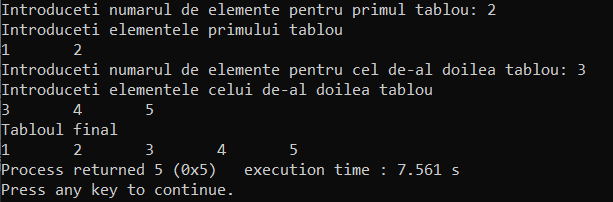
for (i=0; i<size\_fin; i++)

{

printf ("%d\t", arr\_fin[i]);

}

}



232//

1. Scrieți un program C pentru a inversa ordinea elementelor unui tablou.

#include <stdio.h>

void main ()

{

int arr[100];

int size, i, j, temp;

printf ("Introduceti numarul de elemente pentru primul tablou: ");

scanf ("%d", &size);

printf ("Introduceti elementele primului tablou \n");

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

{

scanf ("%d", &arr[i]);

}

for (i=0, j=size-1; i<j; i++, j--)

{

temp=arr[i];

arr[i]=arr[j];

arr[j]=temp;

}

printf ("Tabloul final \n");

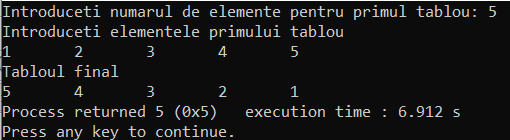
for (i=0; i<size; i++)

{

printf ("%d\t", arr[i]);

}

}



233//

1. Scrieți un program C pentru a pune elemente pare și impare ale unui tablou în două tablouri separate.

#include <stdio.h>

void main ()

{

int array[100], pare[100], impare[100];

int n, num, i, m=0, k=0;

printf ("Introduceti numarul de elemente:");

scanf ("%d", &n);

printf ("Introduceti elementele tabloului initial: ");

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

{

scanf ("%d", &array[i]);

}

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

{

if (array[i]%2==0)

{

pare[m]=array[i];

m++;

}

else

{

impare[k]=array[i];

k++;

}

}

printf ("Tabloul par: \n");

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

{

printf ("%d\t", pare[i]);

}

printf ("\nTablou impar: \n");

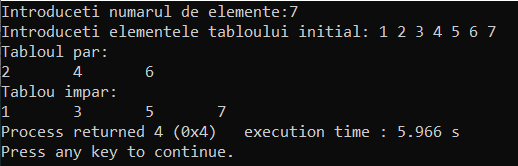
for (i=0; i<k; i++)

{

printf ("%d\t", impare[i]);

}

}



234//

1. Scrieți un program C pentru a căuta un element din tablou și afiza pozițiile găsite.

#include <stdio.h>

void main ()

{

int array[100];

int n, num, i;

printf ("Introduceti numarul de elemente:");

scanf ("%d", &n);

printf ("Introduceti elementele tabloului: ");

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

{

scanf ("%d", &array[i]);

}

printf ("Introduceti numarul dorit pentru cautare:");

scanf ("%d", &num);

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

{

if (array[i]==num)

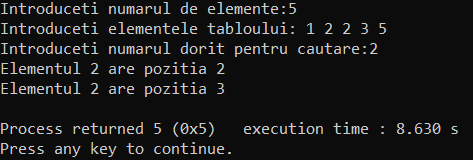
{

printf ("Elementul %d are pozitia %d\n", num, i+1);

}

}

}



235//

1. Scrieți un program C pentru a sorta elementele tabloului în ordine crescătoare sau descrescătoare.

#include <stdio.h>

void main ()

{

int arr[100];

int size, i,j, temp, caz;

printf ("Introduceti numarul de elemente pentru primul tablou: ");

scanf ("%d", &size);

printf ("Introduceti elementele tabloului: \n");

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

{

scanf ("%d", &arr[i]);

}

printf("Pentru ca elementele sa apara:\nIn ordine crescatoare - introduceti 1\nIn ordine descrescatoare - introduceti 2\n");

scanf ("%d", &caz);

switch (caz)

{

case 1:

for(int i=0; i<size; i++)

{

for(int j=i+1; j<size; j++)

{

if(arr[i]<arr[j])

{

temp=arr[i];

arr[i]=arr[j];

arr[j]=temp;

}

}

}

case 2:

for(int i=0; i<size; i++)

{

for(int j=i+1; j<size; j++)

{

if(arr[i]>arr[j])

{

temp=arr[i];

arr[i]=arr[j];

arr[j]=temp;

}

}

}

}

printf ("Tabloul final \n");

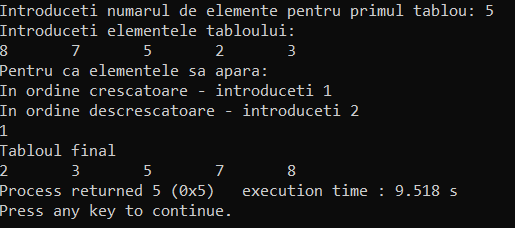
for (i=0; i<size; i++)

{

printf ("%d\t", arr[i]);

}

}



236//

1. Scrieți un program C pentru a roti la stânga un tablou, operația ROL.

#include <stdio.h>

void main ()

{

int arr[100];

int n, i, temp;

printf ("Introduceti numarul de elemente:");

scanf ("%d", &n);

printf ("\n");

printf ("Introduceti elementele tabloului initial: ");

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

{

scanf ("%d", &arr[i]);

}

temp=arr[0];

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

{

arr[i]=arr[i+1];

}

arr[n-1]=temp;

printf ("\n");

printf ("Tablou rotit la stanga cu o pozitie: ");

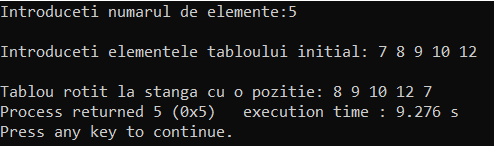
for (int i=0; i<n; i++)

{

printf ("%d\t", arr[i]);

}

}



237//

1. Scrieți un program C pentru a roti dreapta un tablou, operația ROR.

#include <stdio.h>

void main ()

{

int arr[100];

int n, i, temp, temp1;

printf ("Introduceti numarul de elemente:");

scanf ("%d", &n);

printf ("\n");

printf ("Introduceti elementele tabloului initial: ");

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

{

scanf ("%d", &arr[i]);

}

temp=arr[n-1];

for (i=n-1; i>0; i--)

{

arr[i]=arr[i-1];

}

arr[0]=temp;

printf ("\n");

printf ("Tablou rotit la dreapta cu o pozitie: ");

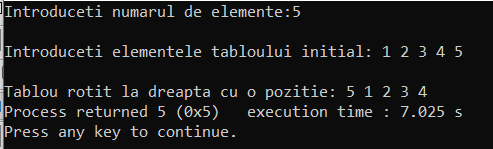
for (int i=0; i<n; i++)

{

printf ("%d ", arr[i]);

}

}



238//

1. Scrieți un program C pentru a găsi suma elementelor diagonale principale ale unei matrice.

#include <stdio.h>

int main ()

{

int i, j;

int row, col;

int arr1[10][10], sum;

printf ("Introduceti numarul de randuri: ");

scanf ("%d", &row);

printf ("Introduceti numarul de coloane : ");

scanf ("%d", &col);

printf ("\n");

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

{

for (j=0; j<col; j++)

{

printf ("Elementul %d%d = ",i+1, j+1);

scanf("%d", &arr1[i][j]);

}

}

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

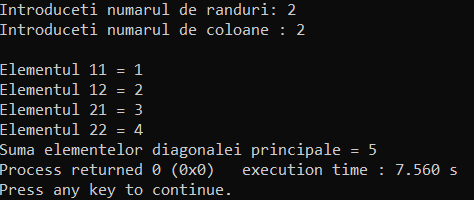
{

sum+=arr1[i][i];

}

printf ("Suma elementelor diagonalei principale = %d",sum);

}



239//

1. Scrieți un program C pentru a găsi suma elementelor diagonalei secundare ale unei matrice.

#include <stdio.h>

int main ()

{

int i, j;

int row, col;

int arr1[10][10], sum;

printf ("Introduceti numarul de randuri: ");

scanf ("%d", &row);

printf ("Introduceti numarul de coloane : ");

scanf ("%d", &col);

printf ("\n");

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

{

for (j=0; j<col; j++)

{

scanf("%d", &arr1[i][j]);

}

}

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

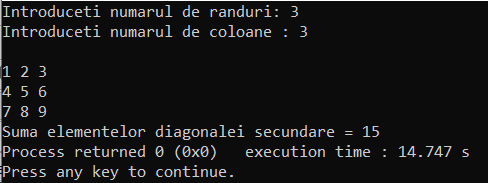
{

sum+=arr1[i][row-1-i];

}

printf ("Suma elementelor diagonalei secundare = %d",sum);

}



240//

1. Scrieți un program C pentru a găsi suma fiecărui rând și coloană a unei matrice.

#include <stdio.h>

int main ()

{

int i, j;

int row, col;

int arr[10][10], sum;

printf ("Introduceti numarul de randuri: ");

scanf ("%d", &row);

printf ("Introduceti numarul de coloane: ");

scanf ("%d", &col);

printf ("\n");

printf ("Introduceti elementele matricei: ");

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

{

for (j=0; j<col; j++)

{

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

}

}

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

{

for (j = 0; j < col; j++)

{

sum=sum+arr[i][j] ;

}

printf("Suma elementelor randului %d = %d\n", i+1, sum);

sum = 0;

}

for (j = 0; j < col; ++j)

{

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

{

sum = 0;

sum = sum + arr[i][j];

}

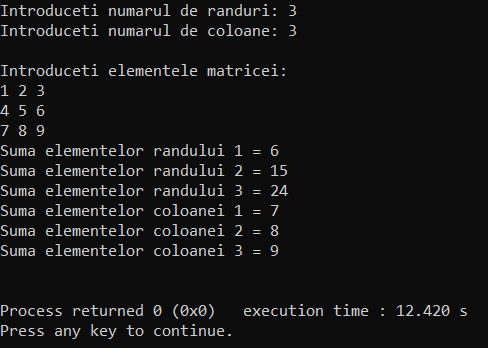
printf("Suma elementelor coloanei %d = %d\n", j+1, sum);

sum = 0;

}

printf ("\n");

}



241//

1. Scrieți un program C pentru a schimba diagonalele unei matrice.

#include <stdio.h>

void main()

{

int i, j, row, col, arr[10][10], temp;

printf("Introduceti numarul de randuri: ");

scanf("%d", &row);

printf("Introduceti numarul de coloane: ");

scanf("%d", &col);

printf("Introduceti elementele matricei patratice: \n");

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

{

for(j=0;j<col; j++)

{

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

}

}

if(i!=j)

printf ("Matricea introdusa nu este una patratica");

else

{

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

{

temp = arr[i][i];

arr[i][i] = arr[i][row-i-1];

arr[i][row-i-1] = temp;

}

printf("\n Matricea dupa schimbarea diagonalelor: \n");

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

{

for(j = 0; j < col; j++)

{

printf("%d \t ", arr[i][j]);

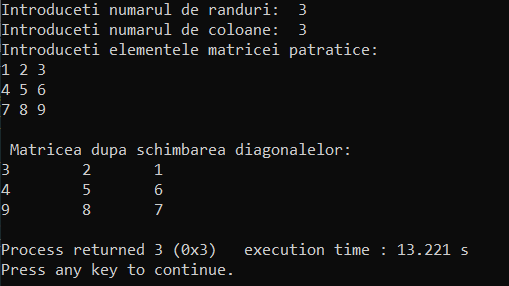
}

printf("\n");

}

}

}



242//

1. Scrieți un program C pentru a găsi matricea triunghiulară superioară.

#include <stdio.h>

void main()

{

int i, j, row, col, arr[10][10];

printf("Introduceti numarul de randuri: ");

scanf("%d", &row);

printf("Introduceti numarul de coloane: ");

scanf("%d", &col);

printf("Introduceti elementele matricei: \n");

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

{

for (j = 0; j < col; j++)

{

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

}

}

printf("\n");

printf("Matricea triunghiulara superioara: \n");

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

{

printf("\n");

for (j = 0; j < col; j++)

{

if (i<=j)

{

printf("%d\t", arr[i][j]);

}

else

{

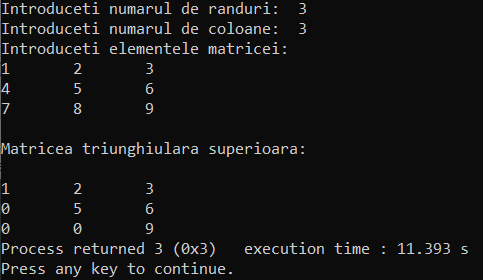
printf("0\t");

}

}

}

}



243//

1. Scrieți un program C pentru a găsi matricea triunghiulară inferioară.

#include <stdio.h>

void main()

{

int i, j, row, col, arr[10][10];

printf("Introduceti numarul de randuri: ");

scanf("%d", &row);

printf("Introduceti numarul de coloane: ");

scanf("%d", &col);

printf("Introduceti elementele matricei: \n");

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

{

for (j = 0; j < col; j++)

{

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

}

}

printf("Matricea triunghiulara inferioara: \n");

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

{

printf("\n");

for (j = 0; j < col; j++)

{

if (j<=i)

{

printf("%d\t", arr[i][j]);

}

else

{

printf("0\t");

}

}

}

}



244//

1. Scrieți un program C pentru a găsi suma matricei triunghiulare superioare.

#include <stdio.h>

void main()

{

int i, j, row, col, arr[10][10], sum=0;

printf("Introduceti numarul de randuri: ");

scanf("%d", &row);

printf("Introduceti numarul de coloane: ");

scanf("%d", &col);

printf("Introduceti elementele matricei: \n");

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

{

for (j = 0; j < col; j++)

{

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

}

}

printf("\n");

printf("Matricea triunghiulara superioara: \n");

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

{

printf("\n");

for (j = 0; j < col; j++)

{

if (i<=j)

{

sum+=arr[i][j];

printf("%d\t", arr[i][j]);

}

else

{

printf("0\t");

}

}

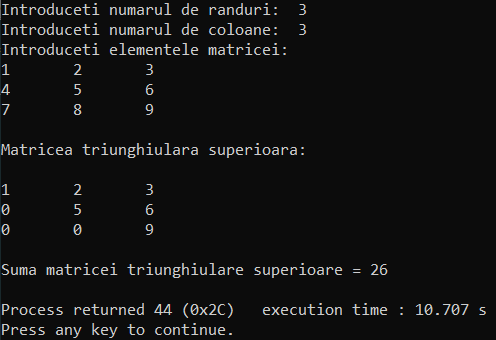
}

printf("\n");

printf("\n");

printf ("Suma matricei triunghiulare superioare = %d\n", sum);

}



245//

1. Scrieți un program C pentru a găsi suma matricei triunghiulare inferioare.

#include <stdio.h>

void main()

{

int i, j, row, col, arr[10][10], sum=0;

printf("Introduceti numarul de randuri: ");

scanf("%d", &row);

printf("Introduceti numarul de coloane: ");

scanf("%d", &col);

printf("Introduceti elementele matricei: \n");

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

{

for (j = 0; j < col; j++)

{

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

}

}

printf("\n");

printf("Matricea triunghiulara inferioara: \n");

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

{

printf("\n");

for (j = 0; j < col; j++)

{

if (j<=i)

{

sum+=arr[i][j];

printf("%d\t", arr[i][j]);

}

else

{

printf("0\t");

}

}

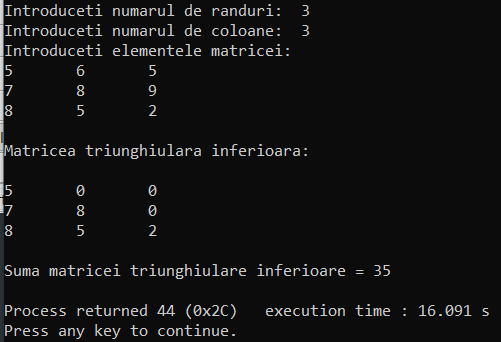
}

printf("\n");

printf("\n");

printf ("Suma matricei triunghiulare inferioare = %d\n", sum);

}



246//

1. Scrieți un program C pentru a roti la stînga o matrice.

#include <stdio.h>

int main()

{

int row, col;

int i, j;

printf("Introduceti numarul de randuri: ");

scanf("%d", &row);

printf("Introduceti numarul de coloane: ");

scanf("%d", &col);

int arr[10][10];

printf("Introduceti elementele matricei: \n");

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

{

for (j = 0; j < col; j++)

{

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

}

}

printf("\n");

printf ("Matricea rotita la stanga: \n");

rotate\_left(arr, row, col);

}

void rotate\_left(int m[10][10],int row, int col)

{

for(int i=row-1; i>=0; i--)

{

for(int j=0; j<col; j++)

{

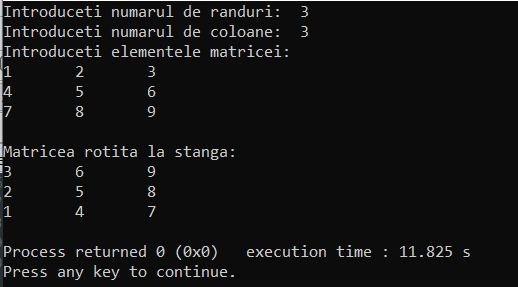
printf("%d\t",m[j][i]);

}

printf("\n");

}

}



247//

1. Scrieți un program C pentru a roti la dreapta o matrice.

#include <stdio.h>

int main()

{

int row, col;

int i, j;

printf("Introduceti numarul de randuri: ");

scanf("%d", &row);

printf("Introduceti numarul de coloane: ");

scanf("%d", &col);

int arr[10][10];

printf("Introduceti elementele matricei: \n");

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

{

for (j = 0; j < col; j++)

{

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

}

}

printf("\n");

rotate(arr, row, col);

}

void rotate(int m[10][10],int row, int col)

{

for(int i=0; i<row; i++)

{

for(int j=col-1; j>=0; j--)

{

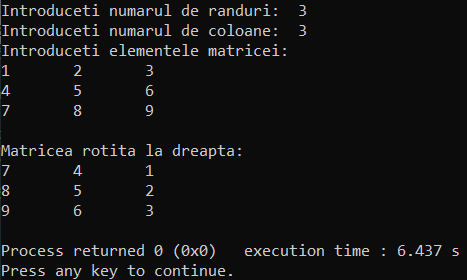
printf("%d\t",m[j][i]);

}

printf("\n");

}

}



248//

1. Scrieți un program C pentru a roti în sus o matrice.

#include <stdio.h>

int main()

{

int row, col;

int i, j;

printf("Introduceti numarul de randuri: ");

scanf("%d", &row);

printf("Introduceti numarul de coloane: ");

scanf("%d", &col);

int arr1[10][10], arr2[10][10];

printf("Introduceti elementele matricei: \n");

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

{

for (j = 0; j < col; j++)

{

scanf("%d", &arr1[i][j]);

}

}

printf("\n");

printf ("Matricea rotita in sus: \n");

rotate\_up(arr1, arr2, row, col);

for(int i=0; i<row; i++)

{

for(int j=0; j<col; j++)

{

printf("%d\t",arr2[i][j]);

}

printf("\n");

}

}

void rotate\_up(int m[10][10], int n[10][10], int row, int col)

{

for(int i=0; i<row; i++)

{

for(int j=0; j<col; j++)

{

if (i==0)

{

n[row-1][j]=m[i][j];

}

else

{

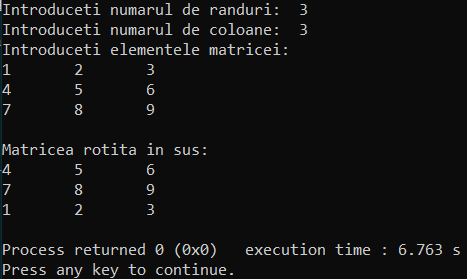
n[i-1][j]=m[i][j];

}

}

}

}

****

249//

1. Scrieți un program C pentru a roti în jos o matrice.

#include <stdio.h>

int main()

{

int row, col;

int i, j;

printf("Introduceti numarul de randuri: ");

scanf("%d", &row);

printf("Introduceti numarul de coloane: ");

scanf("%d", &col);

int arr[10][10];

printf("Introduceti elementele matricei: \n");

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

{

for (j = 0; j < col; j++)

{

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

}

}

printf("\n");

printf ("Matricea rotita in jos: \n");

rotate\_down(arr, row, col);

for(int i=0; i<row; i++)

{

for(int j=0; j<col; j++)

{

printf("%d\t",arr[i][j]);

}

printf("\n");

}

}

void rotate\_down(int m[10][10], int row, int col)

{

int temp;

for(int i=0; i<row; i++)

{

for(int j=0; j<col; j++)

{

temp = m[i][j];

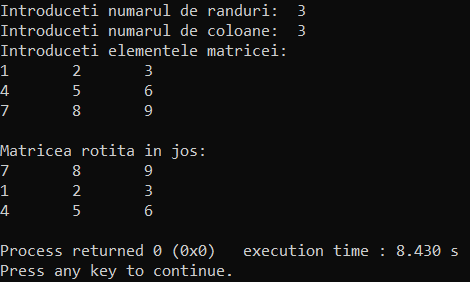
m[i][j]= m[row-1][j];

m[row-1][j]=temp;

}

}

}



250//

1. Scrieți un program C pentru a găsi numărul total de litere, cifre sau caractere speciale dintr-un șir.

#include <stdio.h>

#include <string.h>

void main ()

{

char text [100];

char text\_1 [100];

printf ("Introduceti textul: ");

gets(text);

str(text);

}

void str(char text[], char text\_1[])

{

int count\_num = 0;

for (int i = 0; i<strlen(text); i++)

{

if (text[i]>=48 && text[i]<=57)

{

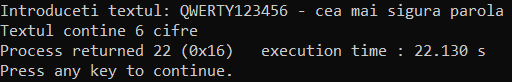
count\_num++;

}

}

printf ("Textul contine %d cifre", count\_num);

}



251//

1. Scrieți un program C pentru a număra numărul total de consoane dintr-un șir.

#include <stdio.h>

#include <string.h>

char num(int j, char str[])

{

int count = 0;

for (int j=0; str[j]; j++)

{

if (str[j]=='a'||str[j]=='A'||str[j]=='o'||str[j]=='O'||str[j]=='E'||str[j]=='e'||str[j]=='i'||str[j]=='I'||str[j]=='U'||str[j]=='u')

count++;

}

printf ("In text sunt %d consoane", strlen(str)-count);

}

void main ()

{

char str1[100], i;

printf ("Introduceti textul: ");

gets(str1);

num(i, str1);

}

