

4 Ден

Рекурзија 2

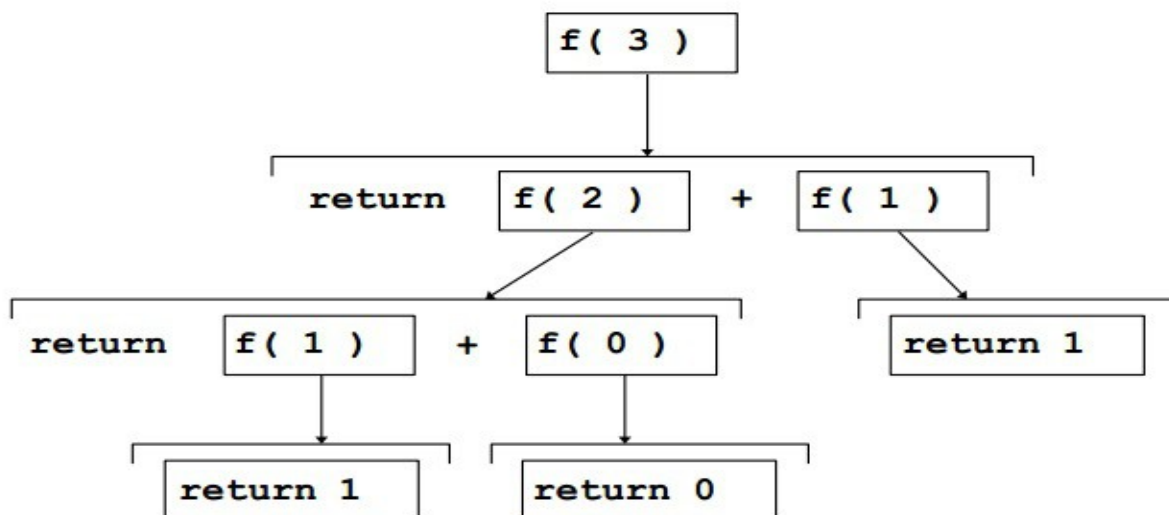
Фибоначиева низа броеви со рекурзија

Фибоначиева низа: 1, 1, 2, 3, 5, 8...

секој број се добива како збир на претходните два

$\text{fib}(n) = \text{fib}(n-1) + \text{fib}(n-2)$ – рекурзивна формула

```
long fibonacci(long n)
{
    if (n==0 || n==1)
        return n;           - граничен случај
    else
        return fibonacci(n-1) + fibonacci(n-2);
}
```



Горниот начин на решавање е добар само за да сфатите како функционира рекурзијата (секој кога се запознава со рекурзија овој пример прв му го преставуваат, но според мене овој пример е многу лош), а инаку како код не е добар бидејќи е многу спор и многу меморија зафаќа. Еве еден пример како може на друг начин да се реши рекурзивно:

```
long fibonacci(long a, long b, int n)
{
    if (n == 1) return b;
    return fibonacci(a+b, a, n-1);
}
```



Напредно програмирање C++

Задача 1

-Да се пресмета збирот:

$1!+2!+3!+\dots+n!$.

$\sum_{(0 \leq i \leq n)} i!$

Решение:

```
#include <iostream>
```

```
using namespace std;
```

```
int faktoriel ( int n)
{
    if (n == 0)
        return 1;
    else
        return n * faktoriel ( n - 1) ;
}
```

```
int suma ( int k)
{
    if (k == 0)
        return 0;
    else
        return faktoriel(k) + suma (k - 1) ;
}
```

```
int main ()
{
    int i , n , rezultat = 0;
    cout << " Vnesete eden pozitiven cel broj " << endl;
    cin >> n;
    if (n > 0)
    {
        cout << " Rezultatot e " << suma(n) ;
    } else
        cout << " Vnesena e pogreshna vrednost " << endl;
    return 0;
}
```



Напредно програмирање C++

Задача 2

-Да се пресмета збирот:

$$1!+(1+2)!+(1+2+3)!+\dots+(1+2+\dots+n)!$$

$$\sum_{(0 \leq i \leq n)} (\sum_{(0 \leq j \leq i)} j)!$$

-Користете рекурзивна функција за пресметување на збирот на првите k природни броеви

-Користете рекурзивна функција за пресметување факториел на еден природен број k

Решение:

```
#include <iostream>
```

```
using namespace std;
```

```
int faktoriel ( int n)
```

```
{  
    if (n == 0)  
        return 1;  
    else  
        return n * faktoriel ( n - 1 );  
}
```

```
int suma ( int k)
```

```
{  
    if (k == 0)  
        return 0;  
    else  
        return k + suma (k - 1) ;  
}
```

```
int main ()
```

```
{  
    int i , n , rezultat = 0;  
    cout << " Vnesete eden pozitiven cel broj " << endl;  
    cin >> n;  
    if (n > 0)  
    {  
        for (i = 1; i < n; i ++)  
        {  
            rezultat += faktoriel ( suma ( i ));  
            cout << suma (i) << " ! + ";  
        }  
        rezultat += faktoriel ( suma ( n ));  
        cout << suma ( n ) << " ! = " << rezultat ;  
    } else  
        cout << " Vnesena e pogreshna vrednost " << endl;  
    return 0;  
}
```



Напредно програмирање C++

Задача 3

- Да се напише програма што ќе ја испишува вредноста на n -тиот член на низата дефинирана со:

$$x_1 = 1$$

$$x_2 = 2$$

•

•

•

$$x_n = ((n-1)/n) * x_{n-1} + (1/n) * x_{n-2}$$

Решение:

```
#include <iostream>
```

```
using namespace std;
```

```
float xnn ( float x1 , float x2 , int n)
{
    if (n == 1)
        return x1 ;
    if (n == 2)
        return x2 ;
    return (n - 1) * xnn ( x1 , x2 , n - 1) / n + xnn ( x1 , x2 , n - 2) / n;
}
```

```
int main ()
{
    int n;
    cout << " Vnesi n : ";
    cin >> n;
    cout << " xnn (1 , 2, " << n << " ) = " << xnn (1 , 2 , n) << endl;
    return 0;
}
```



Напредно програмирање C++

Задача 4

-За даден број n , да се напише рекурзивна функција која ќе ги изброи појавувањата на цифрата 8, со тоа што ако се појави цифра 8 со уште една цифра 8 веднаш лево од неа таа се брои двојно, така 8818 дава 4

Пр:

count8(8) -> 1

count8(818) -> 2

count8(8818) -> 4

Решение:

```
int count8 ( int n )
{
    if (n == 0)
        return 0;
    if ((n / 10) % 10 == 8 && n % 10 == 8)    - или if (n%100 == 88)
        return 2 + count8 (n / 10) ;
    if (n % 10 == 8)
        return 1 + count8 (n / 10) ;
    return count8 ( n / 10) ;
}
```

Задача 5

-Да се пресмета бинарниот број на n .

Пр: $n=14$

$14/2 = 0$ (остатокот се пишува после еднакво)

$7/2 = 1$ (а резултатот се пишува доле)

$3/2 = 1$

$1/2 = 1$

и добиениот број е 1110 (од доле нагоре се земаат битовите)

Решение:

```
int decToBin(int n)
{
    if (n == 0)
        return 0;
    else
        return 10*decToBin(n/2)+n%2;
}
```

За дома, како да го претвориме n во било која основа?



Напредно програмирање C++

Задача 6

-Да се одреди кој збир е поголем, дали на парните цифри или на непарните цифри во n.

Решение 1

```
#include <iostream>

using namespace std;

int zbirParni(int n)
{
    if (n == 0)
        return 0;
    if ((n%10)%2 == 0)
        return n%10+zbirParni(n/10);
    else
        return zbirParni(n/10);
}

int zbirNeparni(int n)
{
    if (n == 0)
        return 0;
    if ((n%10)%2 == 1)
        return n%10+zbirNeparni(n/10);
    else
        return zbirNeparni(n/10);
}

int main ()
{
    int n;
    cin >> n;
    int zP = zbirParni(n);
    int zN = zbirNeparni(n);
    if (zP > zN)
        cout << "Parni" << endl;
    else if (zN > zP)
        cout << "Neparni" << endl;
    else
        cout << "Ednakvi" << endl;
    return 0;
}
```



Напредно програмирање C++

Решение 2 (само со една функција)

```
#include <iostream>

using namespace std;

int zbir(int n)
{
    if (n == 0)
        return 0;
    if ((n%10)%2 == 0)
        return n%10 + zbir(n/10);
    else
        return - n%10 + zbir(n/10);
}

int main ()
{
    int n;
    cin >> n;
    int rez = zbir(n);
    if (n < 0)
        cout << "Neparni";
    else if (n > 0)
        cout << "Parni";
    else
        cout << "Ednakvi";
    return 0;
}
```

За дома, да се одреди кои цифри се побројни, парните или непарните.



Напредно програмирање C++

Задача 7

-Да се напише програма која што за дадена низа од природни броеви која што се внесува од тастатура) ќе го отпечати најголемиот заеднички делител (НЗД) на нејзините елементи. Програмата задолжително треба да содржи рекурзивна функција за пресметување на НЗД на два природни броја.

Решение 1

```
#include <iostream>

using namespace std;

int nzd(int n, int m, int k)
{
    if (n%k == 0 && m%k == 0)
        return k;
    return nzd(n, m, k-1);
}

int main ()
{
    int i, n, a[100];
    cout << " Vnesi ja goleminata na nizata : ";
    cin >> n;
    cout << " Vnesi gi elementite na nizata : " << endl;
    for (i = 0; i < n; i++)
        cin >> a[i ];

    int NZD = nzd (a [0] , a [1], a[0]) ;
    for (i = 2; i < n; i++)
        NZD = nzd ( NZD , a [i ], NZD) ;

    cout << " NZD na elementite na ovaa niza e " << NZD ;
    return 0;
}
```



Напредно програмирање C++

Решение 2 (Евклидов алгоритм)

- НЗД за два броја може да се пресмета со користење на Евклидовиот алгоритм
- За да се пресмета НЗД за броевите m и n , се пресметува остатокот при делење на m со n
 - Ако остатокот не е 0, се пресметува остатокот при делење на n со $m \% n$
 - Постапката се повторува се додека се добиваат ненулни остатоци
 - Ако остатокот е 0, НЗД за двата броја е последниот пресметан ненулни остаток

```
#include <iostream>

using namespace std;

int nzd ( int m , int n )
{
    if (n == 0)
        return m;
    return nzd (n , m % n);
}

int main ()
{
    int i , n , a[100];
    cout << " Vnesi ja goleminata na nizata : ";
    cin >> n;
    cout << " Vnesi gi elementite na nizata : " << endl;
    for (i = 0; i < n; i++)
        cin >> a[i];

    int NZD = nzd (a [0] , a [1]) ;
    for (i = 2; i < n; i++)
        NZD = nzd ( NZD , a [i] );

    cout << " NZD na elementite na ova niza e " << NZD ;
    return 0;
}
```

