

СТРУКТУРА РЕКУРСИВНОГО АЛГОРИТМА

Выполнил студент ИВБО-01-22
Зырянов Максим Алексеевич

РЕКУРСИЯ

Рекурсивная программа - это программа, которая обращается сама к себе



НЕДОСТАТКИ РЕКУРСИИ

```
#include <iostream>
#include <cmath>
#include <cstdlib>
using namespace std;
```

```
int factorial(int a)
```

```
{
```

```
    if (a > 1)
```

```
    {
```

```
        return a * factorial(a - 1);
```

```
    }
```

```
}
```

```
float
```

```
{
```

```
{
```

```
{
```

```
}
```

```
}
```

```
int main()
```

```
{
```

```
    float x;
```

```
    cout << "Nhap x: "<<endl;
```

```
    cin >> x;
```

```
    cout << "E^x = " << series(x,0,0);
```

```
    system("pause");
```

```
    return 0;
```

```
}
```

Exception Unhandled

Unhandled exception at 0x00F02539 in 33b.exe: 0xC00000FD: Stack overflow (parameters: 0x00000001, 0x00F22FF8).

[Copy Details](#)

[Exception Settings](#)

```
pow(x, n) / factorial(n)) == pow(10, -6)) { return b; }
n + 1, b);
```

Переполнение стека (stack overflow)

```
1  #include <iostream>
2
3  using namespace std;
4
5  int fibonacci(int n) {
6      if (n == 1 || n == 2) {
7          return 1;
8      }
9      if (n == 0) {
10         return 0;
11     }
12     return fibonacci(n - 1) + fibonacci(n - 2);
13 }
14
15 int main() {
16     cout << fibonacci(40) << endl;
17     return 0;
18 }
```

Нахождение 40-го числа Фибоначчи с помощью рекурсии

$$T_N = T_{N-1} + T_{N-2}, \text{ при } N \geq 1; T_0 = 1; T_1 = 1$$

Формула для определения сложности рекурсии






ДИНАМИЧЕСКОЕ ПРОГРАММИРОВАНИЕ

```
1 #include <iostream>
2
3 using namespace std;
4
5 int f[100001];
6
7 int main() {
8     int n;
9     cin >> n;
10    n = n - 1;
11    f[1] = f[2] = 1;
12    for (int i = 3; i <= n; i++) {
13        f[i] = f[i - 1] + f[i - 2];
14    }
15    cout << f[n];
16 }
```

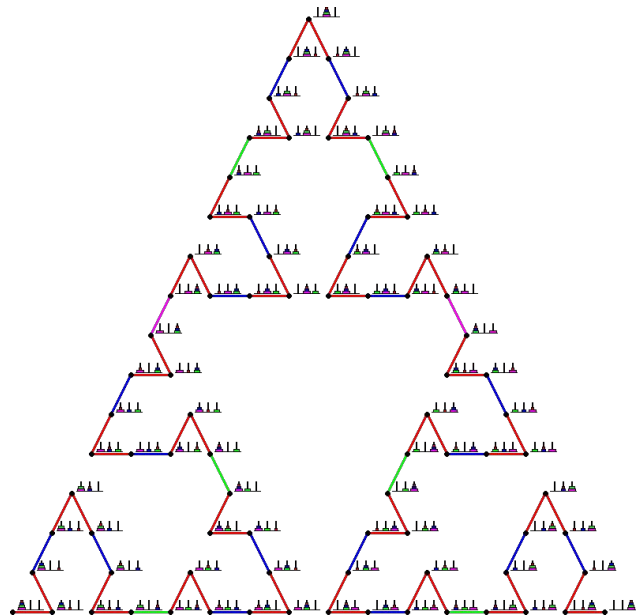
Нахождение чисел Фибоначчи, используя рекурсию и динамическое программирование



РЕКУРСИЯ И ДЕРЕВЬЯ



Ханойские башни



**Ханойские башни,
представленные в виде
дерева**



ЗАКЛЮЧЕНИЕ

