**Laborator №8 : *Funcții***

1. Să se realizeze un program care, folosind un meniu, să calculeze ariile mai multor figuri geometrice, fiecare arie fiind calculată folosind câte o funcție.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

#include <math.h>

void \_menu(void)

{

printf("\n\n------------- Meniu Interactiv ---------------\n"

"0. Iesire\n"

"1. Calculeaza aria unui triunghi\n"

"2. Calculeaza aria unui cerc\n"

"3. Calculeaza aria unui patrat\n"

"4. Calculeaza aria unui dreptunghi\n"

"Optiunea dvs : "

);

}

void \_arie\_triunghi(float\* a, float\* b, float\* c)

{

float s, p;

p = ((\*a) + (\*b) + (\*c)) / 2;

s = sqrt(p \* (p - (\*a)) \* (p - (\*b)) \* (p - (\*c)));

printf("Aria triunghiului : %.2f u.p.\n", s);

}

void \_arie\_patrat(float\* l)

{

printf("Aria patratului : %.2f u.p.\n", (\*l) \* (\*l));

}

void \_arie\_cerc(float\* r)

{

printf("Aria cercului : %.2f u.p.\n", 3.14 \* (\*r) \* (\*r));

}

void \_arie\_dreptunghi(float\* l1, float\* l2)

{

printf("Aria dreptunghiului : %.2f\n", (\*l1) \* (\*l2));

}

int main(void)

{

int opt;

float a, b, c, r;

while (1)

{

\_menu();

scanf("%d", &opt);

switch (opt)

{

case 0:

exit(0);

break;

case 1:

{

printf("Introduceti laturile triunghiului : \n");

printf("a : "); scanf("%f", &a);

printf("b : "); scanf("%f", &b);

printf("c : "); scanf("%f", &c);

\_arie\_triunghi(&a, &b, &c);

}

break;

case 2:

{

printf("Introduceti raza cercului : ");

scanf("%f", &r);

\_arie\_cerc(&r);

}

break;

case 3:

{

printf("Introduceti latura unui patrat : ");

scanf("%f", &a);

\_arie\_patrat(&a);

}

break;

case 4:

{

printf("Introduceti laturile unui dreptunghi : \n");

printf("L1 : "); scanf("%f", &a);

printf("L2 : "); scanf("%f", &b);

\_arie\_dreptunghi(&a, &b);

}

break;

default:

printf("Eroare! Optiunea nedefinita !");

break;

}

}

return 0;

}

**2)** Să se realizeze un program care să rezolve o ecuație de gradul 2 cu coeficienții citiți de la tastatură, folosind o funcție care să returneze două valori - rădacinile, și să precizeze dacă are sau nu rădacini reale).

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

#include <math.h>

void \_ecuatie\_gr\_2(int\* a, int\* b, int\* c, float\* x1, float\* x2)

{

float delta;

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

if (delta == 0)

{

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

\*x2 = (float)(-(\*b) / (2 \* (\*a)));

printf("\nSolutia ecuatiei : x1 = x2 = %.2f", x1);

}

else if (delta > 0)

{

printf("\nSolutiile ecuatiei : \n");

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

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

printf("x1 = %.2f\n", \*x1);

printf("x2 = %.2f\n", \*x2);

}

else

{

printf("Ecuatia nu are solutii reale !\n");

}

}

int main(void)

{

int a, b, c;

float x1, x2;

printf("Introduceti coeficentii unei ecuatii de gradul 2 : \n");

printf("a : "); scanf("%d", &a);

printf("b : "); scanf("%d", &b);

printf("c : "); scanf("%d", &c);

\_ecuatie\_gr\_2(&a, &b, &c, &x1, &x2);

printf("\n");

return 0;

}

1. Să se realizeze un program care să determine dacă un număr citit de la tastatură este prim, utilizând o funcție definită de utilizator.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

#include <math.h>

int \_is\_prime(int\* n)

{

int i, prime = 1;

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

{

if (\*n % i == 0)

prime = 0;

}

return prime;

}

int main(void)

{

int n;

printf("Introduceti un numar : ");

scanf("%d", &n);

switch (\_is\_prime(&n))

{

case 0:

printf("Numarul introdus nu este prim !\n");

break;

case 1:

printf("Numarul introdus este prim !\n");

break;

default:

break;

}

printf("\n");

return 0;

}

1. Să se scrie un program care determină toate numerele prime mai mici sau egale cu un numar N dat de la tastatură.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

#include <math.h>

int \_is\_prime(int\* n)

{

int i, prime = 1;

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

{

if (\*n % i == 0)

prime = 0;

}

return prime;

}

int main(void)

{

int i, n;

printf("Introduceti un numar : ");

scanf("%d", &n);

printf("Toate numerele prime <= %d : \n", n);

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

if (\_is\_prime(&i) == 1)

printf("%d ", i);

printf("\n");

return 0;

}