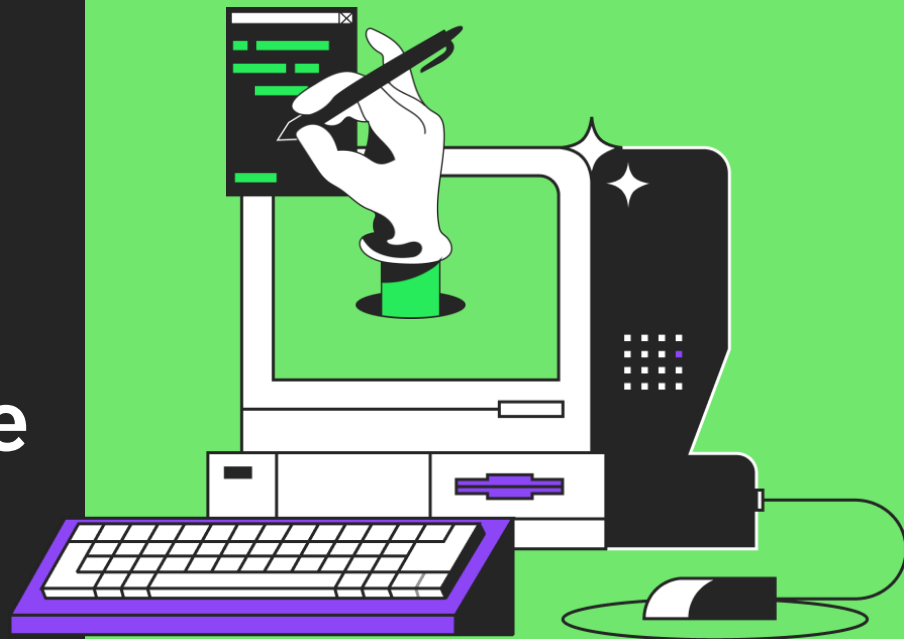


# Введение в программирование

Семинар 2  
Решаем задачи с массивами





## План курса

1

Лекция 1. Введение  
в программирование:  
первые алгоритмы

2

Семинар 1. Создаём  
и записываем свои первые  
алгоритмы

3

Лекция 2. Введение  
в программирование:  
массивы

4

Семинар 2. Решаем задачи  
с массивами

5

Лекция 3. Введение  
в программирование:  
псевдокод и функции

6

Семинар 3. Сравниваем  
разные алгоритмы  
решения задач



## План семинара

- Разбор домашнего задания
- Викторина
- Массивы. Решение задач
- Ответы на вопросы
- Домашнее задание





# Викторина

## Соответствует ли пример атрибуту массива?

### Вагон метро

1. Правда
2. Ложь



## Соответствует ли пример атрибуту массива?

### Вагон метро

1. Правда
2. **Ложь**





## Соответствует ли пример атрибуту массива?

### Фанзона на концерте

1. Правда
2. Ложь





## Соответствует ли пример атрибуту массива?

### Фанзона на концерте

1. Правда
2. Ложь







## Соответствует ли пример атрибуту массива?

### Шкафчики в фитнес-клубе

1. Правда
2. Ложь





## Соответствует ли пример атрибуту массива?

### Шкафчики в фитнес-клубе

1. Правда
2. Ложь



## Соответствует ли пример атрибуту массива?

### Списки учеников в школьном журнале

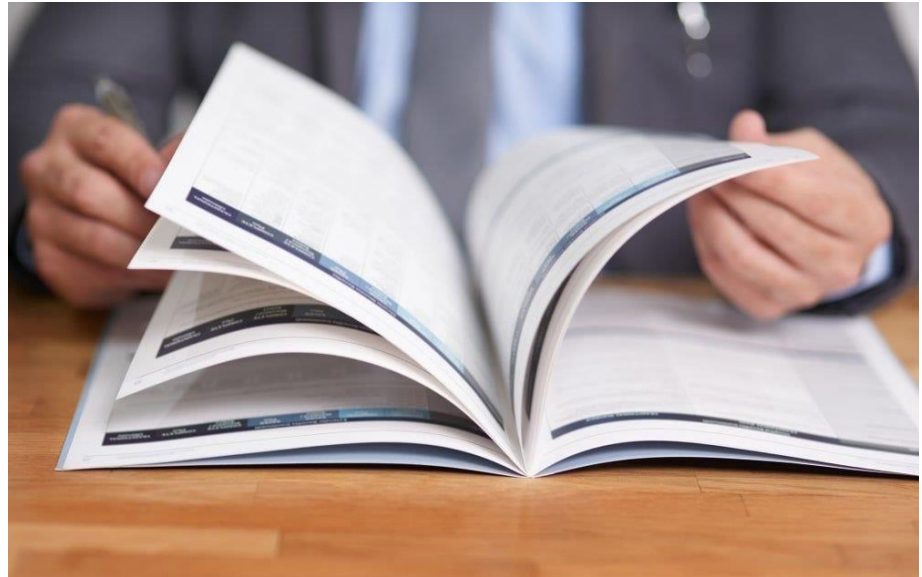
1. Правда
2. Ложь



## Соответствует ли пример атрибуту массива?

### Списки учеников в школьном журнале

1. Правда
2. Ложь





## **Индекс самого первого элемента в массиве в большинстве языков программирования равен?**

1. 0
2. 1



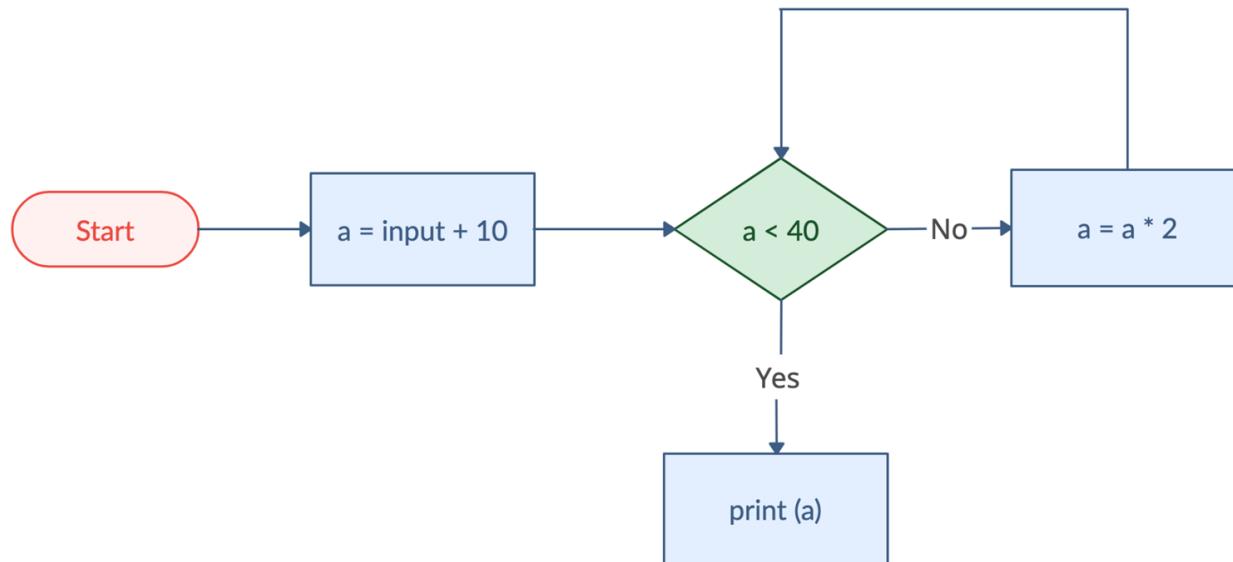
## Индекс самого первого элемента в массиве в большинстве языков программирования равен?

1. 0
2. 1



Что будет выведено в результате алгоритма, если `input=3`?

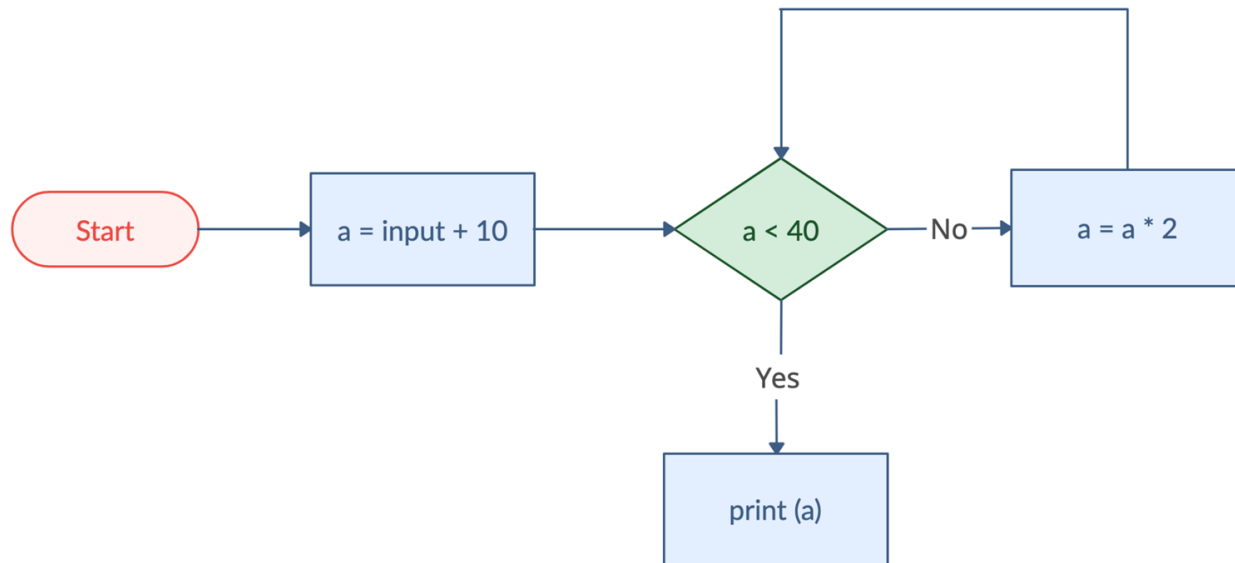
1. 52
2. 15
3. 13
4. 40





Что будет выведено в результате алгоритма, если `input=3`?

1. 52
2. 15
3. 13
4. 40







## Что будет результатом выполнения алгоритма?

1. 16
2. 26
3. 12
4. Тут логическая ошибка в 3 строке

```
1 a = 15
2 b = 4
3 a = a - 5
4 b = 2 * a - b
5 print(b)
```



## Что будет результатом выполнения алгоритма?

1. 16
2. 26
3. 12
4. Тут логическая ошибка в 3 строке

```
1 a = 15
2 b = 4
3 a = a - 5
4 b = 2 * a - b
5 print(b)
```



## Какие значения будут внутри массива List?

1. {2, 1, 0, -1, -2}
2. {-2, -1, 0, 1, -2}
3. {-2, -1, 0, -1, -2}
4. {2, -1, 0, -1, 2}

```
1 list = [-2, -1, 0, 1, 2]
2 index = 0
3 while index < 5:
4     list[index] = -1 * list[index]
5     index = index + 1
```



## Какие значения будут внутри массива List?

1. {2, 1, 0, -1, -2}
2. {-2, -1, 0, 1, -2}
3. {-2, -1, 0, -1, -2}
4. {2, -1, 0, -1, 2}

```
1 list = [-2, -1, 0, 1, 2]
2 index = 0
3 while index < 5:
4     list[index] = -1 * list[index]
5     index = index + 1
```



## Какие значения будут в массиве array?

1. {0, 0, 0, 0, 0}
2. {-2, -1, 0, 1, -2}
3. {-2, -1, 0, 1, 2}
4. {0, 0, 0, 1, 2}

```
1 int[] array = {-2, -1, 0, 1, 2};    Java
2 int index = 0;
3 while (index < array.length) {
4     if (array[index] < 0) {
5         array[index] = 0;
6     }
7     index = index + 1;
8 }
```



## Какие значения будут в массиве array?

1. {0, 0, 0, 0, 0}
2. {-2, -1, 0, 1, -2}
3. {-2, -1, 0, 1, 2}
4. {0, 0, 0, 1, 2}

```
1 int[] array = {-2, -1, 0, 1, 2};    Java
2 int index = 0;
3 while (index < array.length) {
4     if (array[index] < 0) {
5         array[index] = 0;
6     }
7     index = index + 1;
8 }
```



Вопросы?

Вопросы?



Вопросы?





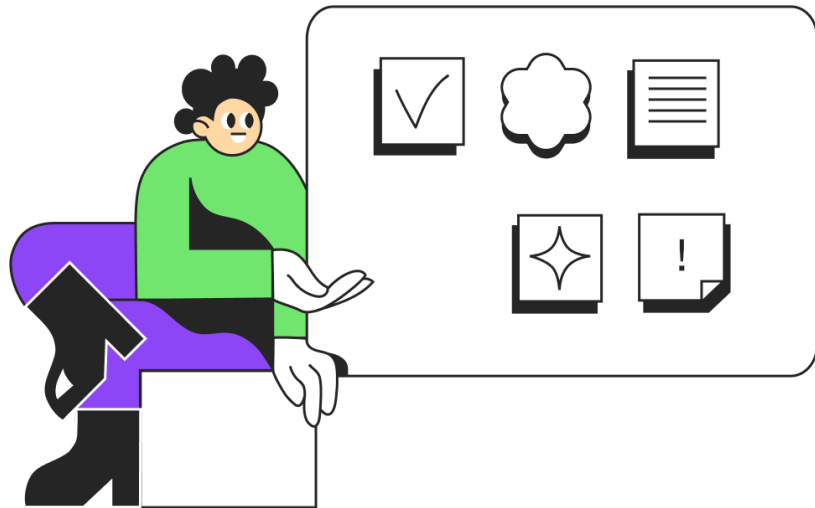
# Немного теории





# Массивы

**Массив** – это группа однотипных элементов, имеющих общее имя и расположенных в памяти рядом.





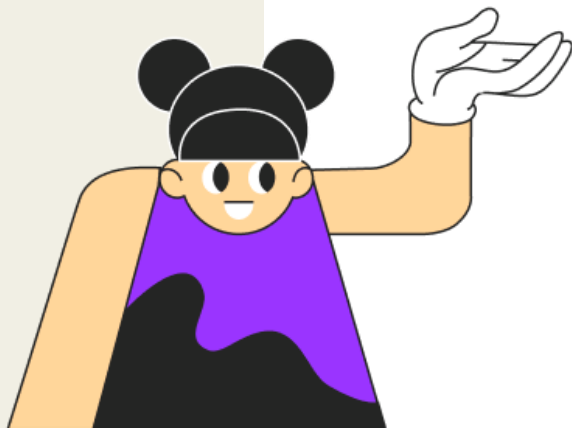
# Массивы

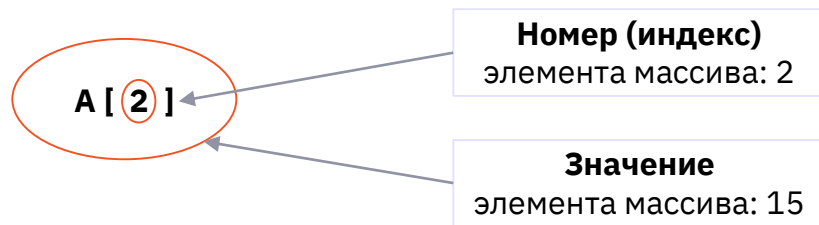
## Особенности:

- ✓ все элементы имеют один тип
- ✓ весь массив имеет одно имя
- ✓ все элементы расположены в памяти рядом

## Примеры:

- список учеников в классе
- квартиры в доме
- школы в городе
- данные о температуре воздуха за год







# Практика



# Задача №1





## Задача №1. Общее обсуждение

Составить алгоритм вычисления суммы чека.  
Сначала берём 2 или 3 позиции товара, затем произвольной размерности.



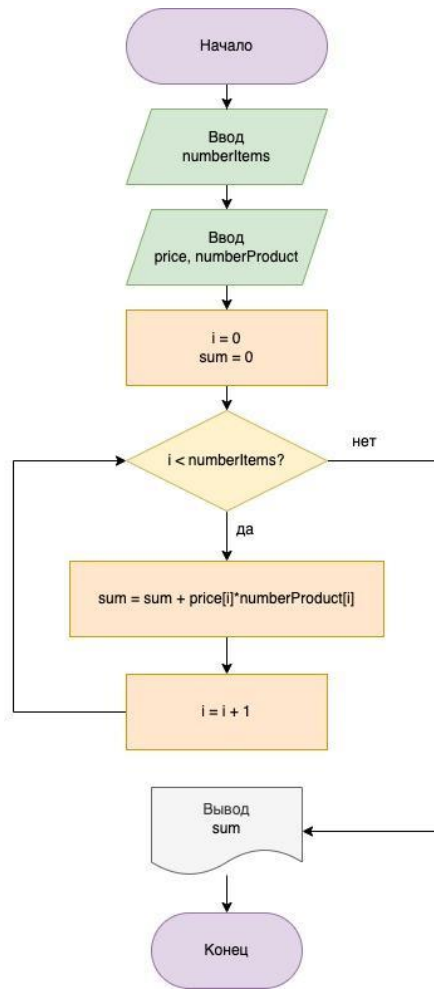
**5 минут**



## Задача №1. Общее обсуждение

Составить алгоритм вычисления суммы чека.  
Сначала берём 2 или 3 позиции товара, затем произвольной размерности.

[Схема](#) (draw.io)





# Задача №2







## Задача №2. Работа в группах

Найти сумму всех элементов массива с нечетными индексами.

A large teal circle on the right side of the slide, containing the text '10 минут' in white.

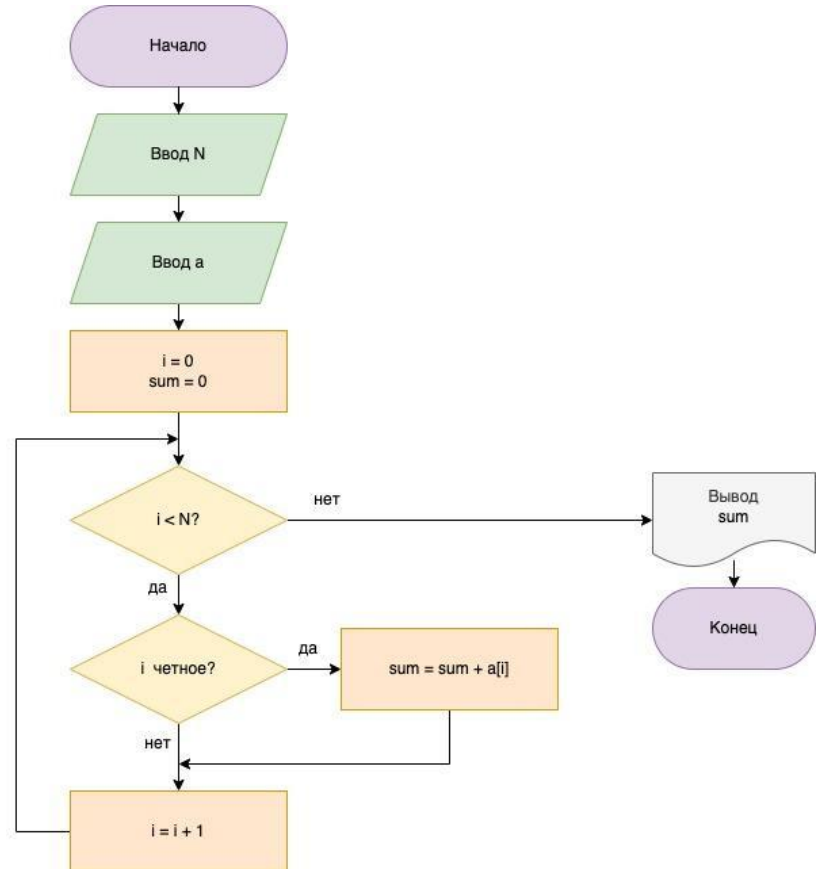
**10 минут**



## Задача №2. Общее обсуждение

Найти сумму всех элементов массива с нечетными индексами.

[Схема](#) (draw.io)





# Задача №3





## Задача №3. Работа в группах

Найти сумму всех элементов массива с нечетным значением.

Например, для массива [1, 2, 4, 6] ответ будет 1, т.к. здесь у нас только один нечетный элемент.



**10 минут**

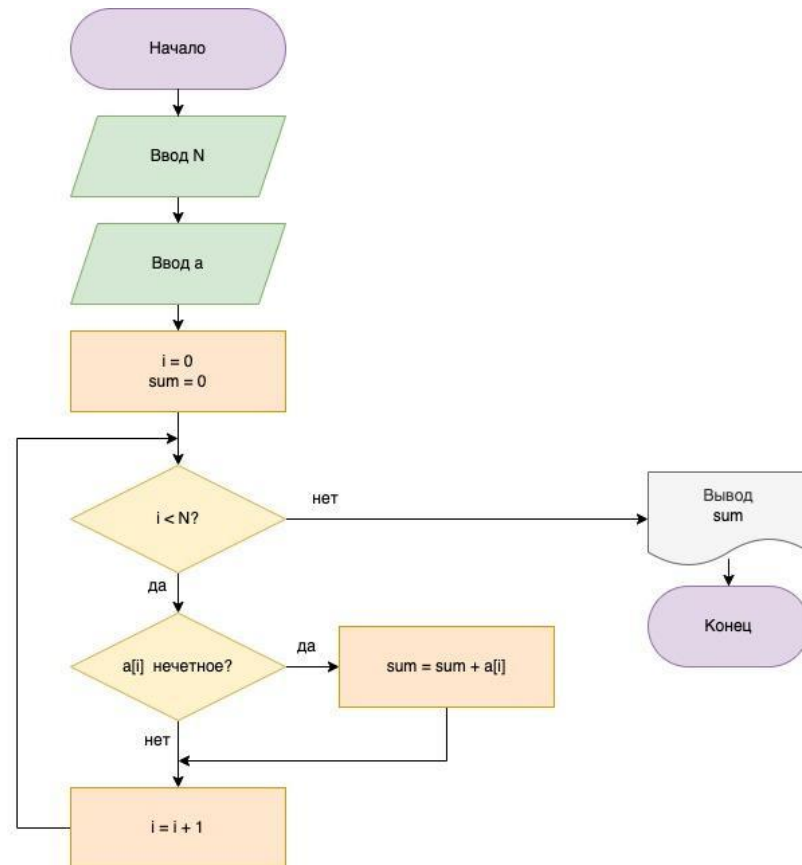


## Задача №3. Общее обсуждение

Найти сумму всех элементов массива с нечетным значением.

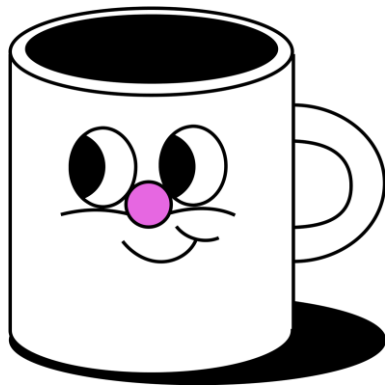
Например, для массива [1, 2, 4, 6] ответ будет 1, т.к. здесь у нас только один нечетный элемент.

[Схема](#) (draw.io)





## Перерыв



<<5:00->>



# Задача №4





## Задача №4. Общее обсуждение

Записать алгоритм, который «перенесет» самое большое значение элемента в правый край массива.



**10 минут**





## Пузырьковая сортировка

Алгоритм состоит из повторяющихся проходов по сортируемому массиву. За каждый проход элементы последовательно сравниваются попарно и, если порядок в паре неверный, выполняется перестановка элементов. Проходы по массиву повторяются  $N-1$  раз или до тех пор, пока на очередном проходе не окажется, что обмены больше не нужны, что означает — массив отсортирован.

При каждом проходе алгоритма по внутреннему циклу очередной наибольший элемент массива ставится на своё место в конце массива рядом с предыдущим «наибольшим элементом», а наименьший элемент перемещается на одну позицию к началу массива («всплывает» до нужной позиции, как пузырёк в воде — отсюда и название алгоритма).

5	2	1	3	9	0	4	6	8	7
---	---	---	---	---	---	---	---	---	---



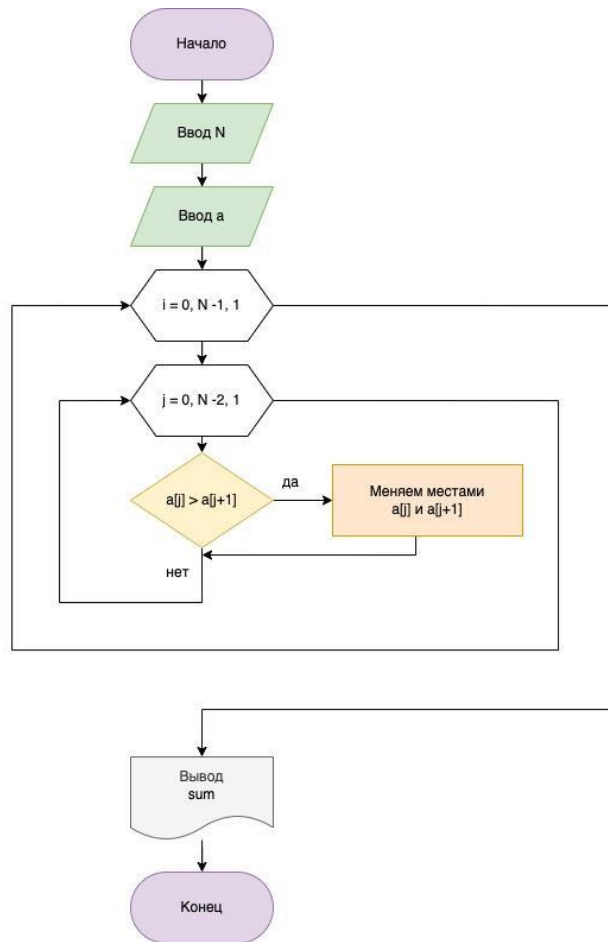
## Задача №4. Общее обсуждение

Записать алгоритм, который «перенесет» самое большое значение элемента в правый край массива.

Схема (draw.io) - пузырьковая сортировка

$i$  – сколько раз сравниваем,  
количество повторов

$j$  – нужен для сравнения  
соседних элементов массива





# Задача №5





## Задача №5. Работа в группах

Записать сортировку массива на основе того, что сделали в предыдущем пункте.

- сортировка пузырьком
- сортировка выбором



**15 минут**



## Задача №5. Общее обсуждение

Записать сортировку массива на основе того, что сделали в предыдущем пункте.

- сортировка пузырьком
- сортировка выбором

[1, 8, 3, 2, 6]

[1, 6, 3, 2, 8]

[1, 6, 3, 2, 8]

[1, 2, 3, 6, 8]

[1, 2, 3, 6, 8]

[1, 2, 3, 6, 8]

[1, 2, 3, 6, 8]

[1, 2, 3, 6, 8]



# Задача №6





## Задача №6. Работа в группах

Перевернуть массив и записать его в обратном порядке.

A large teal circle on the right side of the slide, containing the text '15 минут' in white.

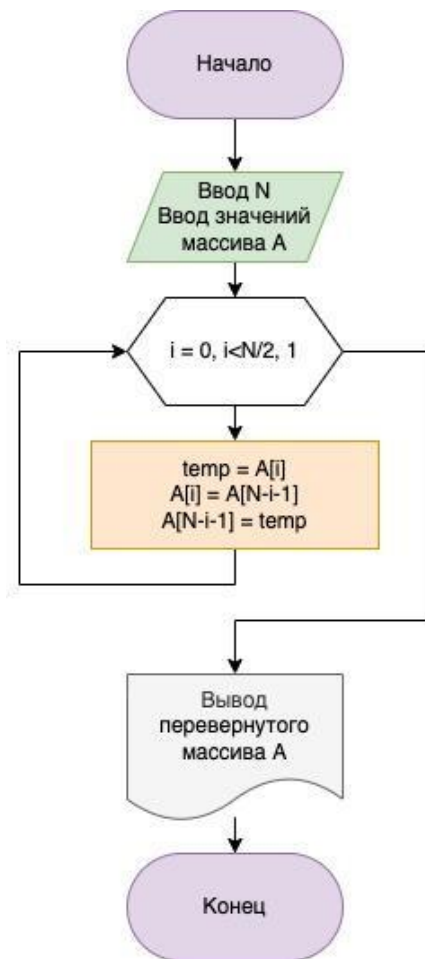
**15 минут**



## Задача №6. Общее обсуждение

Перевернуть массив и записать его в обратном порядке.

[Схема](#) (draw.io)







Вопросы?

Вопросы?



Вопросы?





# Домашнее задание



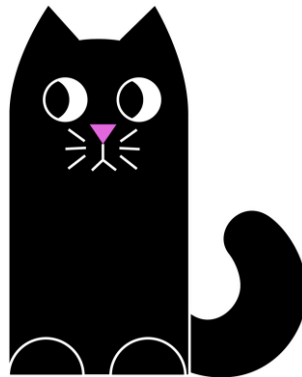
## Домашнее задание

Решить следующие задачи. Нарисовать блок-схемы:

1. Нахождения индексов максимального и минимального элемента массива.
1. Задание на «разворот» массива. Нужно перевернуть массив и записать его в обратном порядке\*.
1. Найти среднее арифметическое среди всех элементов массива
1. Найти сумму элементов массива, лежащих между максимальным и минимальным по значению элементами\*\*

\* если не успели на семинаре

\*\* задачи повышенной сложности





## Рефлексия



**Был урок полезен вам?**



**Узнали вы что-то новое?**



**Что было сложно?**



Спасибо за внимание!