

Техническая специализация Java: Java Core

Семинар 2

Функции, манипулирующие данными





Функции, манипулирующие данными

1

Лекция 1.
Платформа: история и окружение

2

Семинар 1.
Компиляция и интерпретация кода

3

Лекция 2.
Специализация: данные и функции

4

Семинар 2.
Функции, манипулирующие данными

5

Лекция 3.
Специализация: ООП

6

Семинар 3.
Классы и объекты

7

Лекция 4.
ООП и исключения

8

Семинар 4.
Обработка исключений

9

Лекция 5.
Специализация: Тонкости работы

10

Семинар 5.
Внешний мир



Цели семинара

1. Закрепить полученные на лекции знания, хранения примитивных и ссылочных типов данных;
2. Получить практический навык создания функций по описанию;
3. Попрактиковаться в написании простых функций, манипулирующих ссылочными данными.





План семинара

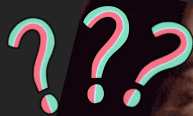
1. Приветствие;
2. Quiz;
3. Рассмотрение домашнего задания;
4. Вопросы и ответы;
5. Практика;
6. Домашнее задание;
7. Рефлексия.





Вопросы?

Вопросы?



Вопросы?





Викторина



Магическое число – это:

1. Числовая константа без пояснений;
2. Число, помогающее в вычислениях;
3. Числовая константа, присваиваемая при объявлении переменной.



Магическое число – это:

1. Числовая константа без пояснений;
2. Число, помогающее в вычислениях;
3. Числовая константа, присваиваемая при объявлении переменной.

```
a = 18;  
b = 3,14;  
c = 100;  
d = 33;
```


Магическое число – это:

1. Числовая константа без пояснений;
2. Число, помогающее в вычислениях;
3. Числовая константа, присваиваемая при объявлении переменной (это инициализация)





Какое значение будет содержаться в переменной a после выполнения строки `int a = 10.0f/3.0f;`



Какое значение будет содержаться в переменной a после выполнения строки `int a = 10.0f/3.0f;`

Результат деления не целочисленный.



Какое значение будет содержаться в переменной a после выполнения строки `int a = 10.0f/3.0f;`

Ответ: 3

Переменная слева может содержать только целую часть получившегося числа



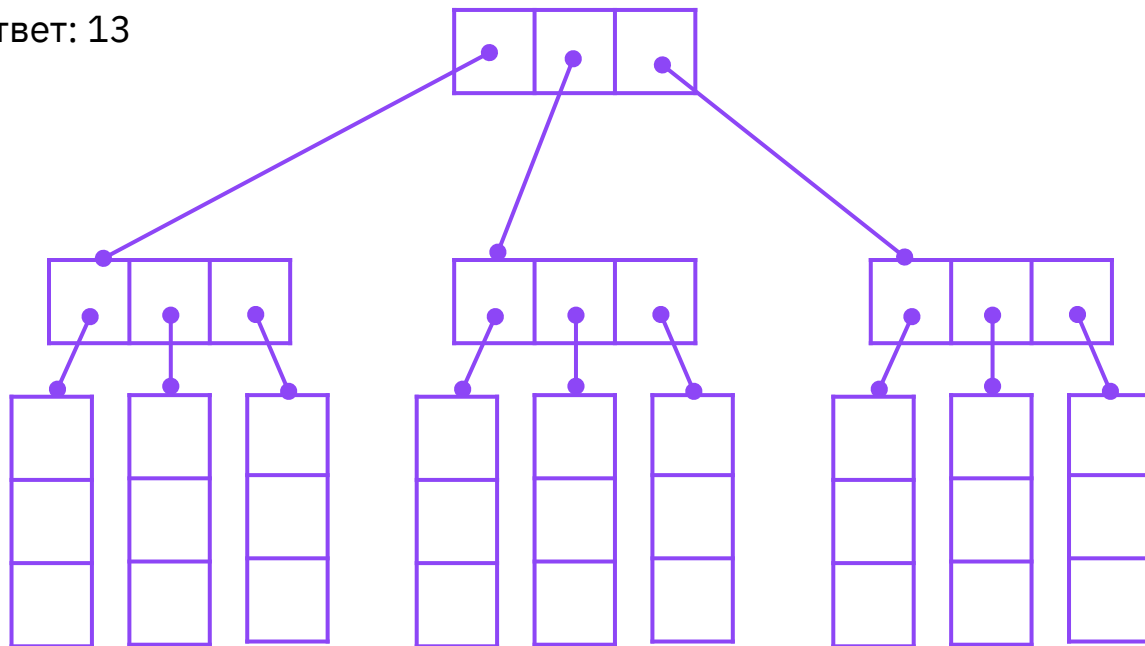
Сколько будет создано одномерных массивов при инициализации массива 3x3x3?



Сколько будет создано одномерных массивов при инициализации массива 3x3x3?

Любой многомерный массив это массив, содержащий массивы.

Ответ: 13





Какой результат будет у сравнения на равенство:

$2 + 2 * 2 == 2 << 2 >> 1$?



Какой результат будет у сравнения на равенство:

$2 + 2 * 2 == 2 << 2 >> 1$?

$n << k = n * 2^k$

$n >> k = n / 2^k$



Какой результат будет у сравнения на равенство:

$2 + 2 * 2 == 2 << 2 >> 1$?

ответ: `false` ($6 \neq 4$)



Рассмотрение домашнего задания



Задание:

Написать метод «Шифр Цезаря», с булевым параметром зашифрования/расшифрования, и числовым ключом;

Решение:

```
4 private static String caesar(String in, int key, boolean encrypt) {  
5     if (in == null || in.isEmpty())  
6         return null;  
7  
8     final int len = in.length();  
9     char[] out = new char[len];  
10    for (int i = 0; i < len; ++i) {  
11        out[i] = (char) (in.charAt(i) + ((encrypt) ? key : -key));  
12    }  
13    return new String(out);  
14 }
```



Задание:

Написать метод, принимающий на вход массив чисел и параметр n. Метод должен осуществить циклический (последний элемент при сдвиге становится первым) сдвиг всех элементов массива на n позиций;

Решение:

```
1 private static void shifter(int[] a, int n) {
2     n %= a.length;
3     int shift = a.length + n;
4     shift %= a.length;
5
6     for (int i = 0; i < shift; i++) {
7         int temp = a[a.length - 1];
8         System.arraycopy(a, 0, a, 1, a.length - 1);
9         a[0] = temp;
10    }
11 }
```

```
1 int[] arr = {1,2,3,4,5};
2 System.out.println(Arrays.toString(arr));
3 shifter(arr, 2);
4 System.out.println(Arrays.toString(arr));
5 shifter(arr, -2);
6 System.out.println(Arrays.toString(arr));
7
```

✓ 0.9s

[1, 2, 3, 4, 5]

[4, 5, 1, 2, 3]

[1, 2, 3, 4, 5]



Задание:

Написать метод, которому можно передать в качестве аргумента массив, состоящий строго из единиц и нулей (целые числа типа `int`). Метод должен заменить единицы в массиве на нули, а нули на единицы и не содержать ветвлений. Написать как можно больше вариантов метода

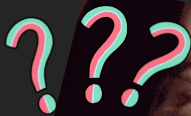
Решение:

```
66 @ private static void change(int[] a) {  
67     for (int i = 0; i < a.length; i++) {  
68         a[i] = 1 - a[i];  
69         // a[i] = (a[i] - 1) * -1;  
70         // a[i] = (a[i] + 1) % 2;  
71     }  
72 }
```



Вопросы?

Вопросы?



Вопросы?





Практика



Задание 1

Задача: Задать одномерный массив. Написать методы поиска в нём минимального и максимального элемента;



10 минут



Задание 1

Задача: Задать одномерный массив. Написать методы поиска в нём минимального и максимального элемента;

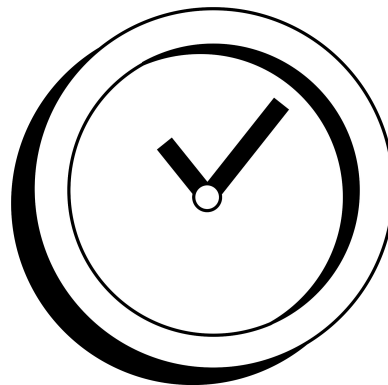
<<10:00->>

```
[1, 2, 0, 5, 7, 8]
```

```
min value = 0
```

```
max index = 5
```

```
Process finished with exit code 0
```





Задание 1

Задача: Задать одномерный массив.
Написать методы поиска в нём минимального
и максимального элемента;

Пример, на вашем экране →

```
19 @ private static int findMin(int[] a) { //min value
20     int min = a[0];
21     for (int i = 1; i < a.length; i++) {
22         if (a[i] < min) {
23             min = a[i];
24         }
25     }
26     return min;
27 }
28
29 @ private static int findMax(int[] a) { // max index
30     int max = 0;
31     for (int i = 1; i < a.length; i++) {
32         if (a[i] > a[max])
33             max = i;
34     }
35     return max;
36 }
```



Задание 1*

Задача: Привести функции к корректному виду и дополнительно написать ещё две функции так, чтобы получились (четыре) функции поиска минимального и максимального как значения, так и индекса.



Задание 2

Задача: Создать квадратный целочисленный массив (количество строк и столбцов одинаковое), заполнить его диагональные элементы единицами, используя цикл(ы)



10 минут

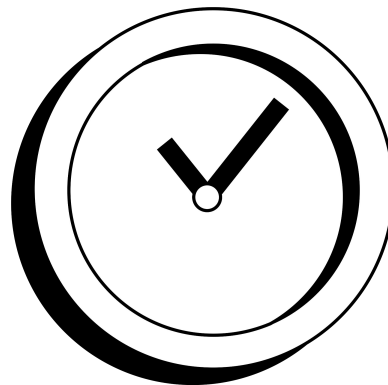


Задание 2

Задача: Создать квадратный целочисленный массив (количество строк и столбцов одинаковое), заполнить его диагональные элементы единицами, используя цикл(ы)

```
[1, 0, 0, 0, 0, 0, 1]
[0, 1, 0, 0, 0, 1, 0]
[0, 0, 1, 0, 1, 0, 0]
[0, 0, 0, 1, 0, 0, 0]
[0, 0, 1, 0, 1, 0, 0]
[0, 1, 0, 0, 0, 1, 0]
[1, 0, 0, 0, 0, 0, 1]
```

<<10:00->>





Задание 2

Задача: Создать квадратный целочисленный массив (количество строк и столбцов одинаковое), заполнить его диагональные элементы единицами, используя цикл(ы)

Пример, на вашем экране →

```
private static void fillDiagonal(int[][] a) {  
    for (int i = 0; i < a.length; i++) {  
        a[i][i] = 1;  
        a[i][a.length - 1 - i] = 1;  
    }  
}
```



Задание 2*

Задача: Допisać функцию вывода двумерного массива в консоль



Перерыв



<<5:00->>



Задание 3

Задача: Написать метод, в который передается не пустой одномерный целочисленный массив, метод должен вернуть true если в массиве есть место, в котором сумма левой и правой части массива равны. Примеры: `checkBalance([1, 1, 1, || 2, 1]) → true`,

`checkBalance([2, 1, 1, 2, 1]) → false`,

`checkBalance([10, || 1, 2, 3, 4]) → true`.

Абстрактная граница показана символами ||, эти символы в массив не входят.



20 минут



Задание 3

Задача: Написать метод, в который передается не пустой одномерный целочисленный массив, метод должен вернуть true если в массиве есть место, в котором сумма левой и правой части массива равны. Примеры:

`checkBalance([1, 1, 1, || 2, 1]) → true,`

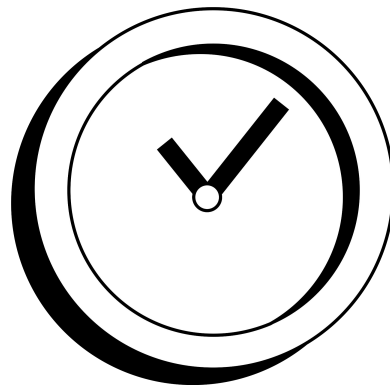
`checkBalance([2, 1, 1, 2, 1]) → false,`

`checkBalance([10, || 1, 2, 3, 4]) → true.`

Абстрактная граница показана символами ||, эти символы в массив не входят.

```
[1, 1, 1, 2, 1] = true  
[2, 1, 1, 2, 1] = false  
[10, 1, 2, 3, 4] = true
```

< <20:00-> >





Задание 3

Задача: Написать метод, в который передается не пустой одномерный целочисленный массив, метод должен вернуть true если в массиве есть место, в котором сумма левой и правой части массива равны.

Примеры:

checkBalance([1, 1, 1, || 2, 1]) → true,

checkBalance([2, 1, 1, 2, 1]) → false,

checkBalance([10, || 1, 2, 3, 4]) → true.

Абстрактная граница показана символами ||, эти символы в массив не входят.

Пример, на вашем экране →

```
private static boolean checkBalance2(int[] a) {  
    int sum = 0;  
    for (int i = 0; i < a.length; i++) {  
        sum += a[i];  
    }  
    if (sum % 2 != 0) return false;  
    int left = 0;  
    for (int i = 0; i < a.length; i++) {  
        left += a[i];  
        sum -= a[i];  
        if (left == sum) return true;  
    }  
    return false;  
}
```



Задание 3*

Задача: написать этот же метод таким образом, чтобы в нём использовался только один цикл.



Задание 4

Задача: Написать функцию добавления элемента в конец массива таким образом, чтобы она расширяла массив при необходимости.

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

15 минут

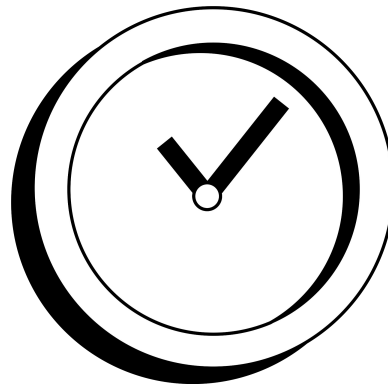


Задание 4

Задача: Написать функцию добавления элемента в конец массива таким образом, чтобы она расширяла массив при необходимости.

<<15:00->>

```
1 int[] add(int[] arr, int current, int value) {
2     if (current == arr.length) {
3         int[] temp = new int[arr.length * 2];
4         System.arraycopy(arr, 0, temp, 0, arr.length);
5         arr = temp;
6     }
7     arr[current++] = value;
8     return arr;
9 }
```





Задание 4

Задача: Написать функцию добавления элемента в конец массива таким образом, чтобы она расширяла массив при необходимости.

Пример, на вашем экране →

```
1  int[] array = {1,2};
2  int size = 2;
3
4  System.out.println(size + " = " + Arrays.toString(array));
5  array = add(array, size++, 6);
6  System.out.println(size + " = " + Arrays.toString(array));
7  array = add(array, size++, 6);
8  System.out.println(size + " = " + Arrays.toString(array));
9  array = add(array, size++, 6);
10 System.out.println(size + " = " + Arrays.toString(array));
11 array = add(array, size++, 6);
12 System.out.println(size + " = " + Arrays.toString(array));
13
```

2 = [1, 2]

3 = [1, 2, 6, 0]

4 = [1, 2, 6, 6]

5 = [1, 2, 6, 6, 6, 0, 0, 0]

6 = [1, 2, 6, 6, 6, 6, 0, 0]



Задание 4*

Задача: Функция должна возвращать ссылку на вновь созданный внутри себя массив, а не использовать глобальный.



Задание 5 (необязательное)

Задача: Написать метод, осуществляющий сортировку одномерного массива подсчётом. Важное ограничение состоит в том, что для этой сортировки диапазон значений исходного массива должен находиться в разумных пределах, например, не более 1000.

$x[2, 1, 0, 4, 3, 0, 0, 1, 2] \rightarrow t[3(x_0), 2(x_1), 2(x_2), 1(x_3), 1(x_4)] \rightarrow x[0, 0, 0, 1, 1, 2, 2, 3, 4]$



20 минут



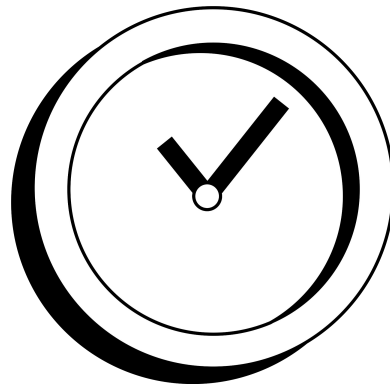
Задание 5 (необязательное)

Задача: Написать метод, осуществляющий сортировку одномерного массива подсчётом. Важное ограничение состоит в том, что для этой сортировки диапазон значений исходного массива должен находиться в разумных пределах, например, не более 1000.

$x[2, 1, 0, 4, 3, 0, 0, 1, 2] \rightarrow t[3(x_0), 2(x_1), 2(x_2), 1(x_3), 1(x_4)] \rightarrow x[0, 0, 0, 1, 1, 2, 2, 3, 4]$

```
[2, 1, 0, 4, 3, 0, 0, 1, 2]
[0, 0, 0, 1, 1, 2, 2, 3, 4]
```

<<20:00->>





Задание 5 (необязательное)

Задача: Написать метод, осуществляющий сортировку одномерного массива подсчётом. Важное ограничение состоит в том, что для этой сортировки диапазон значений исходного массива должен находиться в разумных пределах, например, не более 1000.

$x[2, 1, 0, 4, 3, 0, 0, 1, 2] \rightarrow$

$t[3(x_0), 2(x_1), 2(x_2), 1(x_3), 1(x_4)] \rightarrow$

$x[0, 0, 0, 1, 1, 2, 2, 3, 4]$

Пример, на вашем экране \rightarrow

```
private static void pigeon(int[] arr) {
    final int min = getMin(arr);
    final int max = getMax(arr);
    int[] freq = new int[max - min + 1];
    for (int i = 0; i < arr.length; i++)
        freq[arr[i] - min]++;

    int arrIndex = 0;
    for (int i = 0; i < freq.length; i++)
        for (int elems = freq[i]; elems > 0; elems-- )
            arr[arrIndex++] = i + min;
}
```



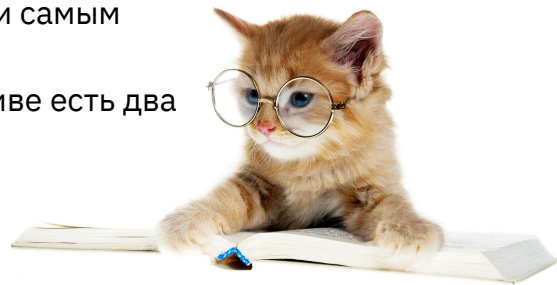
Домашнее задание



Домашнее задание

Задачи:

1. Решить все задания (в том числе «со звёздочкой»), если они не были решены на семинаре, без ограничений по времени;
2. Написать метод, возвращающий количество чётных элементов массива.
`countEvens([2, 1, 2, 3, 4]) → 3`
`countEvens([2, 2, 0]) → 3`
`countEvens([1, 3, 5]) → 0`
3. Написать функцию, возвращающую разницу между самым большим и самым маленьким элементами переданного не пустого массива.
4. Написать функцию, возвращающую истину, если в переданном массиве есть два соседних элемента, с нулевым значением.





Вопросы?



Вопросы?

Вопросы?





Спасибо за внимание!

