**Laborator №5 : *Structuri statice și dinamice***

1. Să se scrie un program în C care să citească de la tastatură două matrice pătratice de numere întregi, de dimensiune specificată de utilizator şi să afişeze suma celor două matrice.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int main(void)

{

int m1[100][100], m2[100][100], n, i, j;

printf("Programul calculeaza suma a doua matrice.\n"

"Introduceti dimensiunea matricelor : ");

scanf("%d", &n);

printf("\nIntroduceti elementele primei matrice : \n");

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

{

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

{

printf("matrice1[%d][%d] : ", i, j);

scanf("%d", &m1[i][j]);

}

printf("\n");

}

printf("\nIntroduceti elementele matricei secunde : \n");

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

{

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

{

printf("matrice2[%d][%d] : ", i, j);

scanf("%d", &m2[i][j]);

}

printf("\n");

}

printf("\nMatricea 1 : \n");

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

{

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

{

printf("%4d", m1[i][j]);

}

printf("\n");

}

printf("\nMatricea 2 : \n");

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

{

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

{

printf("%4d", m2[i][j]);

}

printf("\n");

}

printf("\nMatricea suma : \n");

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

{

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

{

printf("%4d", m1[i][j]+m2[i][j]);

}

printf("\n");

}

printf("\n");

return 0;

}

**2)** Să se scrie un program în C care, folosind un meniu interactive să conțină următoarele opțiuni: 1. Citirea unui vector cu n elemente 2. Afișarea vectorului 3. Afișarea elementelor de pe poziițile pare 4. Afișarea produsului elementelor impare 5. Ieșire  
  
#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

void \_menu(void)

{

printf("Meniul interactiv : \n"

"1. Citirea unui vector cu n elemente\n"

"2. Afisarea vectorului\n"

"3. Afisarea elementelor de pe pozitiile pare\n"

"4. Afisarea produsului elementelor impare\n"

"5. Iesire\n\n"

"Optiunea dvs : ");

}

int main(void)

{

int opt, n, i, v[100], ok, p;

ok = 0;

while (1)

{

printf("\n");

\_menu();

scanf("%d", &opt);

if (opt == 5)

break;

printf("\n");

switch (opt)

{

case 1 :

{

ok = 1;

printf("Introduceti nr de elemente al vectorului : ");

scanf("%d", &n);

printf("\n");

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

{

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

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

}

}

break;

case 2:

{

if (ok == 1)

{

printf("Elementele vectorului : \n");

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

{

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

}

}

else

printf("Introduceti elementele vectorului mai intai !\n");

}

break;

case 3:

{

if (ok == 1)

{

printf("Elementele vectorului de pe pozitiile pare: \n");

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

if (i % 2 == 0)

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

}

else

printf("Introduceti elementele vectorului mai intai !\n");

}

break;

case 4:

{

if (ok == 1)

{

p = 1;

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

if (v[i] % 2 != 0)

p \*= v[i];

printf("Produsul elementelor impare: %d", p);

printf("\n");

}

else

printf("Introduceti elementele vectorului mai intai !\n");

}

break;

default:

printf("Optiune nedefinita !\n");

break;

}

}

printf("\n");

return 0;

}

1. Să se calculeze suma elementelor de pe diagonala principală a unei matrice pătratice citite de la tastatură.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int main(void)

{

int dim, i, j, m[100][100], sum1=0, sum2=0;

printf("Introduceti dimensiunea matricei patratice : ");

scanf("%d", &dim);

printf("\n");

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

{

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

{

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

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

}

printf("\n");

}

printf("\nMatricea introdusa : \n\n");

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

{

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

{

printf("%4d", m[i][j]);

}

printf("\n");

}

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

{

sum1 += m[i][i];

sum2 += m[i][dim - i - 1];

}

printf("\nSuma elementelor de pe diagonala principala : %d", sum1);

printf("\nSuma elementelor de pe diagonala secundara : %d", sum2);

printf("\n");

return 0;

}

1. Să se calculeze suma elementelor dintr-un vector aflate pe poziţii impare.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int main(void)

{

int v[100], dim, i, sum = 0;

printf("Introduceti dimensiunea vectorului : ");

scanf("%d", &dim);

printf("\n");

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

{

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

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

}

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

if (i % 2 != 0)

sum += v[i];

printf("\nSuma elementelor vectorului aflate pe pozitii impare : %d", sum);

printf("\n");

return 0;

}

1. Să se scrie un program în C care să citească de la tastatură o matrice pătratică. Să se creeze un meniu interactive cu următoarele opțiuni: 1. Afișarea matricii pătratice 2. Să se afișeze suma numerelor pare deasupra diagonalei principale 3. Să se afișeze produsul elementelor impare de pe diagonal secundară 4. Să se afișeze elementele prime din matrice 5. Iesire

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

void \_menu(void)

{

printf("\nMeniu interactiv : \n"

"1. Afisarea matricii patratice\n"

"2. Suma numerelor pare deasupra diagonalei principale\n"

"3. Produsul elementelor impare de pe diagonala secundara\n"

"4. Elementele prime din matrice\n"

"5. Iesire\n\n"

"Optiunea dvs : ");

}

int \_prime\_check(int n)

{

int i, prim = 1;

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

{

if (n % i == 0)

prim = 0;

}

return prim;

}

int main(void)

{

int dim, i, j, m[100][100], opt, sum, p;

printf("Introduceti dimensiunea matricei patratice : ");

scanf("%d", &dim);

printf("\n");

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

{

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

{

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

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

}

printf("\n");

}

while (1)

{

\_menu();

scanf("%d", &opt);

if (opt == 5)

break;

switch (opt)

{

case 1:

{

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

{

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

{

printf("%4d", m[i][j]);

}

printf("\n");

}

}

break;

case 2:

{

sum = 0;

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

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

if (j > i)

if (m[i][j] % 2 == 0)

sum += m[i][j];

printf("Suma elementelor pare deasupra diagonalei principale : %d\n", sum);

}

break;

case 3:

{

p = 1;

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

if (m[i][dim - i - 1] % 2 != 0)

p \*= m[i][dim - i - 1];

printf("Produsul numerelor impare de pe deagonala secundara : %d\n", p);

}

break;

case 4:

{

printf("Elementele prime din matrice : \n");

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

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

if (\_prime\_check(m[i][j]) == 1)

printf("%d " , m[i][j]);

printf("\n");

}

break;

default:

printf("Optiunea nedefinita !\n");

break;

}

}

return 0;

}