**Laborator №4 : *Instrucțiunile repetitive***

1. Să se scrie un program în C care citește de la tastatură un număr şi afișează toți divizorii acestuia.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int main(void)

{

int n;

printf("Introduceti un numar intreg : "); scanf("%d", &n);

printf("\n\nDivizorii numarului %d : \n", n);

// O(sqrt(n))

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

{

if (n % i == 0)

{

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

if (i != n / i)

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

}

}

printf("\n");

return 0;

}

**2)** Să se scrie un program în C care citește de la tastatură un număr întreg şi afișează daca acesta este număr prim sau nu.  
  
#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

#include <conio.h>

// O(sqrt(n))

int \_prime\_check(int n)

{

if (n == 1)

return 0;

else

{

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

{

if (n % i == 0)

return 0;

}

}

return 1;

}

int main(void)

{

int n;

printf("Introduceti un numar intreg : "); scanf("%d", &n);

switch (\_prime\_check(n))

{

case 1 :

printf("Numarul %d este prim ! ", n);

break;

case 0:

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

break;

}

printf("\n");

return 0;

}

1. Să se scrie un program în C care citește de la tastatură un şir de numere încheiat cu numărul 0 şi afișează suma numerelor introduse

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int main(void) {

int n, s = 0, i = 0;

printf("Programul citeste un sir de numere, incheiat cu introducerea lui 0.\n"

"Apoi afiseaza suma numerelor introduse.\n\n");

while (1) {

printf("[%d] : ", i); scanf("%d", &n);

if (n == 0)

break;

s += n;

i++;

}

printf("\nSuma numerelor introduse : %d\n", s);

return 0;

}

1. Să se scrie un program în C care citește de la tastatură un şir de numere încheiat cu numărul 0 şi afișează media aritmetică a numerelor introduse.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int main(void)

{

int i = 0;

float n, s = 0;

printf("Introduceti un sir de numere reale, care se va incheia cu introducerea lui 0 : \n\n");

while (1)

{

printf("[ %d ] : ", i); scanf("%f", &n);

s += n;

i++;

if (n == 0)

break;

}

printf("\nMedia aritmetica a numerelor sirului introdus : %.2f", s/(i-1));

printf("\n");

return 0;

}

1. Să se se scrie un program în C care citește de la tastatură un şir de numere încheiat cu numărul 0 şi afișează maximul dintre numerele introduse.  
     
     
   #define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int main(void)

{

int i = 0;

float n, s = 0,max;

printf("Introduceti un sir de numere reale, care se va incheia cu introducerea lui 0 : \n\n");

printf("[ %d ] : ", i); scanf("%f", &n);

max = n;

if (n != 0)

{

i++;

while (1)

{

printf("[ %d ] : ", i); scanf("%f", &n);

if (n > max)

max = n;

i++;

if (n == 0)

break;

}

}

printf("\nValoarea maxima a numerelor sirului introdus : %.2f", max);

printf("\n");

return 0;

}

1. Se citește un număr n. Să se calculeze produsul cifrelor pare din număr.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int main(void)

{

int n,d,p=1;

printf("Introduceti un numar natural : ");

scanf("%d", &n);

printf("\n");

while (n != 0)

{

d = n % 10;

if (d % 2 == 0) {

p \*= d;

}

n = n / 10;

}

printf("Produsul cifrelor pare ale nr. introdus : %d\n", p);

return 0;

}

1. Se citesc n numere. Să se calculeze pentru fiecare număr produsul cifrelor prime.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int \_prime\_check(int n)

{

int p=1;

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

{

if (n % i == 0) {

p = 0;

}

}

return p;

}

int main(void)

{

int n,nr,p=1,d;

printf("Urmeaza sa introduceti n numere de la tastatura.\n"

"Programul calculeaza produsul cifrelor prime pentru fiecare numar introdus.\n"

"Ce valoare preferati pentru n ?\n"

"\nn : ");

scanf("%d", &n);

printf("\n");

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

{

printf("[%d] : ", i); scanf("%d", &nr);

while (nr != 0) {

d = nr % 10;

if (\_prime\_check(d) == 1) {

p \*= d;

}

nr = nr / 10;

}

printf("Produsul cifrelor pare : %d\n\n", p);

p = 1;

}

return 0;

}

1. Se citesc n numere. Să se determine suma numerelor divizibile cu 5 și produsul celor prime.  
     
   #define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int \_prime\_check(int n)

{

int p=1;

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

if (n % i == 0) {

p = 0;

}

}

return p;

}

int main(void)

{

int n, nr, s = 0, p = 1;

printf("Urmeaza sa introduceti de la tastatura n numere.\n"

"Ce valoare preferati pentru n ?\n\n"

"n : ");

scanf("%d", &n);

printf("\n");

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

printf("[%d] : ", i); scanf("%d", &nr);

if (nr % 5 == 0) {

s += nr;

}

if (\_prime\_check(nr) == 1) {

p \*= nr;

}

}

printf("\nSuma numerelor divizivile cu 5 : %d\n", s);

printf("Produsul numerelor prime : %d\n", p);

return 0;   
}

1. Să se afișeze toate numerele prime de la 1 la n.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int \_prime\_check(int n)

{

int p = 1;

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

if (n % i == 0) {

p = 0;

}

}

return p;

}

int main(void)

{

int n;

printf("Programul afiseaza toate numerele prime de la 1 la n.\n"

"Ce valoare preferati pentru n?\n\n"

"n : ");

scanf("%d", &n);

printf("\n");

a

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

if (\_prime\_check(i) == 1) {

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

}

}

return 0;

}

1. Se citește de la tastatură un șir de n numere. Să se afișeze câte din acestea sunt mai mici sau egale cu un număr dat k și să se facă suma lor.  
     
     
     
     
   #define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int main(void)

{

int n,k,nr,s=0,c=0;

printf("Urmeaza sa introduceti un sir de n numere.\n"

"Ce valoare preferati pentru n?\n\n"

"n : ");

scanf("%d", &n);

printf("\nProgramul afiseaza cate din nr. introduse sunt mai mici sau egale ca k, si calculeaza suma lor.\n"

"Ce valoare preferati pentru k?\n"

"k : ");

scanf("%d", &k);

printf("\n");

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

{

printf("[%d] : ", i); scanf("%d", &nr);

if (nr <= k) {

c++;

s += nr;

}

}

printf("\n%d din nr. introduse sunt mai mici sau egale ca %d\n", c, k);

printf("Suma numerelor introduse mai mici ca %d : %d\n", k, s);

return 0;

}

1. Să se afișeze pe ecran următoarele secvențe de numere pentru un număr n citit de la tastatură.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int main(void)

{

int n, i, j;

printf("Introduceti o valoare pentru n : "); scanf("%d", &n);

printf("\n");

for (i = n; i >= 1; i--) {

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

printf("%d ", j);

}

printf("\n");

}

printf("\n");

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

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

printf("%d ", j);

}

printf("\n");

}

printf("\n");

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

for (j = n; j >= i; j--) {

printf("%d ", j);

}

printf("\n");

}

printf("\n");

for (i = n; i >= 1; i--) {

for (j = n; j >= i; j--) {

printf("%d ", j);

}

printf("\n");

}

printf("\n");

return 0;

}

1. Se citesc de la tastatură n numere. Să se calculeze pentru fiecare număr produsul cifrelor prime și să se afișeze toate produsele care sunt mai mici decât un număr nr citit anterior.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int \_prime\_check(int n)

{

int p = 1;

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

if (n % i == 0) {

p = 0;

}

}

return p;

}

int main(void)

{

int n, k, nr, d, p = 1;

printf("Programul citeste de la tastatura n numere, dupa care : \n"

"- calculeaza pentru fiecare numar produsul cifrelor prime\n"

"- afiseaza toate produsele mai mici decat un nr. k\n\n"

"Ce valori preferati pentru n si k ?\n"

"n : "); scanf("%d", &n);

printf("k : "); scanf("%d", &k); printf("\n");

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

printf("[%d] : ", i); scanf("%d", &nr);

p = 1;

while (nr != 0) {

d = nr % 10;

if (\_prime\_check(d) == 1) {

p \*= d;

}

nr = nr / 10;

}

printf("Produsul cifrelor prime : %d", p);

if (p < k) {

printf(" < k = %d", k);

}

printf("\n\n");

}

return 0;

}

1. Se citesc numere de tip întreg până se vor introduce 2 numere consecutive. Să se afișeze câte din aceste numere sunt pătrate perfecte.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

#include <math.h>

int \_consecutiveness\_check(int a, int b) {

int c = 0;

if (abs(a - b) == 1)

c = 1;

return c;

}

int main(void) {

int n, c = 0, aux, i = 0;

printf("Programul citeste un sir de numere intregi, pana la introducerea a 2 nr. consecutive.\n"

"Apoi afiseaza cate dintre nr. introduse sunt patrate perfecte.\n\n");

printf("[%d] : ", i); scanf("%d", &aux);

if (sqrt(aux) == aux)

c++;

while (1) {

i++;

printf("[%d] : ", i); scanf("%d", &n);

if (sqrt(n) == n)

c++;

if (\_consecutiveness\_check(aux, n) == 1)

break;

aux = n;

}

printf("\n%d dintre numerele introduse sunt patrate perfecte !\n", c);

return 0;

}

1. Se dă de la tastatură un număr cu cel mult 9 cifre. Să se afișeze cifrele numărului împreună cu frecvența lor de apariție.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int main(void) {

int i = 0, d, n, x[10];

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

x[i] = 0;

printf("Introduceti un numar n (maxim 9 cifre) : "); scanf("%d", &n);

while (n != 0) {

d = n % 10;

x[d]++;

n = n / 10;

}

printf("\nFrecventa fiecarei cifre din numarul introdus : \n\n");

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

if (x[i] != 0)

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

}

printf("\n");

return 0;

}

1. Se citesc numere până la întâlnirea numărului 0. Să se afișeze numerele atunci când al doile număr este egal cu suma cifrelor pare din primul număr.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

int \_even\_digits\_sum(int n) {

int d,sum = 0;

while (n != 0) {

d = n % 10;

n = n / 10;

if (d % 2 == 0)

sum += d;

}

return sum;

}

int main(void) {

int n, i = 0, aux;

printf("Programul citeste un sir de numere pana la introducerea nr. 0.\n"

"Apoi afiseaza numerele atunci cand al doilea nr. = suma cifrelor pare din primul nr.\n\n");

printf("[%d] : ", i); scanf("%d", &aux); i++;

printf("Suma cifrelor pare : %d\n\n", \_even\_digits\_sum(aux));

if (aux != 0) {

while (1) {

printf("[%d] : ", i); scanf("%d", &n);

if (n == 0)

break;

printf("Suma cifrelor pare : %d\n\n", \_even\_digits\_sum(n));

if (\_even\_digits\_sum(aux) == n)

printf("Suma cifrelor pare a nr. %d = %d\n\n", aux, n);

aux = n;

i++;

}

}

return 0;

}