

Отчёт по “ПЗ 6”

ТРПО

Чистые функции

Задания из пз 3:

1. Посчитать площадь и периметр прямоугольника. Нарисовать получившийся прямоугольник.
2. Ввести число n ($n > 0$). Вывести первые n чисел Фибоначчи. Посчитать факториал числа n .
3. Ввести два числа, определить являются ли они простыми. Найти наибольший общий делитель.
4. Вывести на экран двоичное представление числа.

Разделяю всё на функции, меняю названия переменных.

```
void Perimetr_Square_Painting_OfRectangle(int length, int width, char sign)
{
    cout << "Площадь прямоугольника равна " << SquareOfRectangle(length, width) << endl;
    cout << "Периметр прямоугольника равен " << PerimeterOfRectangle(length, width) << endl;
    DrawingRectangle(length, width, sign);
}

void Factorial_Fibonacci(int numb1)
{
    cout << "Факториал равен " << Factorial(numb1) << endl;
    cout << "Последовательность Фибоначчи: ";
    Fibonacci(numb1);
}

void Simple_Compound_LargestCommonFactor(int numb3, int numb4)
{
    Simple_Compound(numb3);
    Simple_Compound(numb4);
    LargestCommonFactor(numb3, numb4);
}

void Represent_Of_Numb(int numb2, string a)
{
    cout << RepresentOfNumber(numb2, a) << endl;
}
```

Подфункции, получается

Выделяю подфункции.

1-ое задание:

```
int SquareOfRectangle(int length, int width)
{
    return length * width;
}
int PerimeterOfRectangle(int length, int width)
{
    return (length + width) * 2;
}
void DrawingRectangle(int length, int width, char sign)
{
    for (int i = 0; i < width; i++)
    {
        for (int j = 0; j < length; j++)
        {
            cout << sign;
        }
        cout << endl;
    }
}
```

2-ое задание:

```
int Factorial(int numb1)
{
    int fact = 1;
    for (int i = 1; i <= numb1; i++)
    {
        fact *= i;
    }
    return fact;
}
void Fibonacci(int numb1)
{
    int a = 1, b = -1, c = 0;
    for (int i = 0; i < numb1; i++)
    {
        c = b + a;
        b = a;
        a = c;
        cout << c << " ";
    }
}
```

3-е задание:

```
void Simple_Compound(int numb)
{
    int del = 1, sum = 0;
    for (int i = 2; i <= (numb / 2); i++)
    {
        del = numb % i;
        if (del == 0)
        {
            sum++;
        }
    }
    if (sum == 0)
    {
        cout << numb << " - простое" << endl;
    }
    else
    {
        cout << numb << " - не простое" << endl;
    }
}

int LargestCommonFactor(int numb3, int numb4)
{
    int i = 0;
    for (int i = numb3; i > 0; i--)
    {
        if (numb3 % i == 0 && numb4 % i == 0)
        {
            cout << "НОД = " << i;
            break;
        }
    }
}
```

4-ое задание:

```
string RepresentOfNumber(int numb2, string a)
{
    while (numb2 > 0) {
        if (numb2 % 2 == 1)
        {
            a = a + "1";
            numb2 = numb2 / 2;
        }
        else
        {
            a = a + "0";
            numb2 = numb2 / 2;
        }
    }
    reverse(a.begin(), a.end());
    return a;
}
```

Вынесение побочных эффектов и преобразование чистых функций

Избавляюсь от побочных эффектов и все ф-ции
вызываю в main:

```
int main()
{
    setlocale(LC_ALL, "RUS");

    cout << "Задание 1\n";
    int length, width;
    char sign;
    cout << "Введите длину ";
    cin >> length;
    cout << "Введите ширину ";
    cin >> width;
    cout << "Введите символ ";
    cin >> sign;
    Perimetr_Square_Painting_OfRectangle(length, width, sign);
    cout << endl;

    cout << "Задание 2\n";
    int numb1;
    cout << "Введите число ";
    cin >> numb1;
    Factorial_Fibonacci(numb1);
    cout << endl;

    cout << "Задание 3\n";
    int numb3, numb4;
    cout << "Введите первое число \n";
    cin >> numb3;
    cout << "Введите второе число \n";
    cin >> numb4;
    Simple_Compound_LargestCommonFactor(numb3, numb4);
    cout << endl;

    cout << "Задание 4\n";
    int numb2;
    string a = "";
    cout << "Введите число ";
    cin >> numb2;
    Represent_Of_Numb(numb2, a);
    cout << endl;
    return 0;
}
```

А исходный код заданий тутъ:

1-ое задание:

```
int main()
{
    setlocale(LC_ALL, "rus");
    int a, b, i, j;
    char c;
    cout << "Введите сторону a ";
    cin >> a;
    cout << "Введите сторону b ";
    cin >> b;
    cout << "Площадь равна " << a * b << endl;
    cout << "Периметр равна " << (a + b) * 2 << endl;
    cout << "Введите символ ";
    cin >> c;
    for (i = 0; i < b; i++)
    {
        for (j = 0; j < a; j++)
        {
            cout << c;
        }
        cout << endl;
    }
    return 0;
}
```

2-ое задание:

```
int main()
{
    setlocale(LC_ALL, "rus");
    int n, a, i, f = 1, b = 1, c = -1, d;
    cout << "Введите число n ";
    cin >> n;
    for (a = 1; a <= n; a++)
    {
        f *= a;
    }
    cout << "факториал равен " << f << endl;
    for (i = 0; i < n; i++)
    {
        d = c + b;
        c = b;
        b = d;
        cout << d;
    }
    return 0;
}
```

3-е задание:

```
int main() {
    setlocale(LC_ALL, "rus");
    int a, b, c, del1, del2, sum1 = 0, sum2 = 0;
    cout << "Введите два числа \n";
    cin >> a;
    cin >> b;

    for (c = 2; c <= (a / 2); c++)
    {
        del1 = a % c;
        if (del1 == 0)
        {
            sum1++;
        }
    }
    if (sum1 == 0)
    {
        cout << a << " - простое" << endl;
    }
    else
    {
        cout << a << " - не простое" << endl;
    }
    for (int d = 2; d <= (b / 2); d++)
    {
        del2 = b % d;
        if (del2 == 0)
        {
            sum2++;
        }
    }
    if (sum2 == 0)
    {
        cout << b << " - простое" << endl;
    }
    else
    {
        cout << b << " - не простое" << endl;
    }

    for (int i = a; i > 0; i--)
    {
        if (a % i == 0 && b % i == 0)
        {
            cout << "nod = " << i;
            break;
        }
    }
    return 0;
}
```

4-ое задание:

```
#include <iostream>
#include <math.h>
#include <string>
#include <random>

using namespace std;

int main() {
    setlocale(LC_ALL, "rus");

    cout << "Введите число: ";
    int n;
    string number;

    cin >> n;
    while (n > 0)
    {
        number += to_string(n % 2);
        n /= 2;
    }
    for (int i = number.length() - 1; i >= 0; i--)
    {
        cout << number[i];
    }

    return 0;
}
```

Varvashenya Alyona, 114301