

Разбор домашнего задания

Роман Булгаков

Спикер курса

Skillbox

Задача «Максимальное из двух»

Входные данные:

- два числа

Выходные данные:

- максимальное из двух чисел

```
Введите первое число: 3  
Введите второе число: 8  
Максимальное число: 8.0
```

Разбор задачи

$a = 8 \quad b = 3$

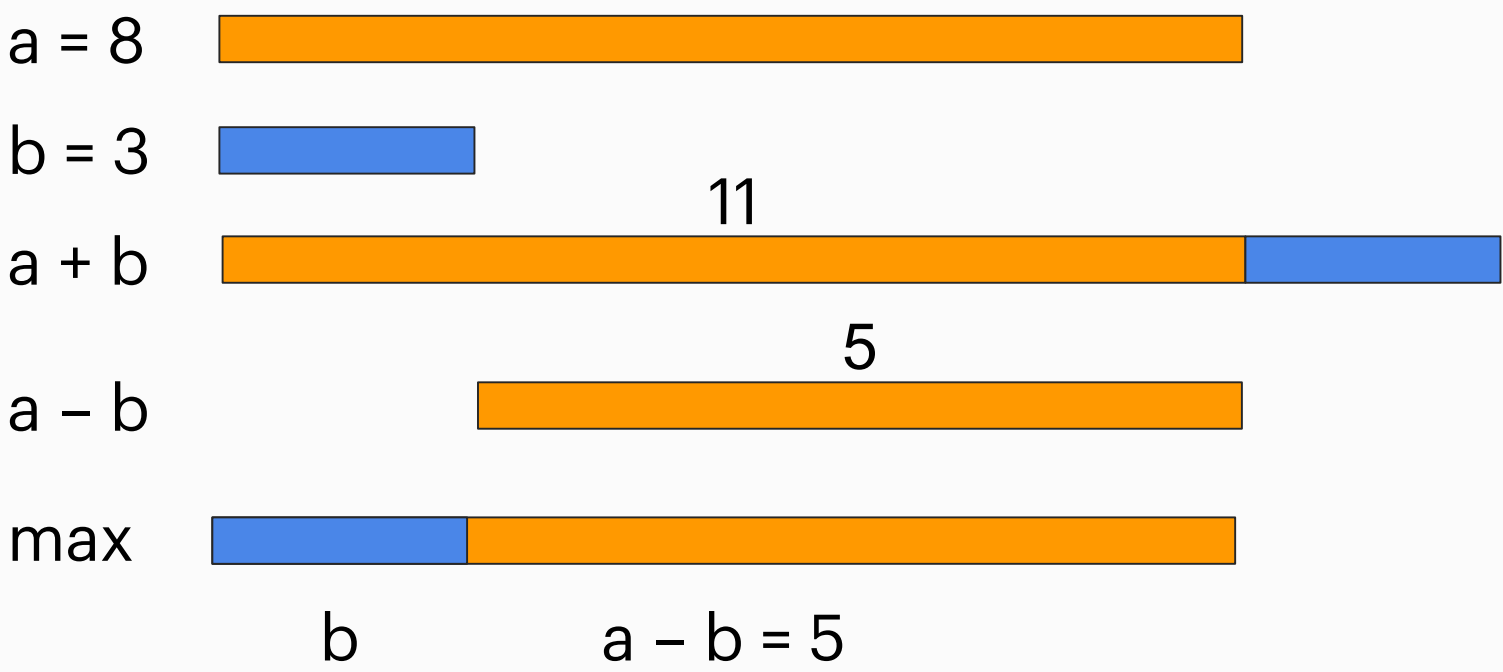
Обмен двух переменных

$a = a + b \quad \#8+3=11$

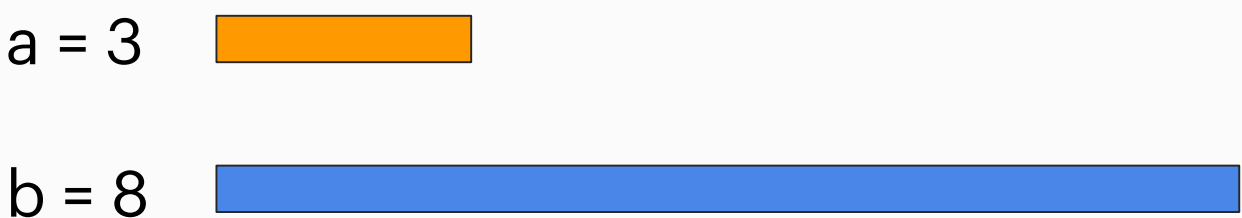
$b = a - b \quad \#11-3=8$

$a = a - b \quad \#11-8=3$

Разбор задачи



$\text{max} = b + (a - b) \quad 3 + (8 - 3) = 8$



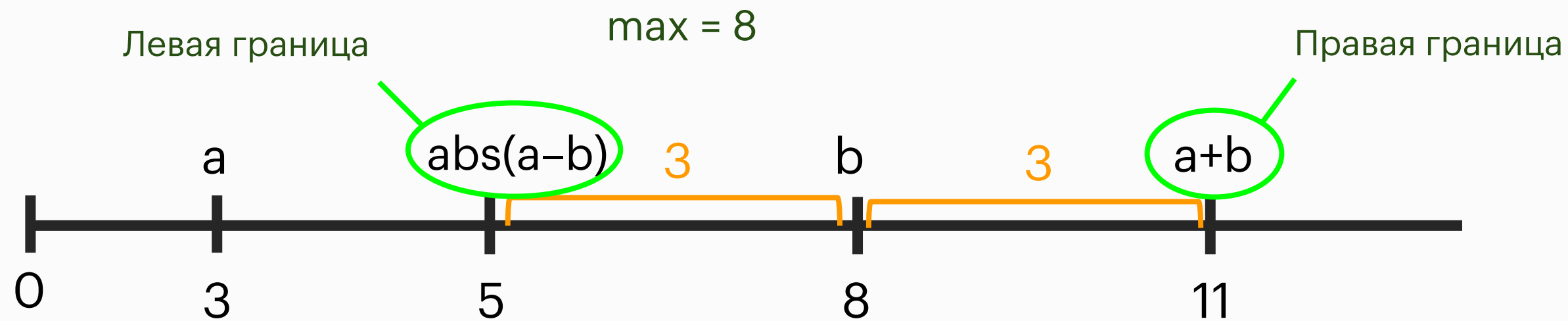
$\text{max} = b + (b - a) \quad 8 + (3 - 8) = 8 + (-5) = 3$

$\text{max} = b + \text{abs}(a - b)$

$a = 8, b = 3 \quad 3 + \text{abs}(8 - 3) = 8$

$8 + \text{abs}(3 - 8) = 8 + 5 = 13 \quad ???$

Разбор задачи



$$\max = (a + b + \text{abs}(a - b)) / 2$$

$$a = 8, b = 3 \quad (8 + 3 + \text{abs}(8 - 3)) / 2 = (11 + 5) / 2 = 16 / 2 = 8$$

$$a = 3, b = 8 \quad (3 + 8 + \text{abs}(3 - 8)) / 2 = (11 + 5) / 2 = 16 / 2 = 8$$

$$a = -2, b = 4 \quad (-2 + 4 + \text{abs}(-2 - 4)) / 2 = (2 + 6) / 2 = 8 / 2 = 4$$

Разбор задачи

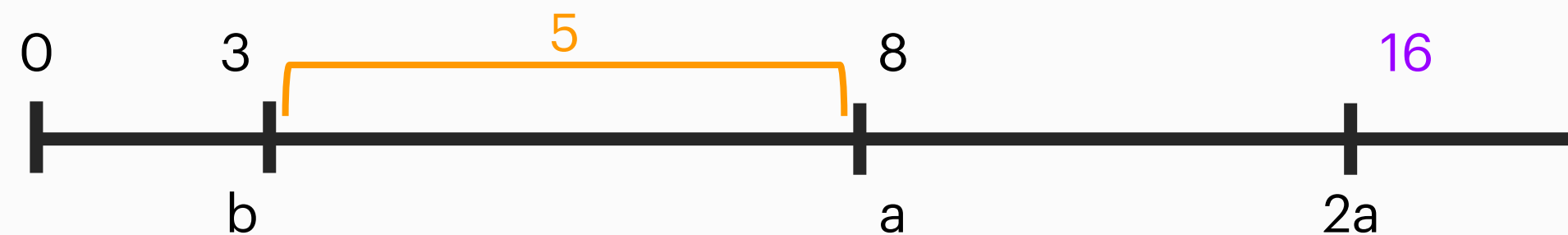
$$a = 8 \quad b = 3$$

$$a + b = 11$$

$$a - b = 5$$

$$11 + 5 = 16$$

$$a + b + (a - b) = 2a$$



$$\text{maximum} = (a + b + (a - b)) / 2 \quad \text{— если } a > b$$

$$\text{maximum} = (a + b + (b - a)) / 2 \quad \text{— если } b > a$$

$$a = 3 \quad b = 8$$

$$\text{maximum} = (a + b + (a - b)) / 2$$

$\underbrace{\hspace{1.5cm}}$
-5

$$\text{maximum} = (a + b + \text{abs}(a - b)) / 2$$

$\underbrace{\hspace{1.5cm}}$
5

Функции и их вызов

Роман Булгаков

Спикер курса

Skillbox

Суммарное количество корма

```
for food in range(3):  
    fruits = int(input('Сколько фруктов? '))  
    vegetables = int(input('Сколько овощей? '))  
    summ = fruits + vegetables  
    print('Всего:', summ)
```


Задача «Зоопарк»

Условие задачи:

- есть обезьяны, жирафы и слоны
- количество фруктов и овощей для каждого вида

Выходные данные:

- сумма для обезьян
- сумма для жирафов
- сумма для слонов

```
for food in range(3):  
    fruits = int(input('Сколько фруктов? '))  
    vegetables = int(input('Сколько овощей? '))  
    summ = fruits + vegetables  
    print('Всего:', summ)
```

Сравнение кода

Старый код

main.py

```
1  print('Обезьяны')
2  fruits = int(input('Сколько фруктов? '))
3  vegetables = int(input('Сколько овощей? '))
4  print('Всего:', fruits + vegetables)
5
6  print('\nЖирафы')
7  fruits = int(input('Сколько фруктов? '))
8  vegetables = int(input('Сколько овощей? '))
9  print('Всего:', fruits + vegetables)
10
11 print('\nСлоны')
12 fruits = int(input('Сколько фруктов? '))
13 vegetables = int(input('Сколько овощей? '))
14 print('Всего:', fruits + vegetables)
```

Новый код

main.py

```
1  def countFood():
2      fruits = int(input('Сколько фруктов? '))
3      vegetables = int(input('Сколько овощей? '))
4      print('Всего:', fruits + vegetables)
5
6  print('Обезьяны')
7  countFood()
8
9  print('\nЖирафы')
10 countFood()
11
12 print('\nСлоны')
13 countFood()
```

Задача «Фигуры»

Входные данные:

- пользователь вводит 1 (треугольник) или 2 (прямоугольник)

Выходные данные:

- фигура из звёздочек, 5 строк

```
Что рисуем? 1 - треугольник, 2 - прямоугольник: 1
  *
 ***
*****
*****
*****
```

```
Что рисуем? 1 - треугольник, 2 - прямоугольник: 2
*****
*      *
*      *
*      *
*****
```

Использование функций

```
1 # CTCI Problem 1.3
2 # Write a method to decide if one string is the permutation of another.
3 # Permutation if same amount of letters and length.
4
5 # O(n*logn), 27.4 usec per loop
6 def permutation(a, b):
7     return sorted(a) == sorted(b)
8
9 # Faster, O(n), 16 usec per loop
10 def permutation1(a, b):
11     mapping_a, mapping_b = {}, {}
12     for char in a:
13         mapping_a[char] = 1 if char not in mapping_a else + 1
14     for char in b:
15         mapping_b[char] = 1 if char not in mapping_b else + 1
16     return(mapping_a == mapping_b)
17
18 # Uses only one dict, but timeit says it's slower: O(n), 23 usec per loop.
19 def permutation2(a, b):
20     mapping = {}
21     for char in a:
22         mapping[char] = 1 if char not in mapping else + 1
23     for char in b:
24         if char not in mapping:
25             return False
26         mapping[char] -= 1
27     return(sum(mapping.values()) == 0)
28
29 # Test cases + cool anagrams.
30 def tests():
31     print(permutation("Tom Marvolo Riddle", "I Am Lord Voldemort")) # False
32     print(permutation("silent", "listen")) # True
33     print(permutation("eleven plus two", "twelve plus one")) # True
34     print(permutation("debit card", "bad credit")) # True
35
36 # python3 -m timeit -s 'from chap1prob3 import permutation2 as p' '(p("eleven plus two", "twelve plus one"))'
```

Итоги урока

- `def myAdress(name):`
 `print('Имя:', name)`
- `name` — это параметр (аргумент)
- Передача аргумента в функцию:
 `myAdress('Василий')`

Функции с одним параметром

Роман Булгаков

Спикер курса

Skillbox

Задача «Почта»

```
def myAdress():  
    print('Фамилия: Иванов')  
    print('Имя: Василий')  
    print('Улица: Пушкина')  
    print('Дом: 32')  
    print()
```

myAdress()

myAdress()

myAdress()

Задача «Функция»

Входные данные:

- икс от -10 до 10

Выходные данные:

- $y = f(x)$

$$y = \begin{cases} e^x, & -5 \leq x \leq 5 \\ 2|x| - 1, & x < -5 \\ 2x, & x > 5 \end{cases}$$

Итоги урока

- Функция — участок кода (подпрограмма), который можно вызывать когда удобно
- `def`
- `def countFood():`
- `def countFood():`
 - `fruits = int(input('Сколько фруктов? '))`
 - `vegetables = int(input('Сколько овощей? '))`
 - `print('Всего:', fruits + vegetables)`

Функции с несколькими параметрами

Роман Булгаков

Спикер курса

Skillbox

Задача «Семья»

Условие задачи:

- 3 человека
- разные имена
- разные номера домов

Выходные данные:

- Ф. И. и адрес каждого члена семьи



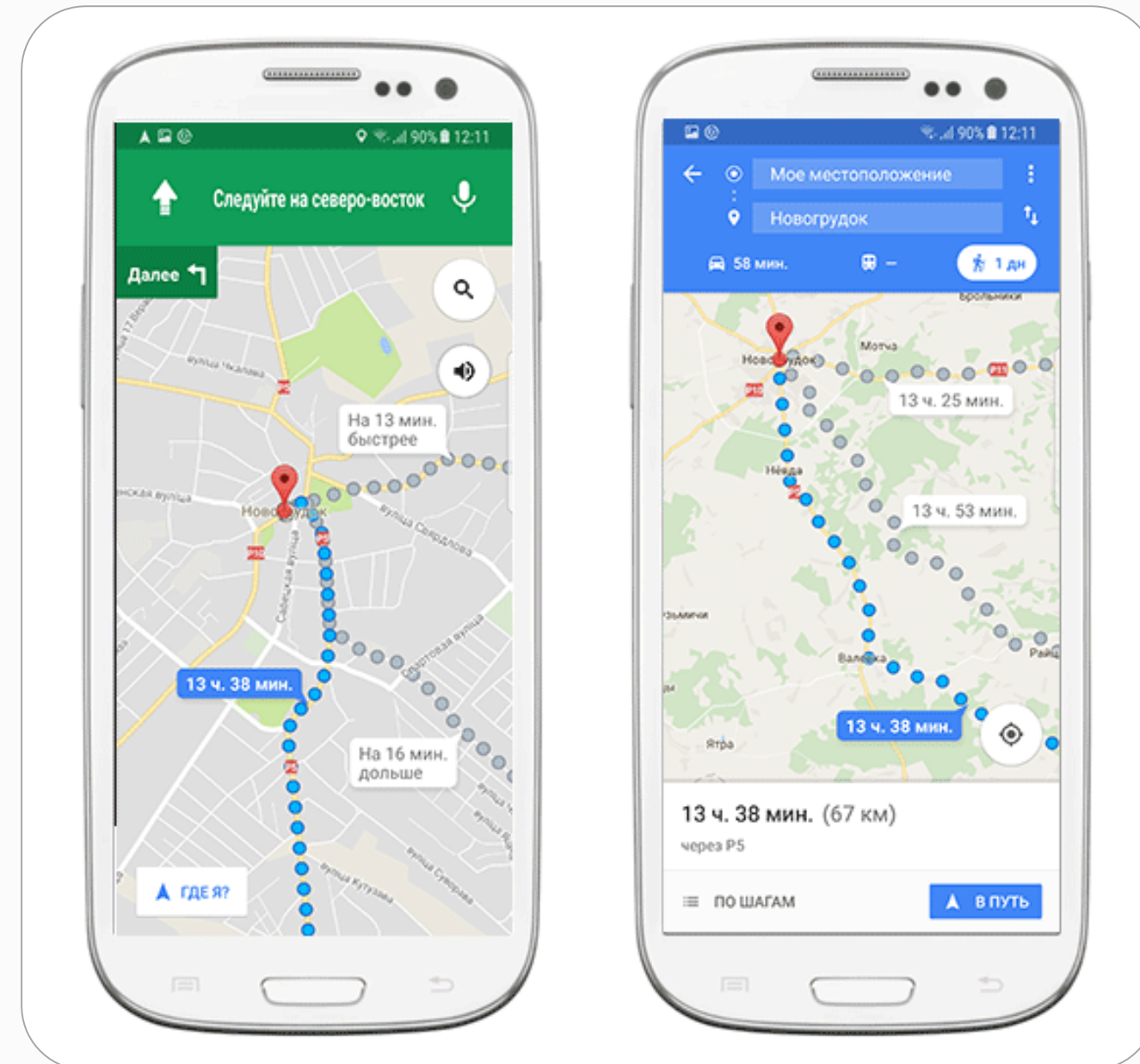
Задача «Навигатор»

Условие задачи:

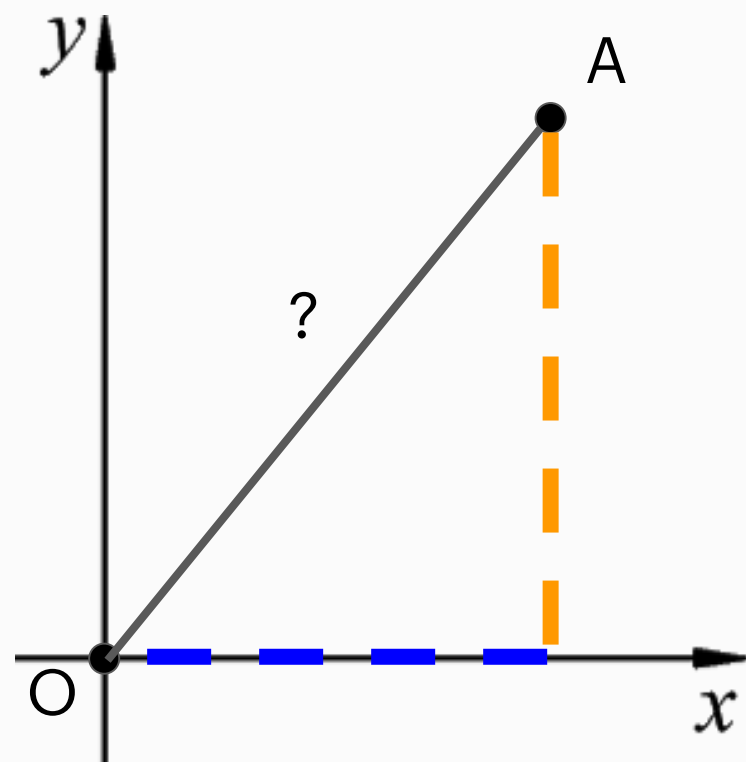
- (x_1, y_1) — координаты первой точки
- (x_2, y_2) — координаты второй точки

Выходные данные:

- расстояние между двумя точками



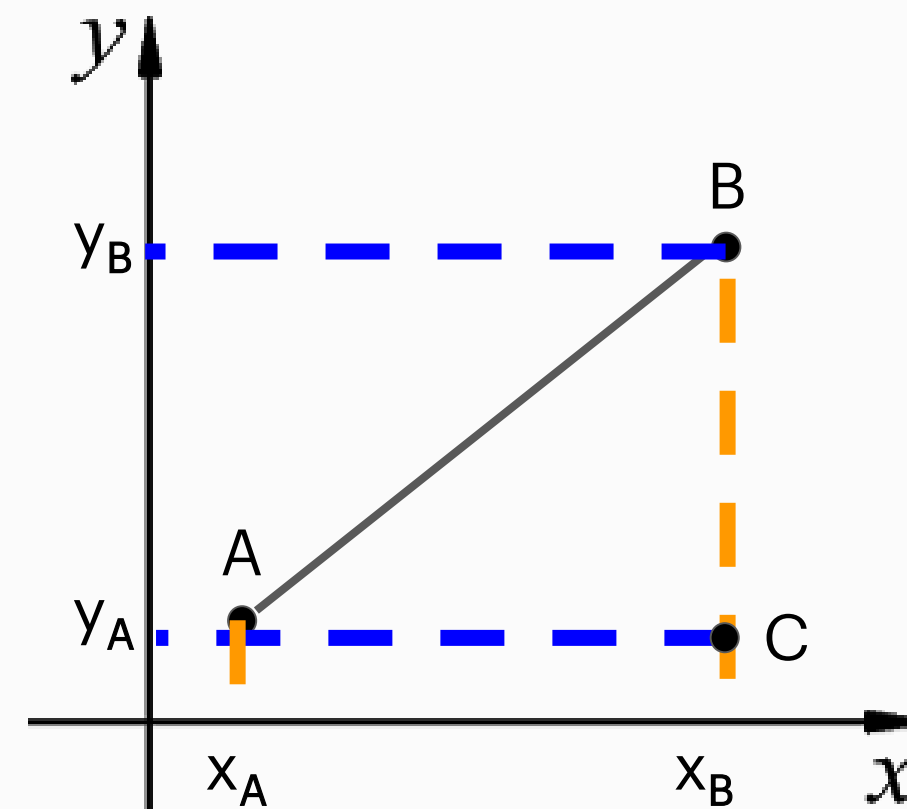
Разбор задачи



Теорема Пифагора

$$(OA)^2 = (Ox)^2 + (Oy)^2$$

$$OA = \sqrt{(Ox)^2 + (Oy)^2}$$



$$AB = \sqrt{(AC)^2 + (BC)^2}$$

$$AC = x_B + x_A$$

$$BC = y_B - y_A$$

Вложенные вызовы функций

Роман Булгаков

Спикер курса

Skillbox

Использование функций

```
15 MENU = '''
16     A) Read collection of contacts from file
17     B) Write collection of contacts to file
18     C) Add new contact
19     D) Remove existing contact
20     E) Update existing contact's phone number
21     F) Update existing contact's email address
22     G) Display contacts by prefix
23     X) Exit from the program'''
24 def update_phone(D):
25     name = input("Name: ")
26     if name in D:
27         phone = input("Phone: ")
28         if validate_phone(phone):
29             D[name][0] = phone
30         else:
31             print("Phone number not acceptable, use format XXX-XXX-XXXX")
32     else:
33         print("Name not found")
34     return D
35 print(MENU)
36 choice = input("Choice: ")
37 D = {}
38 while choice.upper() != 'X':
39     if choice.upper() == 'A':
40         #Function call to read file
41         print("Selected A")
42         D = read_contacts(D)
43     if choice.upper() == 'B':
44         #function call to write file.
45         print("Selected B")
46     if choice.upper() == 'C':
47         #function call to create contact.
48         print("Selected C")
49     if choice.upper() == 'D':
50         #function call to remove contact.
```

Задача «Игра»

Условие задачи:

- выбор: сделать хорошо или плохо
- если плохо, — игра завершается

Выходные данные:

- главное меню
- сообщение «Всё хорошо» или «Всё плохо»

```
1. Сделать что-то хорошее
2. Сделать что-то плохое
Введите номер действия: 1
Всё хорошо
```

```
1. Сделать что-то хорошее
2. Сделать что-то плохое
Введите номер действия: 2
Всё плохо
```



Итоги модуля

- `def countFood():`
 <тело функции>
 `countFood()` # использование функции
- `def myAdress(name, house):`
 `print('Имя:', name)`
 `print('Дом:', house)`
- `def mainMenu():`
 ...
 `if choice == 1:`
 `good()`

