

## Instrucțiuni – exerciții rezolvate recapitulative

1) Sa se scrie un program care executa operatia aritmetica dorita (+,-,/,\*).

```
#include<stdio.h>
void main()
{
    int a,b;
        printf("Introduceti a: ");
        scanf("%d", &a);
        printf("Introduceti b: ");
        scanf("%d", &b);
        char x;
        getchar();
        printf("Introduceti operatia dorita (+ - * /): ");
        scanf("%c", &x);
        switch(x)
        {
            case '+': printf("a+b= %d", a+b); break;
            case '-': printf("a-b= %d", a-b); break;
            case '/': printf("a/b=%.2f",a/(float)b); break;
            case '*': printf("a*b= %d", a*b); break;
            default: printf("Operatie necunoscuta");
        }
        getchar();
        int k;
        scanf("%d", k);
    }
}
```

2) Sa se scrie un program care calculeza suma primelor n numere naturale cu "for".

```
#include<stdio.h>
void main()
{
    int n,s=0;
    printf("Introduceti n:");
    scanf("%d",&n);
    for(int i=1;i<=n;i++)
        s+=i; //s=s+i;
    printf("Suma primelor %d numere naturale este: %d\n",n,s);

    getchar();
    int k;
    scanf("%d", k);
}
}
```

3) Sa se scrie un program care calculeza suma primelor n numere naturale cu "while".

```
#include<stdio.h>

void main()
{
    int n, s=0, i=1;
    printf("Introduceti n:");
    scanf("%d",&n);

    while(i<=n)
    {
        s+=i; //s=s+i;
    }
}
```

```

        i++;
    }

    printf("Suma primelor %d numere naturale este: %d\n",n,s);

    getchar();
    int k;
    scanf("%d", k);
}

```

4) Sa se scrie un program care calculeaza suma primelor n numere naturale cu "do while".

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

```

void main()
{
    int n, s=0, i=1;
    printf("Introduceti n:");
    scanf("%d",&n);

do
{
    s+=i;    //s=s+i;
    i++;
}
while(i<=n)

    printf("Suma primelor %d numere naturale este: %d\n",n,s);

    getchar();
    int k;
    scanf("%d", k);
}

```

5) Sa se scrie un program care rezolva ecuatia de gradul al II-lea.

```

#include <stdio.h>
#include <math.h>
void main (void)
{
    int a,b,c;
    float x1,x2,d;
    printf("Introduceti coeficientii ecuatiei.\n");
    printf("\ta="); scanf("%d",&a);
    printf("\tb="); scanf("%d",&b);
    printf("\tc="); scanf("%d",&c);
    if(!a) //a==0
        if(!b)
            if(!c)
                printf("Ecuatia are o infinitate de
solutii.");
            else
                printf("Ecuatia nu are nici o solutie.");
        else
            {
                x1=-c/(float)b;
                printf("Solutia ec. este: %.2f.",x1);
            }
    else
        x1=-c/(float)b;
        printf("Solutia ec. este: %.2f.",x1);
}

```

```

        {
            d=b*b-4*a*c;
            if(d<0)
                printf("Ec nu are solutii reale.");
            else
                {
                    x1=(-b-sqrt(d))/2*a;
                    printf("Solutiile sunt:\n\tx1 este:
%.2f\n",x1);

                    x2=(-b+sqrt(d))/2*a;
                    printf("\tx2 este: %.2f\n",x2);
                }
        }

    getchar();
    int k;
    scanf("%d", k);
}

```

6) sa se scrie un program care calculeaza valoarea expresiei  $S=1-2+3-4+\dots+$ , folosind functii proprii.

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

```

int suma1(int n) //cu instructiunea FOR
{
    int s=0, i=1, semn=1;
    for(; i<=n; i++)
    {
        s+=semn*i; //s=s+semn*i
        semn=-semn;
    }
    return s;
}

int suma2(int n) //cu instructiunea WHILE
{
    int s=0, i=1, semn=1;
    while (i<=n)
    {
        s+=semn*i; //s=s+semn*i
        semn=-semn;
        i++;
    }
    return s;
}

int suma3(int n) //cu instructiunea DO WHILE
{
    int s=0, i=1, semn=1;
    do
    {
        s+=semn*i; //s=s+semn*i
        semn=-semn;
        i++;
    }
    while (i<=n);
    return s;
}

```

```

void main()
{
    int n;
    printf("Introduceti n:");
    scanf("%d", &n);

    printf("\n\tSuma calculata cu FOR are valoarea: %d" , suma1(n));
    printf("\n\tSuma calculata cu WHILE are valoarea: %d" , suma2(n));
    printf("\n\tSuma calculata cu DO WHILE are valoarea: %d" ,
suma3(n));

    getchar();
    int k;
    scanf("%d", k);
}

```

7) Sa se scrie un program care calculeaza valoarea expresiei:  $P=1*2*3*4*...*n$ .

```

#include<stdio.h>
int produs1(int n)                //cu instructiunea FOR
{
    int P=1;
    for(int i=1; i<=n; i++)
        P=P*i;
    return P;
}
int produs2(int n)                //cu instructiunea WHILE
{
    int P=1; int i=1;
    while(i<=n)
    {
        P=P*i;
        i++;
    }
    return P;
}
int produs3(int n)                //cu instructiunea DO WHILE
{
    int P=1; int i=1;
    do{
        P=P*i;
        i++;
    }
    while(i<=n);
    return P;
}
void main()
{
    int n;
    printf("Introduceti n: ");
    scanf("%d", &n);
    printf("\n\tProdusul calculat cu FOR are valoarea: %d.\n" ,
produs1(n));
    printf("\n\tProdusul calculat cu WHILE are valoarea: %d.\n" ,
produs2(n));
    printf("\n\tProdusul calculat cu DO WHILE are valoarea: %d.\n" ,
produs3(n));

    getchar();
    int k;
}

```

```

scanf("%d", k);
}

```

8) Sa se scrie un program care calculeaza valoarea expresiei:  $P=1*(1/2)*3*(1/4)*5*(1/6)*...n$

```

#include<stdio.h>
#include<math.h>
float produs1(int n)          //cu instructiunea FOR
{
    float p=1;
    float z=1;

    for(int i=1; i<=n; i++)
    {
        if (!(i%2))          //se mai putea scrie if(i%2==0) <=> testeaza
dc nu e impar
        {
            z=1/(float)i;
            p=p*z;
        }

        else p=p*i;         //pt impar
    }
    return p;
}
float produs2(int n)          //cu instructiunea WHILE
{
    float p=1;
    float z=1;
    int i=1;
    while(i<=n)
    {
        if (!(i%2))          //se mai putea scrie if(i%2==0) <=> testeaza
dc nu e impar
        {
            z=1/(float)i;
            p=p*z;
        }
        else p=p*i;         //pt impar
        i++;
    }
    return p;
}
float produs3(int n)          //cu instructiunea WHILE
{
    float p=1;
    float z=1;
    int i=1;
    do
    {
        if (!(i%2))          //se mai putea scrie if(i%2==0) <=> testeaza
dc e par
        {
            z=1/(float)i;
            p=p*z;
        }

        else p=p*i;         //pt impar
        i++;
    }
    while(i<=n);
}

```

```

        return p;
    }
void main()
{
    int n;
    printf("Introduceti n: ");
    scanf("%d", &n);

    printf("\nProdusul cu FOR este: %.5f\n", produs1(n));
    printf("\nProdusul cu WHILE este: %.5f\n", produs2(n));
    printf("\nProdusul cu DO WHILE este: %.5f\n", produs3(n));

    getchar();
    int k;
    scanf("%d", k);
}

```

9) Sa se scrie un program care calculeaza valoarea expresiei:

$S=1+1*2+1*2*3+1*2*3*4+1*2*3*4*5+\dots+1*2*\dots*n.$

```

#include<stdio.h>
void main()
{
    int n;
    int s=0;
    int p=1;

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

    for(int i=1; i<=n; i++)
    {
        p=p*i;
        s=s+p;
    }
    printf("Rezutatul este: %d", s);

    getchar();
    int k;
    scanf("%d", k);
}

```

10) Sa se scrie un program care calculeaza x la puterea y (folosind functia pow).

```

#include <math.h>
#include<stdio.h>
void main()
{
    double x, y, z;

    printf("\nIntroduceti coeficientul: ");
    scanf("%lf", &x);
    printf("\nAti introdus x= %.2lf", x);

    printf("\n\nIntroduceti puterea: ");
    scanf("%lf", &y);
    printf("\nAti introdus y= %.2lf", y);

    z=pow(x,y);
}

```

```

printf("\n\n%.2lf^%.2lf= %.2lf", x,y,z);

getchar();
int k;
scanf("%d", k);
}

```

11) Se citește un text format din cuvinte ce pot fi despartite prin spațiu sau virgulă. Textul citit se termină cu punct sau enter. Să se contorizeze vocalele și consoanele din acel text.

```

#include<stdio.h>
void main()
{
int v=0;
    int c=0;

    printf("Introduceti textul dorit: ");

    char k;
k=getchar(); //citește un caracter, care va fi primul din text, care va fi
trimis variabilei k

    while(k!='.' && k!='\n')
    {
        if (k!=',' && k!=' ')
            switch (k)
            {
                case 'a':
                case 'A':
                case 'e':
                case 'E':
                case 'i':
                case 'I':
                case 'o':
                case 'O':
                case 'u':
                case 'U': v++;
                break;
                default: c++;
            }
        k=getchar(); //urmatorul caracter din text va fi trimis in
variabila k
    }
    printf("Numarul de vocale este: %u\n",v);
    printf("Numarul de consoane este: %u\n",c);

    getchar();
    int z;
    scanf("%d", z);
}

```

12) Se citește un text format din cuvinte ce pot fi despartite prin spațiu sau virgulă. Textul citit se termină cu punct sau enter. Să se contorizeze vocalele și consoanele din acel text. Să se rezolve problema folosind o buclă repetitivă condiționată posterior.

```

#include<stdio.h>
void main()
{
    int v=0;

```

```
int c=0;
printf("Introduceti textul dorit: ");
char k;

do
{
    k=getchar();
    if (k!=',' && k!=' ' && k!='\n' && k!='.')
    switch (k)
    {
        case 'a':
        case 'A':
        case 'e':
        case 'E':
        case 'i':
        case 'I':
        case 'o':
        case 'O':
        case 'u':
        case 'U': v++;
        break;
        default: c++;
    }
}

while(k!='.' && k!='\n');

printf("Numarul de vocale este %u\n",v);
printf("Numarul de consoane este %u\n",c);

getchar();
int z;
scanf("%d", z);
}
```

13) Sa se contorizeze numarul de vocale si cel de consoane din alfabet, folosind instructiunea de salt "goto".

```
#include<stdio.h>
void main()
{
    int v=0;
    int c=0;
    char litera='A';
    eticheta:
        switch (litera)
        {
            case 'A':
            case 'E':
            case 'I':
            case 'O':
            case 'U': v++;
            break;
            default: c++;
        }
    litera++;
    if(litera<='Z') goto eticheta;
    printf("Numarul de vocale este %d\n",v);
    printf("Numarul de consoane este %d\n",c);
}
```



```

    getchar();
    int var;
    scanf("%d", var);
}

```

14) Sa se afiseze din intregul alfabet doar literele caractere mari pana la 'M'. Bucula se va incheia in mod voit, prin folosirea instructiunii "break".

```

#include<stdio.h>
void main()
{
for(char litera='A'; litera<='Z'; litera++)
{
        if(litera=='M') break;
        printf("%c", litera);
}
getchar();
    int var;
    scanf("%d", var);
}

```

15) Sa se transforme un numar din baza 10 intr-o baza p,  $2 \leq p \leq 10$ .

/\* Vom împarti succesiv pe x (numarul dat) la p, reținând cifrele din baza p într-un numar xpi, care va fi apoi rasturnat, pentru a obtine numarul xp, vazut ca reprezentarea în baza p a lui x. Pentru a nu avea probleme în cazul în care prima cifra a lui xpi ar fi 0, folosim o incrementare cu 1 a cifrelor la transformarea x->xpi, respectiv o decrementare la transformarea xpi->xp. \*/

```

#include<stdio.h>
void main ()
{
int x,xpi,xp,p;
printf ("Dati x in baza 10:");
scanf ("%d",&x);
printf("Dati noua baza p:");
scanf("%d",&p);
//un alt mod de a scrie o instructiune for
for (xpi=0; x; xpi=10*xpi+x%p+1,x=x/p);
for (xp=0; xpi; xp=10*xp+(xpi-1)%10,xpi=xpi/10);
printf("Numarul in baza %d este:%d",p,xp);

    getchar();
    int var;
    scanf("%d", var);
}

```

/\* De remarcat, in acest exemplu, folosirea operatorului "," in cadrul instructiunii for. De pilda, primul for se interpreteaza astfel:

- se initializeaza xpi cu 0;
- cat timp x este diferit de 0 executa:
- xpi devine  $10 \cdot xpi + x \% p + 1$
- x se imparte la p (si se face conversia la int).\*/

16) Sa se realizeze un program care intr-o bucla "while" calculeaza suma cifrelor unui numar introdus de la tastatura si afiseaza inversul lui.

```
#include<stdio.h>
#include<math.h>
void main()
{
    int nr;
    int s=0;           //suma cifrelor
    int inv=0;        //inversul numarului
    printf("Introduceti numarul: ");
    scanf("%d",&nr);
    while (nr)        //sau while(n!=0)
    {
        s=s+nr%10;
        inv=inv*10+nr%10;
        nr/=10;
    }
    /* instructiunea repetitiva de mai sus se mai putea scrie si asa:
    while(ninv*=10,ninv+=n%10,s+=n%10,n/=10); */
    printf("Suma este: %d",s);
    printf("\nInversul este: %d",inv);

    getchar();
    int var;
    scanf("%d", var);
}
```

17) Fiind furnizate 3 numere intregi sa se verifice daca acestea pot fi masurile unui triunghi, si in caz afirmativ sa se calculeze perimetrul.

```
#include<stdio.h>
int triunghi(int a,int b,int c)
{
    if((a>0)&&(b>0)&&(c>0)&&(a<b+c)&&(b<a+c)&&(c<a+b))
        return 1;
    else
        return 0;
}
void main()
{
    int a,b,c,s=0;
    printf("a= ");scanf("%d",&a);
    printf("b= ");scanf("%d",&b);
    printf("c= ");scanf("%d",&c);
    if(triunghi(a,b,c)==1)
    {
        s=a+b+c;
        printf("Perimetrul triunghiului ABC este: %d",s);
    }
    else
    printf("Nu este triunghi");

    getchar();
    int var;
    scanf("%d", var);
}
```

18) Sa se calculeze valoarea functiei f definita pe multimea numerelor reale pentru un x citit:

$$F = \begin{cases} x-3, & \text{ptr. } x < 5 \\ x+1, & \text{ptr. } 5 < x < 25 \\ x-5x+6, & \text{ptr. } x > 25 \end{cases}$$

```
#include<stdio.h>
void main()
{
    int x;
    printf("Introduceti numarul: ");
    scanf("%d",&x);

    if(x>=5&& x<=25)
        printf("F(x)= %d",x+1);
    else
    {
        if(x<5)
            printf("F(x)= %d",x-3);
        else
            printf("F(x)= %d",x-5*x+6);
    }
    getchar();
    int var;
    scanf("&d", var);
}
```

19) Sa se afiseze suma cifrelor unui numar natural dat. Indicatie: se detaseaza fiecare cifra de la sfarsitul numarului calculand restul impartirii numarului la 10; numarul ramas dupa desprinderea cifrei este catul impartirii la 10.

```
#include<stdio.h>
void main()
{
    int n,s=0;
    printf("n= ");scanf("%d",&n);
    while(n!=0)
    {
        s=s+n%10;
        n=n/10;
    }
    printf("Suma cifrelor este: %d",s);

    getchar();
    int var;
    scanf("&d", var);
}
```

20) Sa se calculeze suma  $S=1!+2!+3!+\dots+n!$

```
#include<stdio.h>
int factorial(int n)
{
    int p=1;
    for(int i=1;i<=n;i++)
        p*=i;
    return p;
}
```

```
void main()
{
    int m,i,s=0;
    printf("\n\tIntroduceti nr: ");scanf("%d",&m);
    for(i=1;i<=m;i++)
        s=s+factorial(i);
    printf("\tSuma este: %d",s);
    getchar();
    int var;
    scanf("%d", var);
}
```

21) Sa se calculeze produsul:  $P=2*4*6*8*10*...*n$ , unde  $n=2k$ .

```
#include<stdio.h>
void main()
{
    int n,i,p=1;
    printf("Introduceti nr: ");
    scanf("%d",&n);
    n=n*2;

    for(i=2;i<=n;i++)
    {
        if (!(i%2)) //se mai putea scrie if(i%2==0) <=> testeaza
            p*=i;
    }
    printf("Produsul este: %d",p);
    getchar();
    int var;
    scanf("%d", var);
}
```