

Часть 1: «Основы программирования»



Содержание

- Основные понятия.
- Структура программы. Ввод-вывод
- Программирование циклов. Операторы цикла `while`, `for` и `do while`.
- Программирование ветвлений. Оператор `if`

- Обработка числовых последовательностей
- Последовательная обработка символьных данных
- Обработка массивов
- Указатели. Динамические массивы
- Подпрограммы

- Разработка алгоритмов и программ сверху вниз.
- Рекурсивные функции. Библиотечные функции обработки символьных строк
- Структуры и работа с файлами

Пример простой программы на C

```
/* Сложение двух целых чисел */
#include <stdio.h>
main ()
{ int a, b; /* объявление целочисленных переменных a и b */
  printf ("Задайте два числа: "); /* вывод сообщения */
  scanf ("%d %d", &a, &b); /* ввод значений a и b */
  printf ("%d + %d = %d\n", a, b, a+b); /* вывод результата */
  return 0;
}
```

Пример резултата:

Задайте два числа: 328 54

$$328 + 54 = 382$$

Структурное программирование

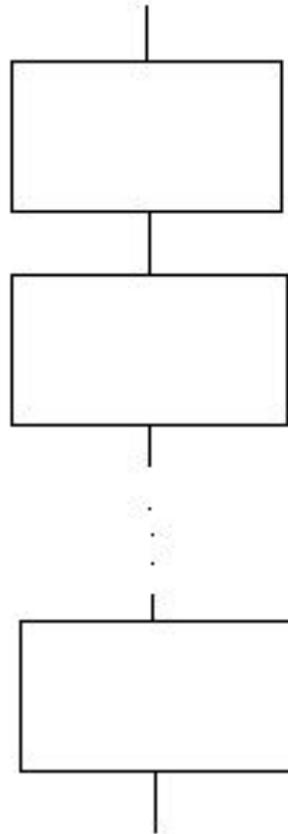
Базовые структуры



Структурное программирование



Последовательность



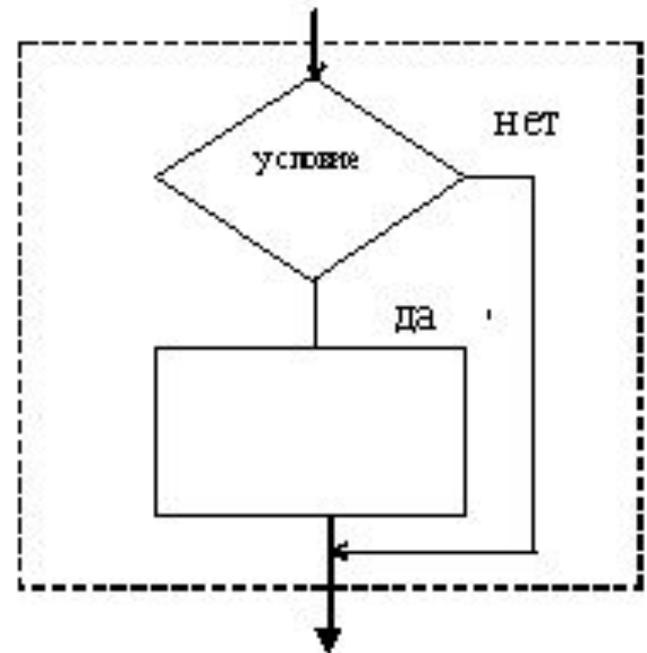
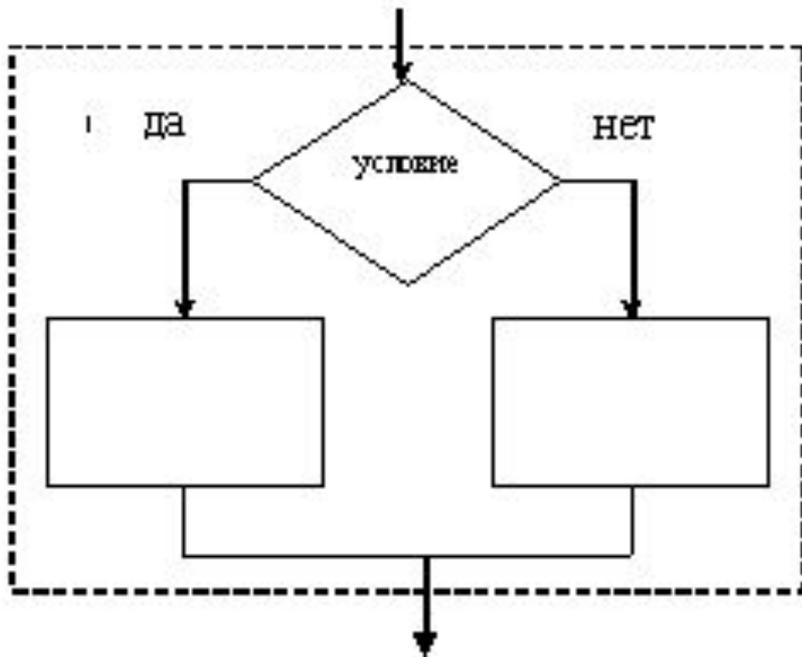
Цикл с предусловием



Цикл с постусловием



Ветвления



Пример решения задачи

- **Задача.** Дано действительное число x .
Вычислить значение $\sin x$ с помощью ряда

$$y = \sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!} + \dots$$

с точностью 10^{-5} (т.е. учитывая только те члены ряда, которые по абсолютной величине больше либо равны 10^{-5}). Для проверки результата вычислить $\sin x$ с помощью стандартной функции.

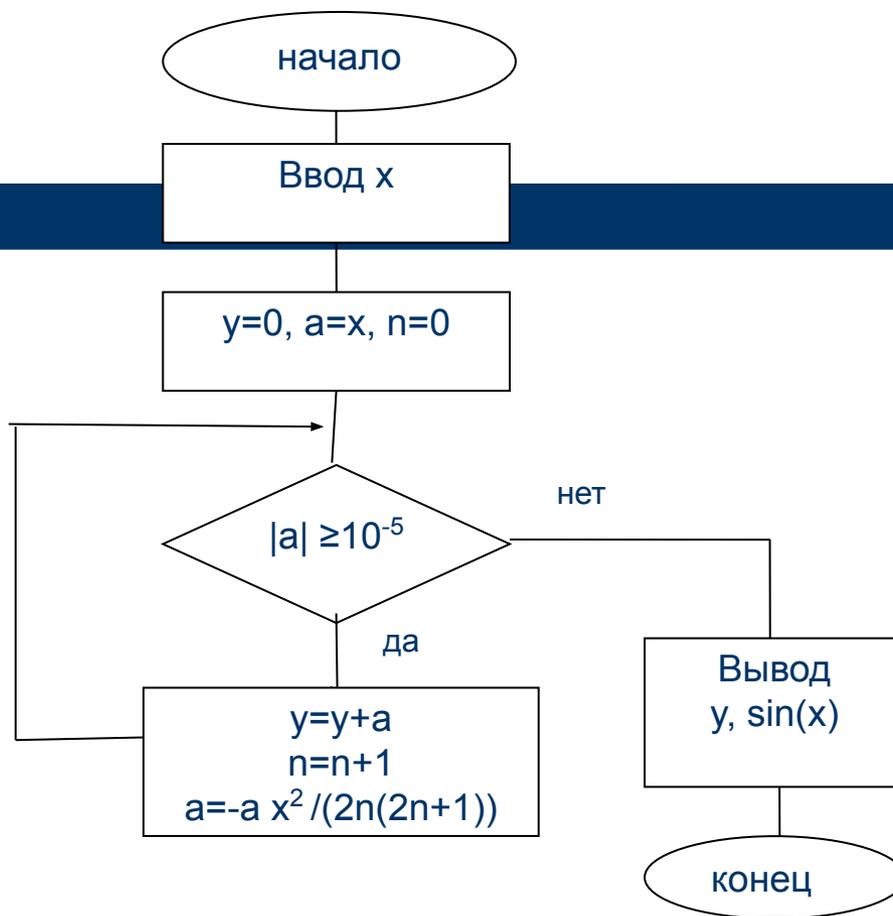


Рис. 2.1. Схема алгоритма вычисления $y = \sin x$

Программа:

- `/* Приближенное вычисление $y = \sin x$ */`
- `#include <stdio.h>`
- `#include <math.h>`
- `#define E 1e-5 /* точность вычисления */`
- `main()`
- `{`
- `float x, /* аргумент функции */`
- `float y; /* сумма ряда */`
- `float a; /* очередной член ряда */`
- `int n; /* номер итерации */`

```
printf ("x = ");
scanf ("%f", &x);
y = 0; a = x; n = 0;
while ( fabs(a) >= E )
{ y = y+a;
  n++;      /* n=n+1; */
  /* вычисление очередного члена
ряда через предыдущий */
  a = -a*x*x/(2*n*(2*n+1));
}
```

```
printf ("y = %f\n", y);  
    printf ("sinx = %f\n", sin(x));  
    return 0;  
}
```

Пример результатов выполнения программы:

$$x = 3.14159$$

$$y = 0.000002$$

$$\sin x = 0.000003$$

$$x = 0$$

$$y = 0.000000$$

$$\sin x = 0.000000$$