



**«Московский государственный технический университет  
имени Н.Э. Баумана»  
(национальный исследовательский университет)  
(МГТУ им. Н.Э. Баумана)**

---

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ  
КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

**О т ч е т**

**по лабораторной работе № 3 (11 вариант)**

**Название лабораторной работы:** Программирование циклического процесса.

Типы циклов

**Дисциплина:** Основы программирования

Студент гр. ИУ6-13Б Be 19.09.2022 В.К. Залыгин  
(Подпись, дата) (И.О. Фамилия)

Преподаватель \_\_\_\_\_  
(Подпись, дата) (И.О. Фамилия)

Москва, 2022

## Цель работы

Рассмотреть решение предложенной задачи с использованием всех трех видов циклов. Отладить программу с наиболее рациональным вариантом цикла. Обосновать выбор.

## Задание

Дано двузначное число  $k$ . Определить  $k$ -ю цифру последовательности 248163264128256... (степени 2). Массивов и строк не использовать.

## Проект программы

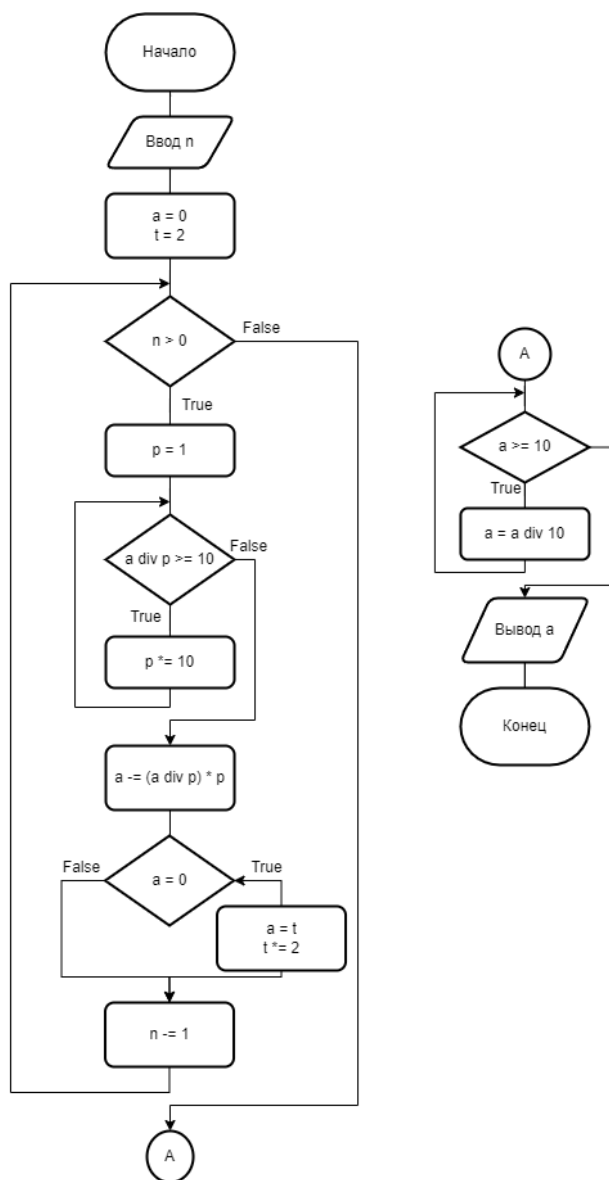


Рисунок 1 - схема алгоритма с циклом while

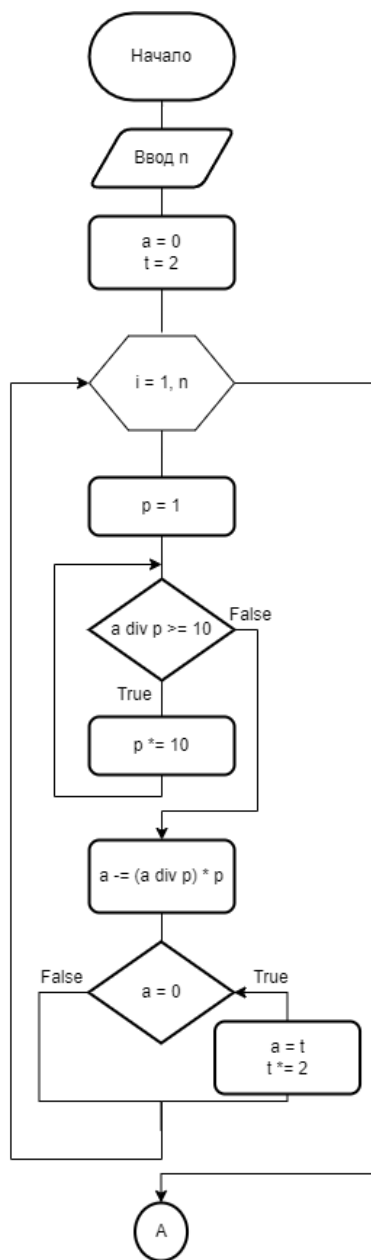


Рисунок 3 - схема алгоритма с циклом for

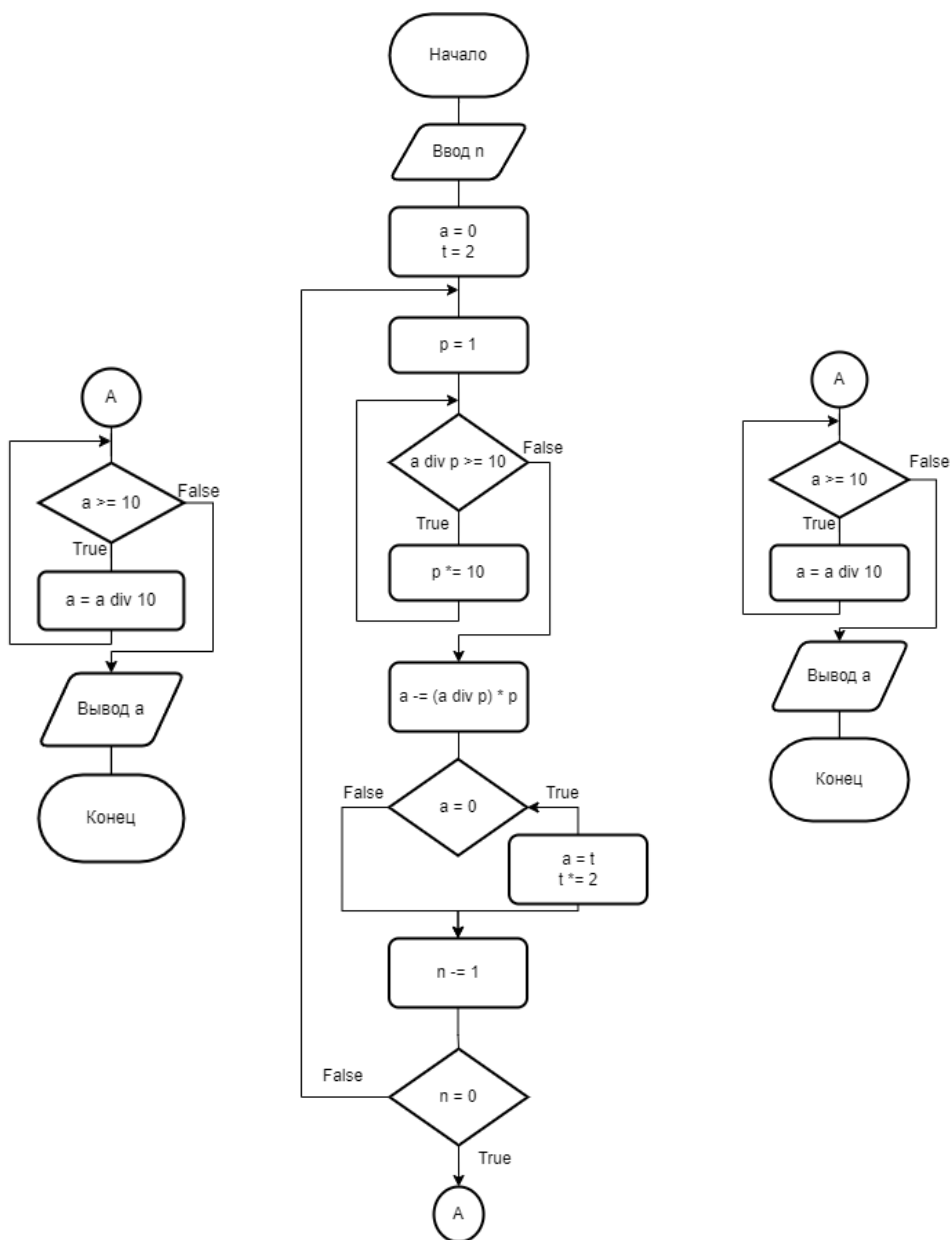


Рисунок 2 - схема алгоритма с циклом until

## Текст программы

```
program lab3;
uses Math;
var i, a, t, n, p :Integer;
begin
  { степень двойки, переменная для проведения поразрядных операций }
  a := 0;
  { степень двойки, которая идёт в последовательности в текущий момент }
  t := 2;
  Write('Введите номер элемента последовательности :> ');
  ReadLn(n);

  { проверка, что мы ещё не дошли до требуемого элемента последовательности }
  for i := 1 to n do
  begin
    { степень 10 для вычитания старшего разряда из операционной степени двойки (переменная a) }
    p := 1;
    { находим старший разряд a }
    while a div p >= 10 do
      p *= 10;
    { вычитаем старший разряд }
    a := a - (a div p) * p;

    { если не осталось цифр в `a`, то смотрим на следующую степень двойки }
    if a = 0 then
    begin
      a := t;
      t := t*2;
    end;
    { обновляем счётчик текущего номера в последовательности }
  end;

  { узнаём цифру старшего разряда `a` }
  while a >= 10 do
    a := a div 10;
  WriteLn('Элемент под введённым номером: ', a);
  ReadLn;
end.
```

Рисунок 4 - код программы с циклом for

```
program lab3;
uses Math;
var a, t, n, p :Integer;
begin
  { степень двойки, переменная для проведения поразрядных операций }
  a := 0;
  { степень двойки, которая идёт в последовательности в текущий момент }
  t := 2;
  Write('Введите номер элемента последовательности :> ');
  ReadLn(n);

  { проверка, что мы ещё не дошли до требуемого элемента последовательности }
  while n > 0 do
  begin
    { степень 10 для вычитания старшего разряда из операционной степени двойки (переменная a) }
    p := 1;
    { находим старший разряд a }
    while a div p >= 10 do
      p *= 10;
    { вычитаем старший разряд }
    a := a - (a div p) * p;

    { если не осталось цифр в `a`, то смотрим на следующую степень двойки }
    if a = 0 then
    begin
      a := t;
      t := t*2;
    end;
    { обновляем счётчик текущего номера в последовательности }
    n -= 1;
  end;

  { узнаём цифру старшего разряда `a` }
  while a >= 10 do
    a := a div 10;
  WriteLn('Элемент под введённым номером: ', a);
  ReadLn;
end.
```

Рисунок 5 - код программы с циклом while

```

program lab3;
uses Math;
var a, t, n, p :Integer;
begin
  { степень двойки, переменная для проведения поразрядных операций }
  a := 0;
  { степень двойки, которая идёт в последовательности в текущий момент }
  t := 2;
  Write('Введите номер элемента последовательности :> ');
  ReadLn(n);

  repeat
    { степень 10 для вычитания старшего разряда из операционной степени двойки (переменная a) }
    p := 1;
    { находим старший разряд a }
    while a div p >= 10 do
      p *= 10;
    { вычитаем старший разряд }
    a := a - (a div p) * p;

    { если не осталось цифр в `a`, то смотрим на следующую степень двойки }
    if a = 0 then
      begin
        a := t;
        t := t*2;
      end;
    { обновляем счётчик текущего номера в последовательности }
    n -= 1;
  { проверка, что мы ещё не дошли до требуемого элемента последовательности }
  until n = 0;

  { узнаём цифру старшего разряда `a` }
  while a >= 10 do
    a := a div 10;
  WriteLn('Элемент под введённым номером: ', a);
  ReadLn;
end.

```

Рисунок 6 - цикл программы с циклом for

## Тестовые данные

Таблица 1 - таблица тестирования

Вводные данные, n	Результат вычисления
1	2
2	4
10	1
11	2
12	8

## Вывод

На мой взгляд рациональнее всего использовать цикл while, так как для нужна дополнительная переменная i, которая в теле цикла не используется. Цикл until не требуется, т.к. тело цикла не нужно вычислять безусловно хотя бы раз.

В данной работе были изучены различные типы циклы, описаны программы с их использованием, решающие задачу на нахождение элемента последовательности.