

Programming. Recursion.

1. Дан рекурсивный алгоритм F. Приведите последовательность чисел (без пробелов), напечатанных на экране при выполнении вызова F(4).

```
void F(int n){
    cout << n;
    if(n > 1){
        F(n - 1);
        F(n - 2);
    }
}
```

2. Даны рекурсивные алгоритмы F и G. Чему равно значение функции F(6).

```
int F(int n){
    if(n > 1)
        return F(n - 1) + G(n - 1);
    else
        return n;
}
int G(int n){
    if(n > 2)
        return G(n - 1) + F(n);
    else
        return n;
}
```

3. Алгоритм вычисления значения функции F(n), где n – натуральное число, задан следующими соотношениями:

$$F(n) = F(n - 1) + n - 2, \text{ при } n > 1$$

$$F(1) = 2.$$

Чему равно значение функции F(7)?

4. Алгоритм вычисления значения функции F(n), где n – натуральное число, задан следующими соотношениями:

$$F(n) = F(n - 1) + F(n / 2), \text{ при } n > 1 \text{ и } n \text{ кратно } 2$$

$$F(n) = F(n - 1) + 1, \text{ при } n > 1 \text{ и } n \text{ не кратно } 2$$

$$F(1) = 1$$

Чему равно значение функции F(8)?

5. Дан рекурсивный алгоритм F. Приведите последовательность чисел (без пробелов), напечатанных на экране при выполнении вызова F(9, 0).

```
void F(int n, int k){
    if(k == 2)
        cout << n;
    else {
        F(n / 2, k + 1);
        F(n - 3, k + 1);
    }
}
```

6. Определите, сколько звездочек будет напечатано в результате вызова F(5) приведенной подпрограммы.

```
void F(int n){
    if(n > 1){
        F(n / 2);
        F(n - 1);
    }
    std::cout << "*";
}
```

ЕГЭ. Задание 11. Programming. Recursion.

7. Дан рекурсивный алгоритм F. Приведите последовательность чисел (без пробелов), напечатанных на экране при выполнении вызова F(5).

```
void F(int n){
    if(n > 1){
        F(n / 2);
        cout << n;
        F(n - 1);
    }
}
```

8. Определите, сколько звездочек будет напечатано в результате вызова F(5) приведенной подпрограммы.

```
void F(int n){
    std::cout << "*";
    if(n > 0){
        F(n - 3);
        F(n - 1);
        std::cout << "*";
    }
}
```

9. Даны рекурсивные алгоритмы F и G. Чему равна сумма чисел, напечатанных на экране при выполнении вызова F(15).

```
int F(int n){
    if(n > 0)
        G(n - 2);
}
int G(int n){
    std::cout << n << endl;
    if(n > 1)
        F(n - 1);
}
```

10. Дан рекурсивный алгоритм F. Приведите последовательность чисел (без пробелов), напечатанных на экране при выполнении вызова F(1).

```
void F(int n){
    cout << n;
    if(n < 4){
        F(n + 2);
        F(n + 1);
    }
}
```

11. Даны рекурсивные алгоритмы F и G. Чему равно значение функции G(6).

```
int F(int n){
    if(n > 2)
        return F(n - 1) + G(n - 2);
    else
        return n;
}
int G(int n){
    if(n > 1)
        return G(n - 1) + F(n);
    else
        return n + 1;
}
```

12. Алгоритм вычисления значения функции F(n), где n – натуральное число, задан следующими соотношениями:

$$F(n) = F(n - 1) + F(n / 2), \text{ при } n > 1 \text{ и } n \text{ кратно } 2$$

$$F(n) = F(n - 1) + F(n - 2), \text{ при } n > 1 \text{ и } n \text{ не кратно } 2$$

$$F(1) = 1$$

Чему равно значение функции F(8)?

- 13.** Определите, сколько звездочек будет напечатано в результате вызова F(3) приведенной подпрограммы.

```
void F(int n){
    if(n > 0){
        F(n - 2);
        F(n - 1);
        F(n - 1);
    }
    std::cout << "*";
}

```

- 14.** Определите, сколько звездочек будет напечатано в результате вызова F(20)

```
void F(int n);
void G(int n);
void F(int n){
    if(n > 0){
        G(n - 1);
    }
}
void G(int n){
    std::cout << "*";
    if(n > 1)
        F(n - 2);
}

```

- 15.** Дан рекурсивный алгоритм F. Приведите последовательность чисел (без пробелов), напечатанных на экране при выполнении вызова F(7).

```
void F(int n){
    if(n > 2){
        F(n / 2);
        cout << n;
        F(n / 3);
    }
    cout << n;
}

```

- 16.** Дан рекурсивный алгоритм F. Чему равна сумма чисел, напечатанных на экране, при выполнении вызова F(1).

```
void F(int n){
    if(n < 5){
        cout << n;
        F(n + 1);
        F(n + 2);
        F(n + 3);
    }
}

```

- 17.** Дан рекурсивный алгоритм F. Чему равно последнее число, напечатанное при вызове F(9).

```
int F(int n){
    int t;
    if(n < 0) t = 1;
    else{
        t = F(n - 2) + n;
        cout << t << " ";
    }
    return t;
}

```

ЕГЭ. Задание 11. Programming. Recursion.

18. Дан рекурсивный алгоритм F. Чему равно последнее число, напечатанное при вызове F(9).

```
int F(int n){
    int t;
    if(n <= 0)
        t = 1;
    else
        if(n % 2 == 0)
            t = F(n - 1);
        else
            t = F(n - 2) * 2;
    std::cout << t << " ";
    return t;
}
```

19. Чему будет равно значение, вычисленное алгоритмом при выполнении вызова F(6)?

```
int F(int n) {
    if (n > 1)
        return 2 * n + F(n - 3) + F(n - 2);
    else
        return n + 5;
}
```

20. Найдите сумму чисел, которые будут выведены при вызове F(2).

```
void F(int n) {
    std::cout << n;
    if (n < 6) {
        std::cout << n;
        F(n + 1);
        F(n * 2);
        F(n * 3);
    }
}
```

21. Сколько символов "звездочка" будет напечатано на экране при выполнении вызова F(6)?

```
void F(int n) {
    std::cout << "*";
    if (n > 0) {
        F(n - 3);
        F(n - 2);
        F(n / 2);
        F(n / 2);
    }
}
```

22. Дан рекурсивный алгоритм F. Приведите последовательность чисел (без пробелов), напечатанных на экране при выполнении вызова F(4).

```
void F(int n){
    cout << n;
    if(n > 1){
        F(n / 2);
        F(n - 1);
    }
}
```