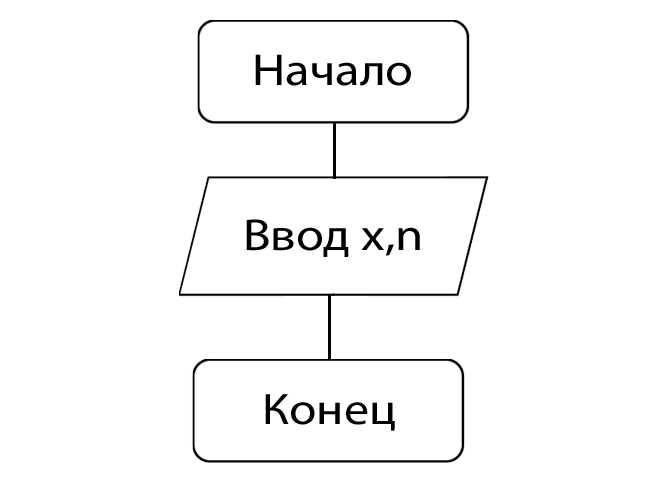
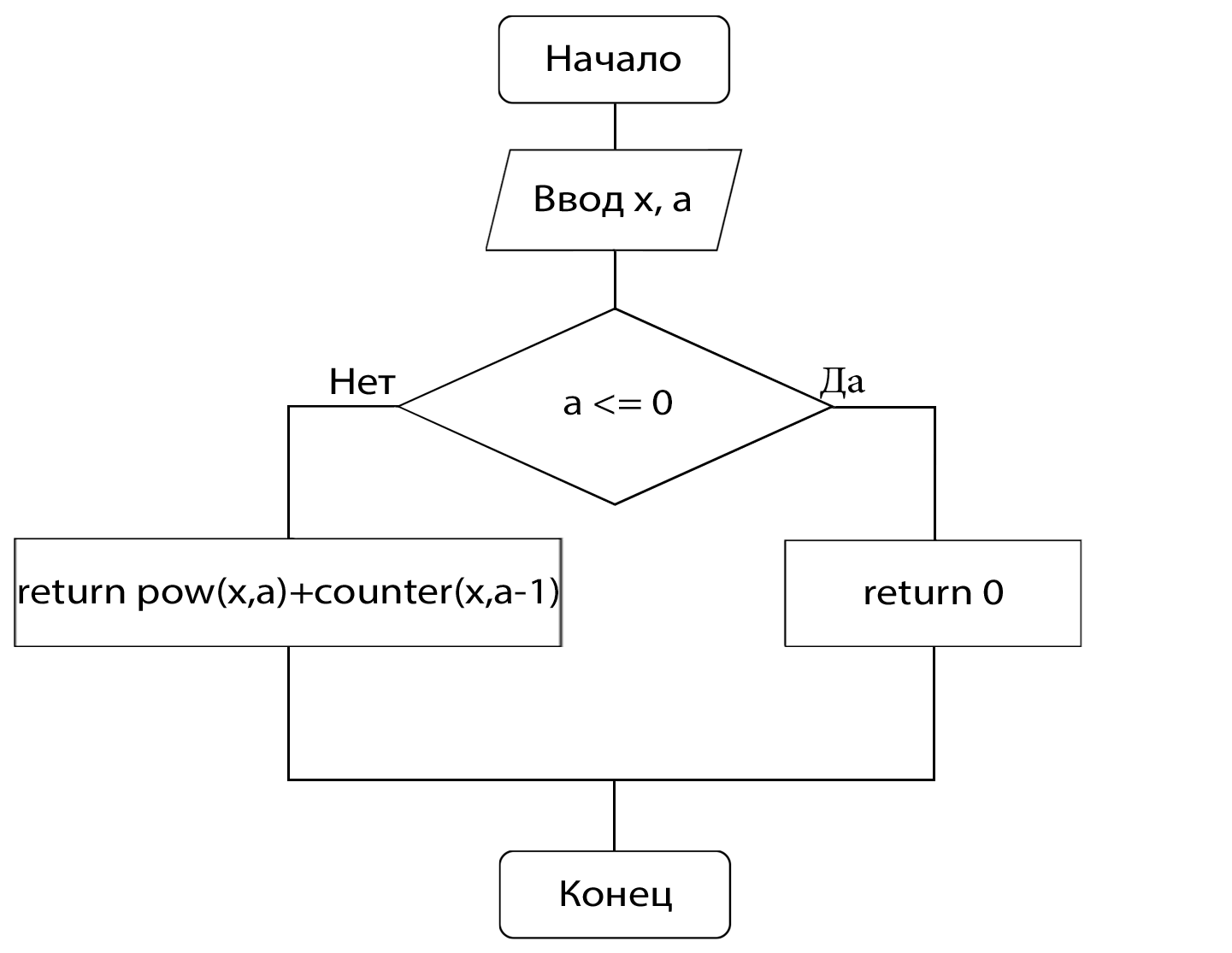
Задание №1

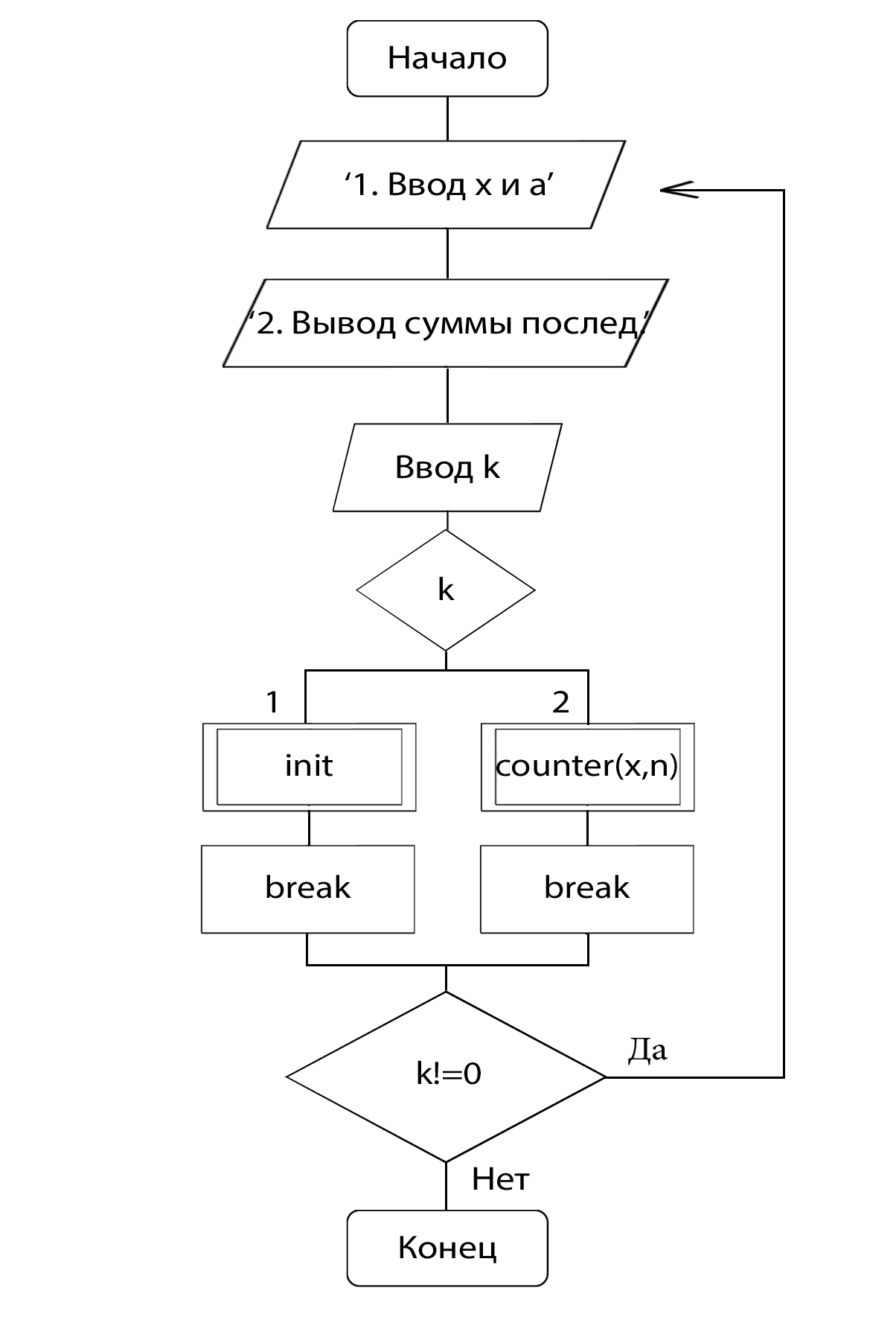
Условие:

S = x1 + x2 + ... + xn

Алгоритм функции init:



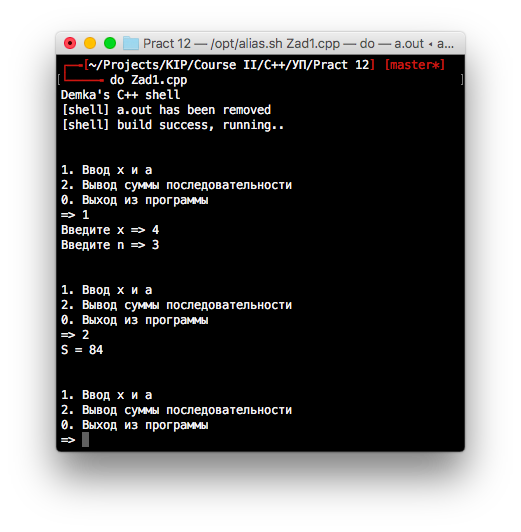
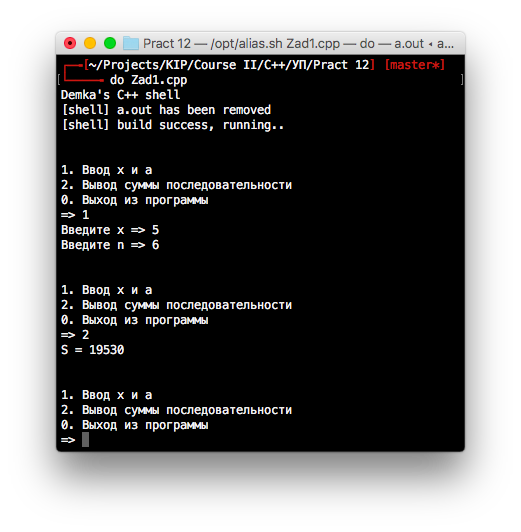
Алгоритм функции counter:

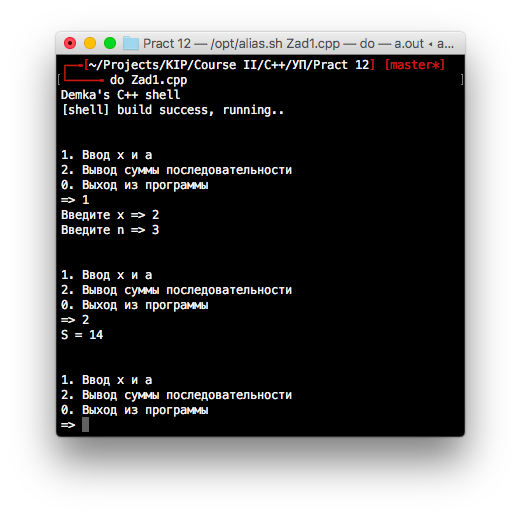
Алгоритм основной программы:  


Исходный код программы:

#include **<iostream>**#include **<cmath>  
   
using namespace** std;  
  
**int** x, n, k;  
  
**void** init(){  
 cout<<**"Введите x => "**; cin>>x;  
 cout<<**"Введите n => "**; cin>>n;  
}  
  
**int** counter(**int** x,**int** a){  
 **if** (a <= 0)  
 **return** 0;  
 **else  
 return** pow(x,a) + counter(x,a-1);  
}  
  
**int** main(){  
 **do** {  
 cout<<**"\n\n1. Ввод x и a\n2. Вывод суммы последовательности\n0. Выход из программы\n=> "**;  
 cin>>k;  
 **switch** (k)   
 {   
 **case** 1:   
 init();  
 **break**;  
   
 **case** 2:   
 cout<<**"S = "**<<counter(x,n)<<**"\n"**;  
 **break**;  
 }   
  
 }  
 **while** (k!=0);  
  
cout<<**"\n"**;  
**return** 0;  
}

Скриншоты программы:





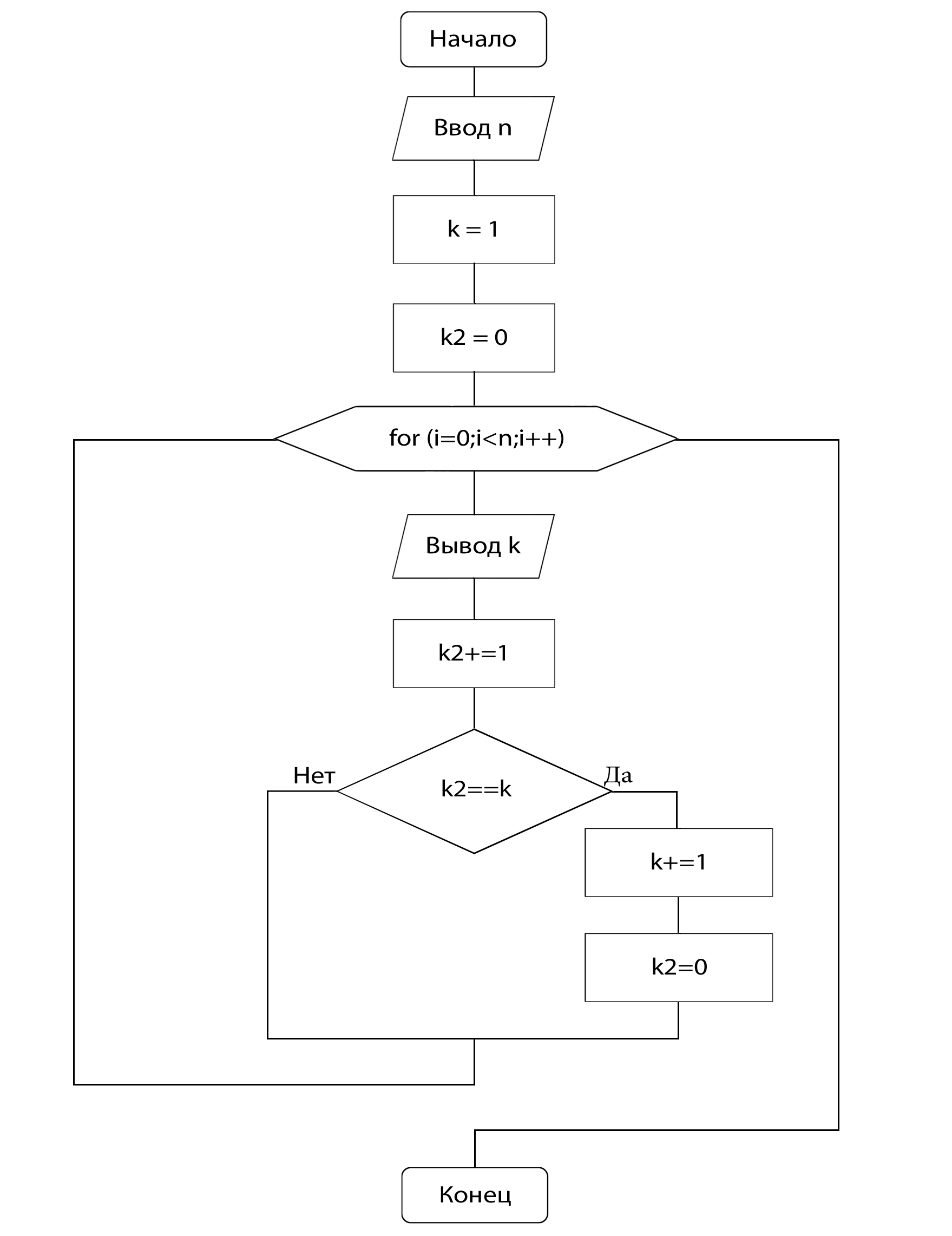
Задание №2

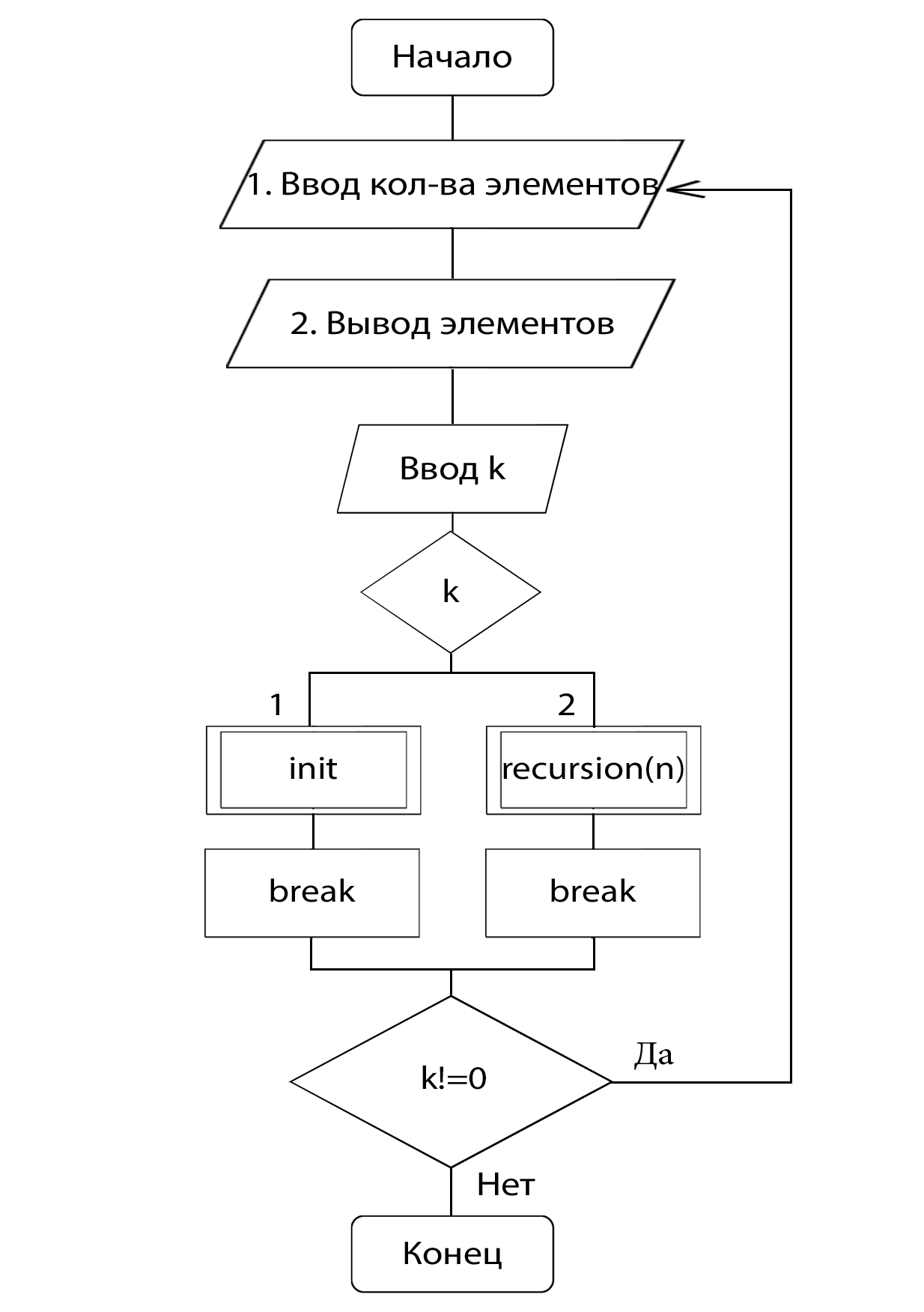
Условие:  
Дана монотонная последовательность, в которой каждое натуральное число k встречается ровно k раз: 1, 2, 2, 3, 3, 3, 4, 4, 4, 4, ...

По данному натуральному n выведите первые n членов этой последовательности. Попробуйте обойтись только одним циклом for.

Алгоритм функции init:



Алгоритм функции recursion:

Алгоритм основной программы:

Исходный код программы:

#include **<iostream>**#include **<ctime>  
using namespace** std;  
  
**int** n, i, k;  
  
**void** init(){  
 cout<<**"Введите n => "**; cin>>n;  
}  
  
**void** recursion(**int** n){  
 **int** k = 1;  
 **int** k2 = 0;  
 cout<<**"Первые "**<<n<<**" элементов последовательности:\n"**;  
 **for** (i=0;i<n;i++){  
 cout<<k<<**' '**;  
 k2+=1;  
 **if** (k2==k){  
 k+=1;  
 k2=0;  
 }  
 }  
}  
  
  
**int** main(){  
  
 **do** {  
 cout<<**"\n\n1. Ввод кол-ва элементов последовательности\n2. Вывод элементов последовательности\n0. Выход из программы\n=> "**;  
 cin>>k;  
 **switch** (k)   
 {   
 **case** 1:   
 init();  
 **break**;  
   
 **case** 2:   
 recursion(n);  
 **break**;  
 }   
  
 }  
 **while** (k!=0);  
  
cout<<**"\n"**;  
**return** 0;  
}

Скриншоты программы:

