

# Цикл for

```
1  for (начало; условие; шаг) {  
2    // ... тело цикла ...  
3  }
```

- ▶ Цикл for – параметрический цикл (цикл с фиксированным числом повторений). Для организации такого цикла необходимо осуществить три операции:
  - Инициализация – присваивание параметру цикла начального значения
  - Условие – проверка условия повторения цикла, чаще всего сравнение величины параметра с некоторым граничным значением
  - Модификация – изменение значения параметра для следующего прохождения тела цикла

# Задача 1

- ▶ Даны два целых числа  $A$  и  $B$ . Выведите все числа от  $A$  до  $B$  включительно, в порядке возрастания, если  $A < B$ , или в порядке убывания в противном случае

## Список задач:

- 1) Даны два целых числа  $A$  и  $B$  (при этом  $A \leq B$ ). Выведите все числа от  $A$  до  $B$  включительно.
- 2) Даны два целых числа  $A$  и  $B$ ,  $A > B$ . Выведите все нечётные числа от  $A$  до  $B$  включительно, в порядке убывания.

### Список задач:

- 1) По данному натуральному  $n$  вычислите сумму  $1^3+2^3+3^3+\dots+n^3$ .
- 2) Факториалом числа  $n$  называется произведение  $1 \times 2 \times \dots \times n$ .  
Обозначение:  $n!$ . По данному натуральному  $n$  вычислите значение  $n!$ . Пользоваться математической библиотекой `math` в этой задаче запрещено.
- 3) Дано вещественное число — цена 1 кг конфет. Вывести стоимость 1.2, 1.4, ..., 2 кг конфет.

# Подготовка к Problem Set 1

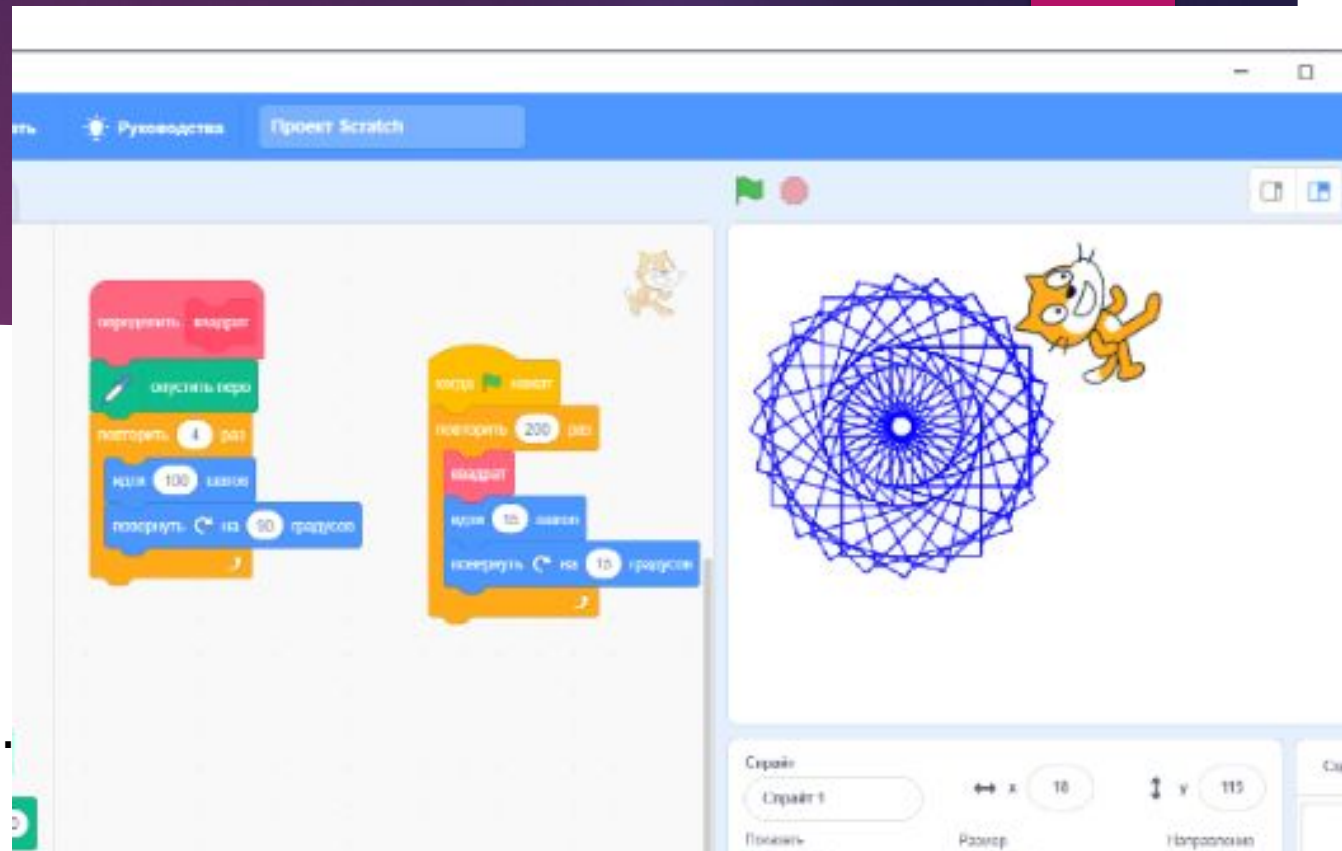
# Домашнее задание

```
$ ./mario
Height: -1
Height: 0
Height: 42
Height: 50
Height: 4
#
##
###
####
```

```
$ ./mario
Height: 8
#
##
###
####
#####
#####
#####
#####
#####
```

# Функции в языке C

**Функция** — это самостоятельная единица программы, которая спроектирована для реализации конкретной подзадачи. Функция является подпрограммой, которая может содержаться в основной программе, а может быть создана отдельно (в библиотеке). Каждая функция выполняет в программе определенные действия.



## Список задач:

1. Написать функцию, принимающего положительное число  $a$ , и целое число  $n$ , и возвращающую число  $a$  в степени  $n$ .
2. Написать функцию, которая по данному целому неотрицательному числу  $n$  возвращает  $n$ -е число Фибоначчи. В этой задаче нельзя использовать циклы. Ученики должны использовать рекурсию.

Последовательность Фибоначчи определяется следующим образом:

$$\begin{aligned}F_0 &= 0, \\F_1 &= 1, \\F_n &= F_{n-1} + F_{n-2}.\end{aligned}$$

Несколько первых её членов:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, , 55, 89, ...



Функция	Описание
<code>int abs(int x)</code>	Модуль целого числа $x$
<code>double acos(double x)</code>	Арккосинус $x$
<code>double asin(double x)</code>	Арсинус $x$
<code>double atan(double x)</code>	Арктангенс $x$
<code>double cos(double x)</code>	Косинус $x$
<code>double cosh(double x)</code>	Косинус гиперболический $x$
<code>double exp(double x)</code>	Экспонента $x$
<code>double fabs(double x)</code>	Модуль вещественного числа
<code>double fmod(double x, double y)</code>	Остаток от деления $x/y$
<code>double log(double x)</code>	Натуральный логарифм $x$
<code>double log10(double x)</code>	Десятичный логарифм $x$
<code>double pow(double x, double y)</code>	$x$ в степени $y$
<code>double sin(double x)</code>	Синус $x$
<code>double sinh(double x)</code>	Синус гиперболический $x$
<code>double sqrt(double x)</code>	Квадратный корень $x$
<code>double tan(double x)</code>	Тангенс $x$
<code>double tanh(double x)</code>	Тангенс гиперболический $x$

```
~/pset1/cash/ $ clang cash.c -lcs50 -lm  
~/pset1/cash/ $ ./a.out
```

### Список задач:

1. Написать функцию `PowerA234(A, B, C, D)`, вычисляющую вторую, третью и четвертую степень числа `A` и возвращающую эти степени соответственно в переменных `B`, `C` и `D` (`A` — входной, `B`, `C`, `D` — выходные параметры; все параметры являются вещественными). С помощью этой функции найти вторую, третью и четвертую степень пяти данных чисел
2. Написать функцию `Mean(X, Y, AMean, GMean)`, вычисляющую среднее арифметическое  $AMean = (X + Y)/2$  и среднее геометрическое  $GMean = \sqrt{X \cdot Y}$  двух положительных чисел `X` и `Y` (`X` и `Y` — входные, `AMean` и `GMean` — выходные параметры вещественного типа). С помощью этой функции найти среднее арифметическое и среднее геометрическое для пар (`A`, `B`), (`A`, `C`), (`A`, `D`), если даны `A`, `B`, `C`, `D`.
3. Написать функцию `Swar(X, Y)`, меняющую содержимое переменных `X` и `Y` (`X` и `Y` — вещественные параметры, являющиеся одновременно входными и выходными). С ее помощью для данных переменных `A`, `B`, `C`, `D` последовательно поменять содержимое следующих пар: `A` и `B`, `C` и `D`, `B` и `C` и вывести новые значения `A`, `B`, `C`, `D`.

# Одномерные массивы в языке С

- ▶ Массив – это непрерывный участок памяти, содержащий последовательность объектов одинакового типа, обозначаемый одним именем.

### Список задач:

- 1) Выведите все элементы массива с четными индексами (то есть  $A[0]$ ,  $A[2]$ ,  $A[4]$ , ...).
- 2) Выведите все чётные элементы (то есть  $A[0]$ ,  $A[2]$ ,  $A[4]$ , ...).
- 3) Дан список чисел. Выведите все элементы списка, которые больше предыдущего элемента.

# Аргументы командной строки

- ▶ Аргументы `argc` типа `integer` содержит в себе количество аргументов командной строки.
- ▶ Аргумент `argv` типа `char` – указатель на массив строк. Каждый элемент массива указывает на аргументы командной строки. Один параметр отделяется от другого пробелом.
  - `argv[0]` - полное имя запущенной программы
  - `argv[1]` - первая строка, записанная после имени программы
  - `argv[2]` - вторая строка, записанная после имени программы
  - `argv[argc-1]` - последняя строка, записанная после имени программы
  - `argv[argc]` - NULL



### Список задач:

- 1) Создать функцию, принимающую в себя целое число и создающую массив заданных длины. Сделать заполнения массива клавиатурным вводом и добавлять в массив только чётные элементы. Вывести получившейся массив.
- 2) Дан массив чисел. Посчитайте, сколько в нем пар элементов, равных друг другу. Считается, что любые два элемента, равные друг другу образуют одну пару, которую необходимо посчитать.
- 3) Даны массивы A и B одинакового размера N. Поменять местами их содержимое и вывести вначале элементы преобразованного массива A, а затем – элементы преобразованного массива B.
- 4) Дан список чисел. Определите, сколько в этом списке элементов, которые больше двух своих соседей, и выведите количество таких элементов. Крайние элементы списка никогда не учитываются, поскольку у них недостаточно соседей.

# Домашнее задание

- ▶ Readability