Uniwersytet Jana Długosza w Częstochowie

Mykhailo Hulii

Studia stacjonarne 1 stopnia,

2 rok informatyka, grupa 1

Zadanie 0

```java
private static void task0() {
    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter the number of integers you want to input: ");
    int size = scanner.nextInt();
    int[] numbers = new int[size];

    for (int i = 0; i < size; i++) {
        System.out.print("Enter integer #" + (i + 1) + ": ");
        numbers[i] = scanner.nextInt();
    }

    System.out.println("\nEntered numbers:");

    for (int i = 0; i < size; i++) {
        System.out.println("Integer #" + (i + 1) + ": " + numbers[i]);
    }

    System.out.print("\nEnter the new size of the array: ");
    int newSize = scanner.nextInt();
    int[] newNumbers = new int[newSize];

    System.arraycopy(numbers, srcPos: 0, newNumbers, destPos: 0, Math.min(size, newSize));
    numbers = newNumbers;

    System.out.println("\nArray size changed to " + newSize + ".");
    System.out.println("\nNumbers in the updated array:");

    for (int i = 0; i < newSize; i++) {
        System.out.println("Integer #" + (i + 1) + ": " + numbers[i]);
    }
}
```

```
Enter the number of integers you want to input: 4
Enter integer #1: 1
Enter integer #2: 2
Enter integer #3: 3
Enter integer #4: 4

Entered numbers:
Integer #1: 1
Integer #2: 2
Integer #3: 3
Integer #4: 4

Enter the new size of the array: 2

Array size changed to 2.

Numbers in the updated array:
Integer #1: 1
Integer #2: 2
```

Zadanie A1

```java
private static void task1() {
    int size = 10;
    int[][] multiplicationTable = new int[size][size];

    for (int i = 1; i <= size; i++) {
        for (int j = 1; j <= size; j++) {
            multiplicationTable[i - 1][j - 1] = i * j;
        }
    }

    System.out.println("Multiplication Table:");

    for (int i = 0; i < size; i++) {
        for (int j = 0; j < size; j++) {
            System.out.printf("%4d", multiplicationTable[i][j]);
        }
        System.out.println();
    }
}
```

```
Multiplication Table:
   1   2   3   4   5   6   7   8   9  10
   2   4   6   8  10  12  14  16  18  20
   3   6   9  12  15  18  21  24  27  30
   4   8  12  16  20  24  28  32  36  40
   5  10  15  20  25  30  35  40  45  50
   6  12  18  24  30  36  42  48  54  60
   7  14  21  28  35  42  49  56  63  70
   8  16  24  32  40  48  56  64  72  80
   9  18  27  36  45  54  63  72  81  90
  10  20  30  40  50  60  70  80  90 100
```

Zadanie 2

```java
private static void task2() {
    int[] numbers = new int[20];
    Random random = new Random();
    for (int i = 0; i < 20; i++) {
        numbers[i] = random.nextInt( bound: 100);
    }
    System.out.println("Original Array: " + Arrays.toString(numbers));
    for (int i = 0, j = numbers.length - 1; i < j; i++, j--) {
        int temp = numbers[i];
        numbers[i] = numbers[j];
        numbers[j] = temp;
    }

    System.out.println("Reversed Array: " + Arrays.toString(numbers));

    int sum = 0;
    for (int number : numbers) {
        sum += number;
    }

    System.out.println("Sum of Elements: " + sum);
    Arrays.sort(numbers);
    System.out.println("Sorted Array: " + Arrays.toString(numbers));
    Scanner scanner = new Scanner(System.in);
    System.out.print("Enter a value to find its occurrences: ");

    int userValue = scanner.nextInt();
    int occurrences = 0;
    for (int number : numbers) {
        if (number == userValue) {
            occurrences++;
        }
    }
    System.out.println("Occurrences of " + userValue + ": " + occurrences);
    int[] increasedValuesCopy = Arrays.copyOf(numbers, numbers.length);
    for (int i = 0; i < increasedValuesCopy.length; i++) {
        increasedValuesCopy[i]++;
    }
    System.out.println("Copy with Increased Values: " + Arrays.toString(increasedValuesCopy));
}
```

```
Original Array: [40, 48, 83, 81, 14, 57, 9, 42, 31, 19, 50, 42, 59, 26, 24, 66, 12, 98, 2, 59]
Reversed Array: [59, 2, 98, 12, 66, 24, 26, 59, 42, 50, 19, 31, 42, 9, 57, 14, 81, 83, 48, 40]
Sum of Elements: 862
Sorted Array: [2, 9, 12, 14, 19, 24, 26, 31, 40, 42, 42, 48, 50, 57, 59, 59, 66, 81, 83, 98]
Enter a value to find its occurrences: 42
Occurrences of 42: 2
Copy with Increased Values: [3, 10, 13, 15, 20, 25, 27, 32, 41, 43, 43, 49, 51, 58, 60, 60, 67, 82, 84, 99]
```

Zadanie 3

```java
private static void task3() {
    Scanner scanner = new Scanner(System.in);

    char selectedOption;

    System.out.println("Menu Selection:");
    System.out.println("a. Option A");
    System.out.println("b. Option B");
    System.out.println("c. Option C");
    System.out.println("d. Option D");
    System.out.println("e. Option E");

    System.out.print("Enter a letter from the menu (a - e or A - E): ");
    String input = scanner.next();

    if (input.length() == 1) {
        selectedOption = input.charAt(0);

        if ((selectedOption >= 'a' && selectedOption <= 'e') || (selectedOption >= 'A' && selectedOption <= 'E')) {
            int asciiCode = (int) selectedOption;
            String letterCase = selectedOption >= 'a' ? "lowercase" : "uppercase";

            System.out.println("You selected: " + selectedOption);
            System.out.println("ASCII code: " + asciiCode);
            System.out.println("Letter case: " + letterCase);
        } else {
            System.out.println("Invalid selection. Please choose a letter from the menu.");
        }
    } else {
        System.out.println("Invalid input. Please enter only one character.");
    }
}
```

```
Menu Selection:
a. Option A
b. Option B
c. Option C
d. Option D
e. Option E
Enter a letter from the menu (a - e or A - E): A
You selected: A
ASCII code: 65
Letter case: uppercase
```

```
Menu Selection:
a. Option A
b. Option B
c. Option C
d. Option D
e. Option E
Enter a letter from the menu (a - e or A - E): a
You selected: a
ASCII code: 97
Letter case: lowercase
```

Zadanie 4

```java
private static void task4() {
    Scanner scanner = new Scanner(System.in);

    System.out.println("Enter a sentence (press Enter to finish):");
    String userInput = scanner.nextLine();

    int letterCount = 0;
    int digitCount = 0;
    int spaceCount = 0;

    for (char ch : userInput.toCharArray()) {
        if (Character.isLetter(ch)) {
            letterCount++;
        } else if (Character.isDigit(ch)) {
            digitCount++;
        } else if (Character.isWhitespace(ch)) {
            spaceCount++;
        }
    }

    System.out.println("\nCharacter counts:");
    System.out.println("Letters: " + letterCount);
    System.out.println("Digits: " + digitCount);
    System.out.println("Spaces: " + spaceCount);
}
```

```
Enter a sentence (press Enter to finish):
Hello, World!1

Character counts:
Letters: 10
Digits: 1
Spaces: 1
```

Zadanie 5

```java
private static void task5() {
    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter a text to reverse: ");
    String userInput = scanner.nextLine();

    String reversedText = "";

    for (int i = userInput.length() - 1; i >= 0; i--) {
        reversedText += userInput.charAt(i);
    }

    System.out.println("\nReversed Text: " + reversedText);

}
```

```
Enter a text to reverse: Hello, World!

Reversed Text: !dlroW ,olleH
```

Zadanie 6

```java
private static void task6() {
    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter a text: ");
    String text = scanner.nextLine();

    System.out.print("Enter a phrase to search for: ");
    String phrase = scanner.nextLine();

    int position = -1;

    for (int i = 0; i <= text.length() - phrase.length(); i++) {
        boolean found = true;

        for (int j = 0; j < phrase.length(); j++) {
            if (text.charAt(i + j) != phrase.charAt(j)) {
                found = false;
                break;
            }
        }

        if (found) {
            position = i;
            break;
        }
    }

    if (position != -1) {
        System.out.println("\nThe phrase '" + phrase + "' is found in the text.");
        System.out.println("It starts at position: " + position);
    } else {
        System.out.println("\nThe phrase '" + phrase + "' is not found in the text.");
    }
}
```

```
Enter a text: Hello, World!1
Enter a phrase to search for: ,

The phrase ',' is found in the text.
It starts at position: 5
```

Zadanie 7

```java
private static void task7() {
    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter the first string: ");
    String firstString = scanner.nextLine();

    System.out.print("Enter the second string: ");
    String secondString = scanner.nextLine();

    System.out.println("\nResult using + operator: " + firstString + secondString);
    System.out.println("Result using concat() method: " + firstString.concat(secondString));
}
```

```
Enter the first string: Hello,
Enter the second string:  World!

Result using + operator: Hello, World!
Result using concat() method: Hello, World!
```

Zadanie 8

```java
private static void task8() {
    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter an integer value between 0 and 1000: ");

    int inputValue = scanner.nextInt();

    if (inputValue < 0 || inputValue > 1000) {
        System.out.println("Invalid input. Please enter an integer between 0 and 1000.");
    } else {
        System.out.println("\nOctal representation: " + Integer.toOctalString(inputValue));
        System.out.println("Hexadecimal representation: " + Integer.toHexString(inputValue));
    }
}
```

```
Enter an integer value between 0 and 1000: 34

Octal representation: 42
Hexadecimal representation: 22
```

Zadanie 9

```java
private static void task9() {
    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter a text with whitespace characters: ");
    String inputText = scanner.nextLine();

    StringBuilder resultText = new StringBuilder();

    for (int i = 0; i < inputText.length(); i++) {
        char currentChar = inputText.charAt(i);

        if (!isWhitespace(currentChar)) {
            resultText.append(currentChar);
        }
    }

    System.out.println("\nText without whitespace characters: " + resultText);
}
1 usage
private static boolean isWhitespace(char ch) {
    return ch == ' ' || ch == '\t' || ch == '\n' || ch == '\r' || ch == '\f';
}
```

```
Enter a text with whitespace characters: Hello, World!

Text without whitespace characters: Hello,World!
```

Zadanie 10

```java
private static void task10() {
    for (int i = 1; i <= 32; i++) {
        BigDecimal result = BigDecimal.ONE.divide(BigDecimal.valueOf(i), scale: 32, BigDecimal.ROUND_HALF_UP);
        System.out.println("1 / " + i + " = " + result);
    }
}
```

```
1 / 1 = 1.00000000000000000000000000000000
1 / 2 = 0.50000000000000000000000000000000
1 / 3 = 0.33333333333333333333333333333333
1 / 4 = 0.25000000000000000000000000000000
1 / 5 = 0.20000000000000000000000000000000
1 / 6 = 0.16666666666666666666666666666667
1 / 7 = 0.14285714285714285714285714285714
1 / 8 = 0.12500000000000000000000000000000
1 / 9 = 0.11111111111111111111111111111111
1 / 10 = 0.10000000000000000000000000000000
1 / 11 = 0.09090909090909090909090909090909
1 / 12 = 0.08333333333333333333333333333333
1 / 13 = 0.07692307692307692307692307692308
1 / 14 = 0.07142857142857142857142857142857
1 / 15 = 0.06666666666666666666666666666667
1 / 16 = 0.06250000000000000000000000000000
1 / 17 = 0.05882352941176470588235294117647
1 / 18 = 0.05555555555555555555555555555556
1 / 19 = 0.05263157894736842105263157894737
1 / 20 = 0.05000000000000000000000000000000
1 / 21 = 0.04761904761904761904761904761905
1 / 22 = 0.04545454545454545454545454545455
1 / 23 = 0.04347826086956521739130434782609
1 / 24 = 0.04166666666666666666666666666667
1 / 25 = 0.04000000000000000000000000000000
1 / 26 = 0.03846153846153846153846153846154
1 / 27 = 0.03703703703703703703703703703704
1 / 28 = 0.03571428571428571428571428571429
1 / 29 = 0.03448275862068965517241379310345
1 / 30 = 0.03333333333333333333333333333333
1 / 31 = 0.03225806451612903225806451612903
1 / 32 = 0.03125000000000000000000000000000
```

Zadanie 12

```java
private static void task12() {
    int[][] array = new int[10][10];

    fillArrayLeftDiagonal(array);
    fillArrayRightDiagonal(array);

    System.out.println("Filled Array:");
    printArray(array);
}
1 usage
private static void fillArrayLeftDiagonal(int[][] array) {
    for (int i = 0; i < array.length; i++) {
        for (int j = 0; j <= i; j++) {
            array[i][j] = i + 1;
        }
    }
}
1 usage
private static void fillArrayRightDiagonal(int[][] array) {
    for (int i = 0; i < array.length; i++) {
        for (int j = i + 1; j < array[i].length; j++) {
            array[i][j] = array.length - i;
        }
    }
}
2 usages
private static void printArray(int[][] array) {
    for (int[] row : array) {
        for (int value : row) {
            System.out.printf("%4d", value);
        }
        System.out.println();
    }
}
```

```
Filled Array:
   1  10  10  10  10  10  10  10  10  10
   2   2   9   9   9   9   9   9   9   9
   3   3   3   8   8   8   8   8   8   8
   4   4   4   4   7   7   7   7   7   7
   5   5   5   5   5   6   6   6   6   6
   6   6   6   6   6   6   5   5   5   5
   7   7   7   7   7   7   7   4   4   4
   8   8   8   8   8   8   8   8   3   3
   9   9   9   9   9   9   9   9   9   2
  10  10  10  10  10  10  10  10  10  10
```

Zadanie 13

```java
private static void task13() {
    final int length = 10;
    int[][] array = new int[length][length];

    printArray(array);
    System.out.println();

    for (int i = 0; i < array.length; i++) {
        int row = i;
        int col = array.length - 1 - i;

        array[row][col] = 1;
    }

    printArray(array);
}
```

```
  0   0   0   0   0   0   0   0   0   0
  0   0   0   0   0   0   0   0   0   0
  0   0   0   0   0   0   0   0   0   0
  0   0   0   0   0   0   0   0   0   0
  0   0   0   0   0   0   0   0   0   0
  0   0   0   0   0   0   0   0   0   0
  0   0   0   0   0   0   0   0   0   0
  0   0   0   0   0   0   0   0   0   0
  0   0   0   0   0   0   0   0   0   0
  0   0   0   0   0   0   0   0   0   0

  0   0   0   0   0   0   0   0   0   1
  0   0   0   0   0   0   0   0   1   0
  0   0   0   0   0   0   0   1   0   0
  0   0   0   0   0   0   1   0   0   0
  0   0   0   0   0   1   0   0   0   0
  0   0   0   0   1   0   0   0   0   0
  0   0   0   1   0   0   0   0   0   0
  0   0   1   0   0   0   0   0   0   0
  0   1   0   0   0   0   0   0   0   0
  1   0   0   0   0   0   0   0   0   0
```