

Министерство образования Республики Беларусь

Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Лабораторная работа № 2  
«Операторы цикла»

Проверил:  
Богдан Е.В.

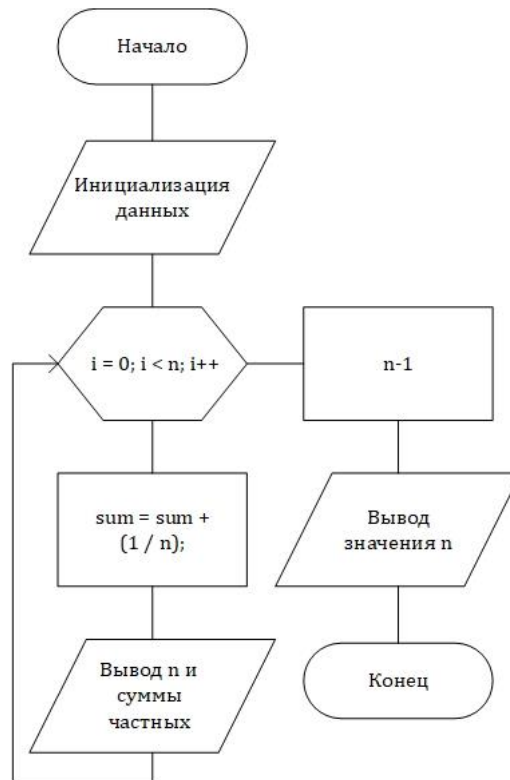
Выполнил:  
Бекетова М.А.

МИНСК 2022

## Задача 1

1. Сколько слагаемых должно быть в сумме  $1+1/2+1/3+1/4+\dots+1/n$ , чтобы эта сумма оказалась больше 5 ?


Блок-схема:




Код программы:

```
void task1()
{
    float sum = 0;
    float n;
    for (n = 1; sum <= 5; n++)
    {
        sum = sum + (1 / n);
        printf("n = %.f sum=%f\n", n, sum);
    }
    n--;
    printf("\n The number of elements of the sum is equal to the value of the last divisor\n So n = %.f\n", n);
}
```


## Вывод программы:

 D:\vs\LAB\_2\x64\Debug\LAB\_2.exe

```
n = 1 sum=1.000000
n = 2 sum=1.500000
n = 3 sum=1.833333
n = 4 sum=2.083333
n = 5 sum=2.283334
n = 6 sum=2.450000
n = 7 sum=2.592857
n = 8 sum=2.717857
n = 9 sum=2.828969
n = 10 sum=2.928968
n = 11 sum=3.019877
n = 12 sum=3.103211
n = 13 sum=3.180134
n = 14 sum=3.251562
n = 15 sum=3.318229
n = 16 sum=3.380729
n = 17 sum=3.439553
n = 18 sum=3.495108
n = 19 sum=3.547740
n = 20 sum=3.597740
n = 21 sum=3.645359
n = 22 sum=3.690813
n = 23 sum=3.734292
n = 24 sum=3.775958
n = 25 sum=3.815958
n = 26 sum=3.854420
n = 27 sum=3.891457
n = 28 sum=3.927171
n = 29 sum=3.961654
n = 30 sum=3.994987
n = 31 sum=4.027246
n = 32 sum=4.058496
n = 33 sum=4.088799
n = 34 sum=4.118210
n = 35 sum=4.146782
n = 36 sum=4.174560
n = 37 sum=4.201587
n = 38 sum=4.227902
n = 39 sum=4.253543
n = 40 sum=4.278543
```

 D:\vs\LAB\_2\x64\Debug\LAB\_2.exe

```
n = 41 sum=4.302934
n = 42 sum=4.326743
n = 43 sum=4.349999
n = 44 sum=4.372726
n = 45 sum=4.394948
n = 46 sum=4.416687
n = 47 sum=4.437964
n = 48 sum=4.458797
n = 49 sum=4.479206
n = 50 sum=4.499206
n = 51 sum=4.518814
n = 52 sum=4.538044
n = 53 sum=4.556912
n = 54 sum=4.575431
n = 55 sum=4.593613
n = 56 sum=4.611470
n = 57 sum=4.629014
n = 58 sum=4.646255
n = 59 sum=4.663204
n = 60 sum=4.679871
n = 61 sum=4.696265
n = 62 sum=4.712394
n = 63 sum=4.728267
n = 64 sum=4.743892
n = 65 sum=4.759276
n = 66 sum=4.774428
n = 67 sum=4.789353
n = 68 sum=4.804059
n = 69 sum=4.818552
n = 70 sum=4.832838
n = 71 sum=4.846922
n = 72 sum=4.860811
n = 73 sum=4.874509
n = 74 sum=4.888023
n = 75 sum=4.901356
n = 76 sum=4.914514
n = 77 sum=4.927501
n = 78 sum=4.940322
n = 79 sum=4.952980
n = 80 sum=4.965480
```

 D:\vs\LAB\_2\x64\Debug\LAB\_2.exe

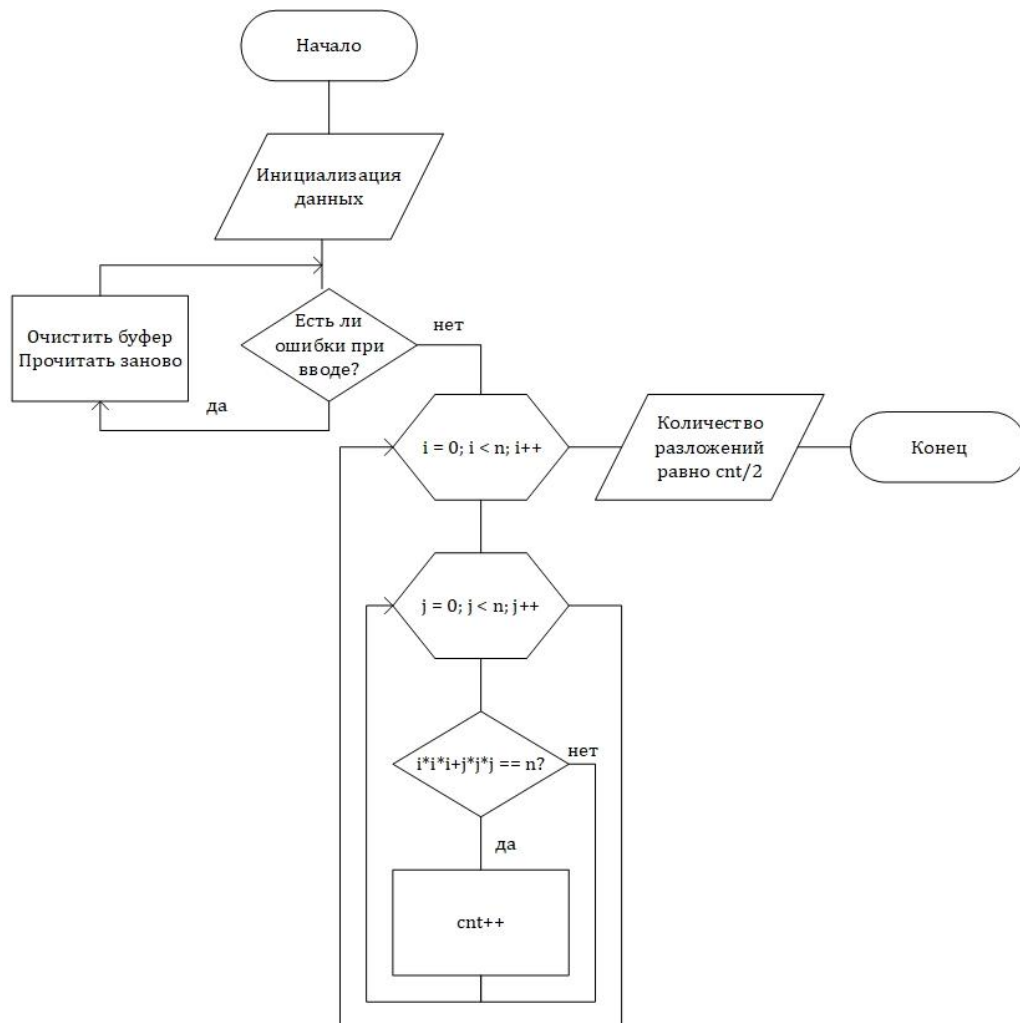
```
n = 81 sum=4.977826
n = 82 sum=4.990021
n = 83 sum=5.002069
```

```
The number of elements of the sum is equal to the value of the last divisor
So n = 83
Enter the number of the task (from 1 to 3), or enter '0' to end the program
```

## Задача 2

1. Составить алгоритм, определяющий количество способов, какими задуманное число  $n > 1$  можно представить в виде суммы  $n = i^3 + j^3$ , считая, что перестановка слагаемых нового способа не даёт.

## Блок-схема:



## Код программы:

```
void task2 ()
{
    int n, i, j;
    int cnt = 0;
    printf("Enter the number n>1\n");
    while (scanf_s("%i", &n) != 1) //проверка на ввод
    {
        while (getchar() != '\n')
            continue;
        printf("Just enter the number higher then 1\n ");
    }
    for (i = 0; i < n; i++)
    {
        for (j = 0; j < n; j++)
        {
            if (i * i * i + j * j * j == n)
                cnt++;
        }
    }
    printf("Ways to represent a number: %i\v", cnt/2);
}
```

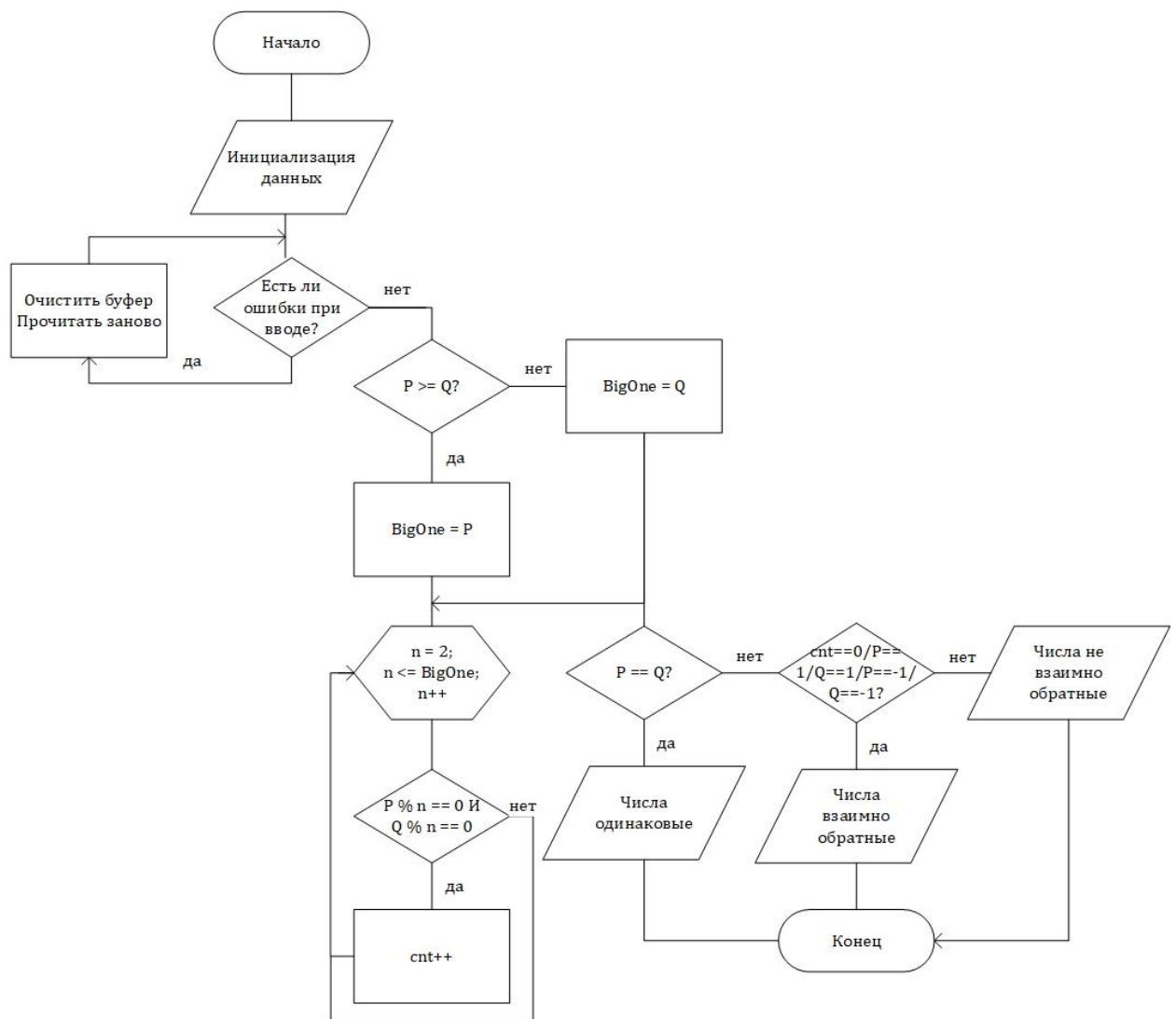
## Вывод программы:

cmd	D:\vs\LAB_2\x64\Debug\LAB_2.exe	cmd	Выбрать D:\vs\LAB_2\x64\Debug\LAB_2.exe	cmd	D:\vs\LAB_2\x64\Debug\LAB_2.exe
Enter the number n>1	Enter the number n>1	Enter the number n>1		Enter the number n>1	
8	1729	45			
Ways to represent a number: 1	Ways to represent a number: 2	Ways to represent a number: 0			

## Задача 3

1. Проверить являются ли числа P и Q взаимно простыми.

## Блок-схема:



## Код программы:




```
void task3()
{
    int P, Q, n, BigOne;
    int cnt = 0;
```

```

printf("Enter P, Q\n");
while (scanf_s("%d%d", &P, &Q) != 2) // проверка на ввод
{
    while (getchar() != '\n')
        continue;
    printf("Incorrect input\n");
}
if (P >= Q)
    BigOne = P;
else
    BigOne = Q;
for (n = 2; n <= BigOne; n++)
{
    if (P % n == 0 && Q % n == 0)
        cnt++;
}
if (P == Q)
{
    printf("Numbers are the same\n");
}
else if (cnt == 0 || P == 1 || Q == 1 || P == -1 || Q == -1)
{
    printf("Numbers are mutually prime\n");
}
else
{
    printf("Numbers are not mutually prime\n");
}
}

```

### Вывод программы:

 D:\vs\LAB_2\x64\Debug\LAB_2.exe Enter P, Q 56 71 Numbers are mutually prime	 D:\vs\LAB_2\x64\Debug\LAB_2.exe Enter P, Q 48 22 Numbers are not mutually prime	 D:\vs\LAB_2\x64\Debug\LAB_2.exe Enter P, Q 6 6 Numbers are the same
--	--	--