NAME : MUCHAMMAD NABIL HAYKAL WIDARTO
CLASS : SIB-1G
NIM : 2341760152

JOBSHEET 9

https://github.com/muchnabil/daspro-jobsheet9

EXPERIMENT

1.
```java
public class ArrayNumber14 {
    Run | Debug
    public static void main(String[] args) {

        int[] num = new int[4];
        num[0] = 5;
        num[1] = 12;
        num[2] = 7;
        num[3] = 20;

        System.out.println(num[0]);
        System.out.println(num[1]);
        System.out.println(num[2]);
        System.out.println(num[3]);
    }
}
```

2.

```java
import java.util.Scanner;
public class ArrayValue14 {
    Run | Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int[] finalscore = new int [10];

        for (int i = 0; i < 10; i++) {
            System.out.print("Enter the final score " + i + " : ");
            finalscore[i] = sc.nextInt();
        }
        for (int i = 0; i < 10; i++) {
            System.out.println("Final score " + i + " is " + finalscore[i]);
        }
    }
}
```

3.

```java
import java.util.Scanner;
public class ArrayAverageScore14 {
    Run | Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int[] score = new int[10];
        double total = 0;
        double average;

        for (int i = 0; i < score.length; i++) {
            System.out.print("Enter student score " + (i + 1) + " : ");
            score[i] = sc.nextInt();
        }
        for (int i = 0; i < score.length; i++) {
            total += score[i];
        }
        average = total / score.length;
        System.out.println("The class average score is " + average);
    }
}
```

4.

```java
public class LinearSearch14 {
    Run | Debug
    public static void main(String[] args) {
        int[] arrayInt = { 34, 18, 26, 48, 72, 20, 56, 63 };
        int key = 20;
        int result = 0;

        for (int i = 0; i < arrayInt.length; i++) {
            if (arrayInt[i] == key) {
                result = i;
                break;
            }
        }
        System.out.println("The key in the array is located at index position " + result);
    }
}
```

QUESTION 2.1

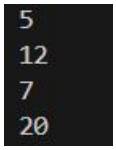
1. If the contents of each element of the array **num** are changed with numbers 5.0, 12867, 7.5, 2000000. What happens? How can it be like that?
2. Modify the program code by initializing the array elements at the same time when declaring the array.
3. Change the statement in step 6 to be like this

```
for (int i = 0; i < 4; i++) {
    System.out.println(num[i]);
}
```

   What is the result? How can it be like that?
4. If the condition in the for-loop statement is changed to i <= 4, what is the output of the program? Why is the result like that?
5. Commit and push the changes to GitHub.

ANSWER

1. Error because the data type used is int[], while the value to be entered must use the double[] data type.

3. 

the result is the same, but the process is different, this process uses looping to make the program shorter

4.

```
5
12
7
20
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index 4 out of bounds for length 4
        at ArrayNumber14.main(ArrayNumber14.java:12)
```

error results because the declared limit is only 4.

QUESTION 2.2

1. Change the statement in step 5 to be like this

```
for (int i = 0; i < finalScore.length; i++) {
    System.out.print("Enter the final score " + i + ": ");
    finalScore[i] = sc.nextInt();
}
```

Run the program. Have there been any changes? How can it be like that?

2. Apa yang dimaksud dengan kondisi **i < finalScore.length**?
3. Change the statement in step 6 to be like this, so that the program only displays the grades of students who passed, students who have a score > 70

```
for (int i = 0; i < finalScore.length; i++) {
    if (finalScore[i] > 70) {
        System.out.println("Student " + i + " Passed!");
    }
}
```

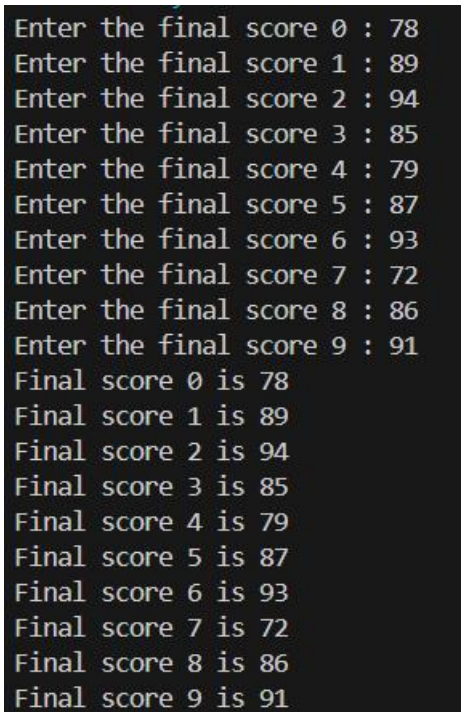
Run the program and describe the flow of the program!

4. Modify the program so that it displays all students, and mark which one passed, and which did not!

```
Enter the final score 0: 87
Enter the final score 1: 65
Enter the final score 2: 78
Enter the final score 3: 95
Enter the final score 4: 92
Enter the final score 5: 58
Enter the final score 6: 89
Enter the final score 7: 67
Enter the final score 8: 85
Enter the final score 9: 78
Student 0 Passed!
Student 1 Failed!
Student 2 Passed!
Student 3 Passed!
Student 4 Passed!
Student 5 Failed!
Student 6 Passed!
Student 7 Failed!
Student 8 Passed!
Student 9 Passed!
```

5. Commit and push the changes to GitHub.

ANSWER

1. 

The results are the same but the method is shorter and easier to understand.

2. *finalScore.length* is a property in Java that gives the length (number of elements) of the finalScore array. So, if finalScore contains 5 elements, *finalScore.length* will be 5

```
Enter the final score 0 : 78
Enter the final score 1 : 89
Enter the final score 2 : 94
Enter the final score 3 : 85
Enter the final score 4 : 79
Enter the final score 5 : 87
Enter the final score 6 : 93
Enter the final score 7 : 72
Enter the final score 8 : 86
Enter the final score 9 : 91
Student 0 Passed!
Student 1 Passed!
Student 2 Passed!
Student 3 Passed!
Student 4 Passed!
Student 5 Passed!
Student 6 Passed!
Student 7 Passed!
Student 8 Passed!
Student 9 Passed!
```

3. The final score of all grades is declared passed because the passing limit is > 70

4.
```java
import java.util.Scanner;
public class ArrayValue14 {
    Run | Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int[] finalscore = new int [10];

        for (int i = 0; i < finalscore.length; i++) {
            System.out.print("Enter the final score " + i + " : ");
            finalscore[i] = sc.nextInt();
        }
        for (int i = 0; i < finalscore.length; i++) {
            if (finalscore[i] > 70){
                System.out.println("Student " + i + " Passed!");
            }
            if (finalscore[i] < 70) {
            System.out.println("Student " + i + " Failed!");
            }
        }
    }
}

```

QUESTION 2.3

1. Modify the program in Experiment 3 so that the program can display the number of students who passed, students who have a score greater than 70 (>70)
2. Modify the program in Experiment 3 so that it can produce output like the following display

```
Enter the number of students: 5
Enter the final score 0: 81
Enter the final score 1: 76
Enter the final score 2: 90
Enter the final score 3: 68
Enter the final score 4: 63
The average score of students who passed is 82.33333333333333
The average score of students who failed is 65.5
```

3. Commit and push the changes to GitHub

ANSWER

1.
```
Enter student score 1 : 90
Enter student score 2 : 94
Enter student score 3 : 68
Enter student score 4 : 56
Enter student score 5 : 84
Enter student score 6 : 32
Enter student score 7 : 94
Enter student score 8 : 44
Enter student score 9 : 84
Enter student score 10 : 91
The class average score is 73.7
Number of students who graduated : 6.0
```

2.
```
Enter Total Student : 4
Enter student score 1 : 89
Enter student score 2 : 67
Enter student score 3 : 91
Enter student score 4 : 70
The average score of student who passed is : 90.0
The average score of student who failed is : 68.5
```

Question 2.4

Questions!

1. Explain the meaning of the **break;** statement on line 13 of the program code in Experiment 4.
2. Modify the program code in experiment 4 so that the program can receive input in the form of the number of array elements, the contents of the array, and the key you want to search for. Then, print to the screen the index of the element positions of the searched key. Example of program results:

```
Enter the number of array elements: 8
Enter the array element 0: 12
Enter the array element 1: 18
Enter the array element 2: -6
Enter the array element 3: 10
Enter the array element 4: 6
Enter the array element 5: 15
Enter the array element 6: 11
Enter the array element 7: 9
Enter the key you want to search for: 10
The key in the array is located at index position 3
```

3. Modify the program in experiment 4 so that the program will give the message "key not found" if the key is not in the array. Example of program results:

```
Enter the number of array elements: 6
Enter the array element 0: 19
Enter the array element 1: 23
Enter the array element 2: 29
Enter the array element 3: 31
Enter the array element 4: 37
Enter the array element 5: 43
Enter the key you want to search for: 11
Key not found
```

1. break; statement on line 13 serves to stop the search or loop after the searched value is found. For example, here the key is 20, the program will continue to run the loop until the compared value is equal to the the value of key or 20, when the value is the same the break statement will be executed, so the the loop stops.

2.
```
Enter the number of array elements : 8
Enter the array element 0 : 12
Enter the array element 1 : 18
Enter the array element 2 : -6
Enter the array element 3 : 10
Enter the array element 4 : 6
Enter the array element 5 : 15
Enter the array element 6 : 11
Enter the array element 7 : 9
Enter the key you want to search for : 10
The key in the array is located at index position 3
```

3.
```
Enter the number of array elements : 6
Enter the array element 0 : 19
Enter the array element 1 : 23
Enter the array element 2 : 29
Enter the array element 3 : 31
Enter the array element 4 : 37
Enter the array element 5 : 43
Enter the key you want to search for : 11
The key is not found in the array.
```

Assignment

1. Create a program to produce the highest value, lowest value, and average from an array containing integer type numbers.
   Terms:
   • Input: Number of elements, value of each element
   • Output: Highest value, lowest value, average value

1.

```java
import java.util.Scanner;
public class Assignment14 {
    Run | Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print(s:"Enter the number of element : ");
        int n = sc.nextInt();

        int[] number = new int [n];

        for (int i = 0; i < n; i++) {
            System.out.print("Enter the element " + (i + 1) + " : " );
            number[i] = sc.nextInt();
        }

        int highest = number[0];
        int lowest = number[0];
        int sum = 0;

        for (int i = 0; i < n; i++) {
            if(number[i] > highest) {
                highest = number[i];
            }
            if (number[i] < lowest) {
                lowest = number[i];
            }
            sum += number[i];
        }
    double average = sum / n;
    System.out.println("The highest value : " + highest);
    System.out.println("The lowest value : " + lowest);
    System.out.println("The average value : " + average);
    }
}
```

```
Enter the number of element : 5
Enter the element 1 : 80
Enter the element 2 : 86
Enter the element 3 : 90
Enter the element 4 : 70
Enter the element 5 : 56
The highest value : 90
The lowest value : 56
The average value : 76.0
```