
Laboratorul 3 – Expresii

/// 3.2. Să se scrie un program pentru calculul valorii $z = x^y$, unde x și y sunt de tipul `double`. Se va utiliza funcția `pow` din biblioteca `math.h`

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

int main()
{
    double x, y, z;

    printf("Introduceti x: ");
    scanf("%lf", &x);

    printf("Introduceti y: ");
    scanf("%lf", &y);

    z=pow(x, y);

    printf("z = %.2f", z);

    return 0;
}
```

/// 3.3. Să se scrie un program pentru a evidenția deosebirea între împărțirea reală și cea întreagă.

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

int main()
{
    int a, b;

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

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

    printf("Catul impartirii: %d / %d: %d\n", a, b, a/b);
    printf("Impartire reala: %d / %d: %.2f\n", a, b, a/(float)b);

    return 0;
}
```

/// 3.4. Să se scrie un program care citește un unghi în grade sexagesimale și calculează valoarea funcției sinus, cosinus și tangentă. Se vor folosi funcțiile trigonometrice din biblioteca math.h

```
#include <stdio.h>
#include <stdlib.h>
#define M_PI 3.14159265
#include <math.h>

int main()
{
    int g;

    printf("Introduceti unghiul in grade sexagesimale: ");
    scanf("%d", &g);

    printf("Sin(%d) = %.2f", g, sin((g*M_PI)/180.0));

    return 0;
}
```

/// 3.5. Să se scrie un program care citește un număr natural din intervalul [1600, 4900] ce reprezintă un an și verifică dacă este bisect sau nu.

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

int main()
{
    int an;

    printf("Introduceti un an intre 1600 si 4900: ");
    scanf("%d", &an);

    if(an<1600 || an>4900) printf("%d nu este in intervalul [1600, 4900]", an);
    else if(an%4==0 && an%100!=0 || an%400==0) printf("%d este an bisect", an);
    else printf("%d nu este an bisect", an);

    return 0;
}
```

/// 3.6. Folosind expresii condiționale, să se scrie un program care citește valoarea reală a lui x și calculează valoarea funcției:

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

int main()
{
    double x;

    printf("Introduceti o valoare pentru x: ");
    scanf("%lf", &x);

    if(x<-2) printf("f(%.2f) = %.2f", x, x*x+4*x+4);
    else if(x== -2) printf("f(%.2f) = 0", x);
    else printf("f(%.2f) = %.2f", x, x*x+5*x);

    return 0;
}
```

/// 3.7. Să se scrie un program care citește un număr real x , ce reprezintă măsura unui unghi în radiani, și îl transformă în grade, minute și secunde sexagesimale.

```
#include <stdio.h>
#include <stdlib.h>
#define M_PI 3.14159265
#include <math.h>

int main()
{
    double r, grade, minute, secunde;

    printf("Introduceti gradele: ");
    scanf("%lf", &r);

    grade=r*(180/M_PI);
    minute=(grade-(int)grade)*60.0;
    secunde=(minute-(int)minute)*60.0;

    printf("%.2lf grade, %.2lf minute, %.2lf secunde", grade, minute, secunde);

    return 0;
}
```

/// 3.8. Scrieți un program pentru a indica numărul de octeți ocupați în memorie de tipurile de date din limbajul C.

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

int main()
{
    printf("char: %d\nunsigned char: %d\nshort int: %d\nunsigned short int: %d\nint: %d\nunsigned int: %d\nlong long int: %d\nunsigned long long int: %d\nfloat: %d\ndouble: %d\nlong double: %d", sizeof(char), sizeof(unsigned char), sizeof(short int), sizeof(unsigned short int), sizeof(int), sizeof(unsigned int), sizeof(long int), sizeof(long long int), sizeof(unsigned long long int), sizeof(float), sizeof(double), sizeof(long double));
    return 0;
}
```

/// 3.9. Converteți în binar, prin calcul, anul nașterii și anul curent. Arătați cum se reprezintă ca o dată de tip int. Efectuați operațiile de deplasare stânga cu 4 biți, dreapta cu 2 biți, complement față de 1 asupra lor, precum și operațiile pe biți &, ^, |, având ca operanzi cele două date. Scrieți un program pentru a verifica corectitudinea calculelor dumneavoastră.

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

int main()
{
    int an1, an2;

    printf("Introduceti anul nasterii: ");
    scanf("%d", &an1);

    printf("Introduceti anul curent: ");
    scanf("%d", &an2);

    printf("Anul nasterii in hexa: %x\nAnul curent in hexa: %x\n\n", an1, an2);

    printf("an1<<4 = %x\n", an1<<4);
    printf("an1>>2 = %x\n", an1>>2);

    return 0;
}
```

/// 3.10. Să se scrie un program ce efectuează operații aritmetice asupra două date de tip întreg și real. Executați-l pentru valori care conduc la rezultat în afara limitelor de reprezentare internă. Ce se întâmplă în acest cazuri?

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

int main()
{
    int a = INT_MAX;
    printf("%x / %d\n\n", a, a);

    printf("a+1 = %x / %d\n", a+1, a+1);
    printf("4*a = %x / %d\n", 4*a, 4*a);

    float b = 1e15, c=13-5, d;

    d=b+c;

    printf("%f + %f = %f\n", b, c, d);

    return 0;
}
```

/// 3.11. Să se scrie un program care citește numerele întregi a, b, c, d și afișează valoarea cea mai mare dintre fracțiile a/b și c/d.

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

int main()
{
    int a, b, c, d;

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

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

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

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

    if((double)a/b > (double)c/d) printf("%d/%d", a, b);
    else printf("%d/%d", c, d);

    return 0;
}
```

/// 3.12. Să se scrie un program care determină poziția relativă a unei linii drepte față de un cerc. Programul va citi: coordonatele centrului cercului, raza cercului și coordonatele a două puncte care determină linia.

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

int main()
{
    int xC, yC, r, xA, yA, xB, yB, n;
    double a, b, c, m, dist, d; /// m = panta dreptei

    printf("Cititi coordonatele centrului cercului (x, y): ");
    scanf("(%d, %d)", &xC, &yC);

    printf("Cititi raza cercului: ");
    scanf("%d%c", &r);

    printf("Cititi coordonatele lui A (x, y): ");
    scanf("(%d, %d)%c", &xA, &yA);

    printf("Cititi coordonatele lui B (x, y): ");
    scanf("(%d, %d)", &xB, &yB);

    m=(yB-yA)/(xB-xA);
    n=yA-m*xA;

    dist=abs(m*xC-yC+n)/sqrt(1+m*m);

    if(dist>r) printf("Dreapta este in exteriorul cercului");
    else if(dist==r) printf("Dreapta este tangenta cercului");
    else printf("Dreapta trece prin cerc");

    return 0;
}
```

/// 3.13. Să se scrie un program pentru calculul masei ideale a unei persoane.

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

int main()
{
    int inaltime, varsta;
    double masa_barbat, masa_femeie;

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

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

    masa_barbat=50+0.76*(inaltime-150)+(varsta-20)/4;
    masa_femeie=masa_barbat-10;

    printf("Greutatea ideala pentru barbat: %.2f\n", masa_barbat);
    printf("Greutatea ideala pentru femeie: %.2f", masa_femeie);

    return 0;
}
```

/// 3.14. Să se scrie un program pentru conversia coordonatelor carteziene ale unui punct dat în coordonate polare. În calculul unghiului din coordonatele polare se poate utiliza funcția atan2 din biblioteca math.h

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

int main()
{
    float x, y, tetha, x_pol, y_pol, r;

    printf("Introduceti coordonatele punctului (x, y): ");
    scanf("(%d, %d)", &x, &y);

    r=sqrt(x*x+y*y);
    tetha=atan(y/x);

    x_pol=r*cos(tetha);
    y_pol=r*sin(tetha);

    printf("Coordonatele polare: %.21f(cos(%.21f) + i*sin(%.21f))", r, tetha, tetha);

    return 0;
}
```