**Лабораторная работа 11.**

**Класс Arrays**

*Класс java.util.Arrays содержит полезные методы для распространенных операций с массивами, таких как сортировка и поиск.*

Класс**java.util.Arrays** содержит различные статические методы сортировки массива, поиска в массиве, сравнения массивов, заполнения элементов массива и возвращения строкового представления массива. Эти методы перегружены для всех примитивных типов.

Для сортировки всего или части массива можно использовать метод **sort()** или **parallelSort()**. Например, следующий код сортирует массив чисел и массив символов:

double[] numbers = {6.0, 4.4, 1.9, 2.9, 3.4, 3.5};

java.util.Arrays.sort(numbers); *// Сортирует весь массив*

java.util.Arrays.parallelSort(numbers); *// Сортирует весь массив*

char[] chars = {'a', 'A', '4', 'F', 'D', 'P'};

java.util.Arrays.sort(chars, 1, 3); *// Сортирует часть массива*

java.util.Arrays.parallelSort(chars, 1, 3); *// Сортирует часть массива*

При вызове функции **sort(numbers)** сортируются все числа массива. При вызове функции **sort(chars, 1, 3)** сортируется часть массива, начиная с **chars[1]** и заканчивая **chars[3-1]**. Если компьютер имеет несколько процессоров, то метод **parallelSort()** является более эффективным.

Для поиска ключа в массиве можно использовать метод **binarySearch()**. Массив должен быть предварительно отсортирован в возрастающем порядке. Если ключ в массиве не найден, то этот метод возвращает ‑**(insertionIndex + 1)**. Например, следующий код ищет ключи в массиве целых чисел и массиве символов:

int[] list = {2, 4, 7, 10, 11, 45, 50, 59, 60, 66, 69, 70, 79};

System.out.println("1. Индекс равен " +

java.util.Arrays.binarySearch(list, 11));

System.out.println("2. Индекс равен " +

java.util.Arrays.binarySearch(list, 12));

char[] chars = {'a', 'c', 'g', 'x', 'y', 'z'};

System.out.println("3. Индекс равен " +

java.util.Arrays.binarySearch(chars, 'a'));

System.out.println("4. Индекс равен " +

java.util.Arrays.binarySearch(chars, 't'));

Выходные данные этого кода будут выглядеть следующим образом:

1. Индекс равен 4

2. Индекс равен -6

3. Индекс равен 0

4. Индекс равен -4

Чтобы проверить, являются ли два массива строго равными друг другу, можно использовать метод **equals()**. Два массива строго равны, если равны их соответствующие элементы. В следующем коде **list1** и **list2** равны, а **list2** и **list3** — не равны.

int[] list1 = {2, 4, 7, 10};

int[] list2 = {2, 4, 7, 10};

int[] list3 = {4, 2, 7, 10};

System.out.println(java.util.Arrays.equals(list1, list2)); *// true*

System.out.println(java.util.Arrays.equals(list2, list3)); *// false*

Для заполнения всего массива или его части можно использовать метод **fill()**. Например, следующий код заполняет массив **list1** значением **5**, а часть массива, начиная с **list2[1]** и заканчивая **list2[5-1]**, заполняет значением **8**.

int[] list1 = {2, 4, 7, 10};

int[] list2 = {2, 4, 7, 7, 7, 10};

java.util.Arrays.fill(list1, 5); *// Заполняет значением 5 весь массив*

java.util.Arrays.fill(list2, 1, 5, 8); *// Заполняет значением 8 часть массива*

Для возвращения строки, представляющей все элементы массива, можно использовать метод **toString()**. Это быстрый и простой способ отображения всех элементов массива. Например, следующий код:

int[] list = {2, 4, 7, 10};

System.out.println(java.util.Arrays.toString(list));

отображает

[2, 4, 7, 10].

**Задание 1.**

Что будет отображено в консоли следующим фрагментом кода?

Напишите программу и подтвердите свой ответ.

int[] list1 = {2, 4, 7, 10};

java.util.Arrays.fill(list1, 7);

System.out.println(java.util.Arrays.toString(list1));

int[] list2 = {2, 4, 7, 10};

System.out.println(java.util.Arrays.toString(list2));

System.out.print(java.util.Arrays.equals(list1, list2));

Ваш ответ на задание 1.

import java.util.Arrays;

public class Main{

public static void main(String[] args){

int[] list1 = {2, 4, 7, 10};

java.util.Arrays.fill(list1, 7);

System.out.println(java.util.Arrays.toString(list1));

int[] list2 = {2, 4, 7, 10};

System.out.println(java.util.Arrays.toString(list2));

System.out.print(java.util.Arrays.equals(list1, list2));

}

}

Отображено будет

[7, 7, 7, 7]

[2, 4, 7, 10]

false

\*\* Process exited - Return Code: 0 \*\*