Basic Programming Practicum
Jobsheet 4 Selection 1

Name
Dicha Zelianivan Arkana

NIM
2241720002

Class
1i

Department
Information Technology

Study Program
D4 Informatics Engineering

Contents

1 Laboratory

1.1 Experiment 1

1. Observe the flowchart

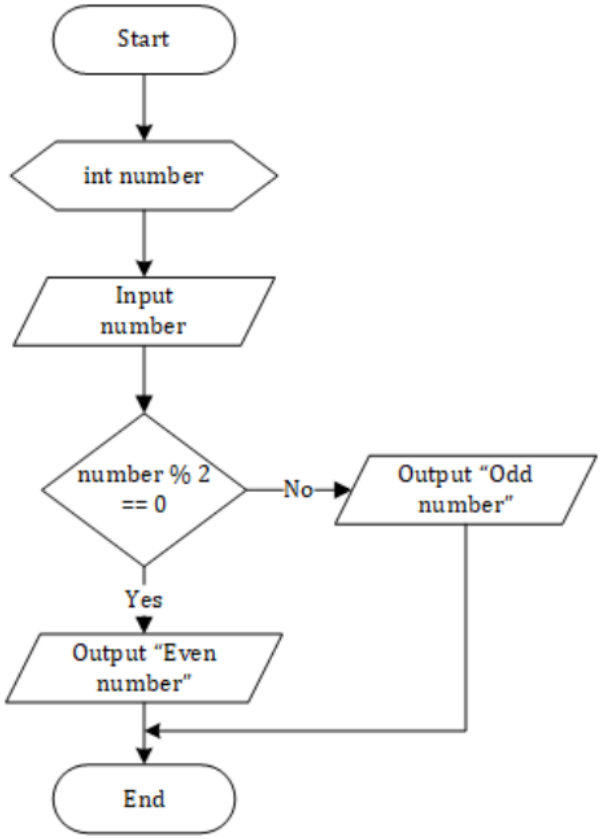


Figure 1: Flowchart

The flowchart is used to determine odd or even numbers, then we will make the program based on the flowchart.

2. Open a text editor. Create a new file, name it **Selection1.java**
3. Write the basic structure of the Java programming language which contains the `main()` function
4. Add the scanner library. Write the following code at the top **outside the class**

```
import java.util.Scanner;
```

5. Make a scanner declaration. Write the following code in **in the** `main()` function

```
Scanner input = new Scanner(System.in);
```

6. Create an `int` variable with the name **number**

```
int number;
```

7. Write down the syntax for entering the value from keyboard.

```
System.out.print("Enter a number: ");
number = input.nextInt();
```

8. Create a selection structure to check whether the number is even or odd

```
if (number % 2 == 0) {
    System.out.println("Even number");
} else {
    System.out.println("Odd number");
}
```

9. Compile and run the program. Observe the results!

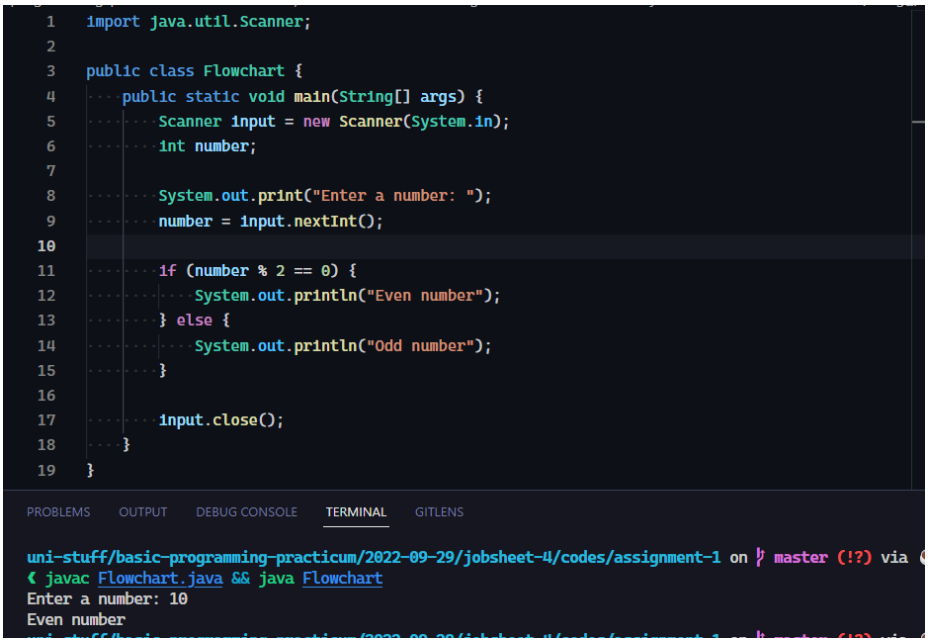


Figure 2: Implementation according to flowchart

Questions

1. Modify the program in its selection structure so that it becomes as follows:

```
String output = (number % 2 == 0) ? "Even numbers" : "Odd numbers";
System.out.println(output);
```

2. Compile, run, and observe the results!

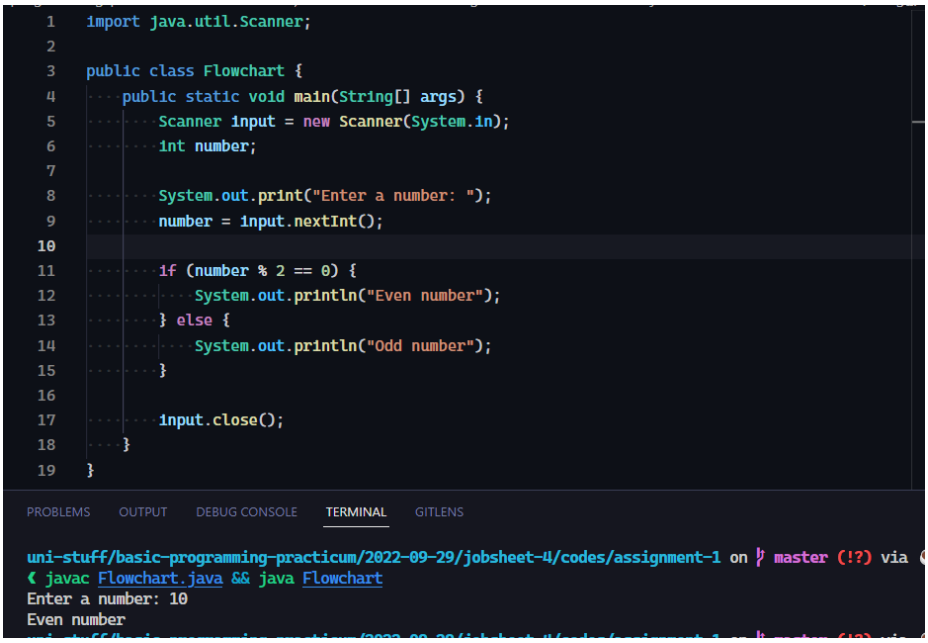


Figure 3: Implementation with ternary

3. Explain why the modified program output is the same as the program output before it was modified!

   Because we only replaced the if statement with a ternary but we don't change the condition. It will stay `true` when `number % 2 == 0`

1.2 Experiment 2

1. Open a text editor. Create a new file, named it **Selection2.java**
2. Write the basic structure of the Java programming language which contains the `main()` function
3. Add the **Scanner** library. Write the following code at the top **outside the class**

   ```
   Scanner input = new Scanner(System.in);
   ```

4. Create an `int` variable with the name **score**

   ```
   int score;
   ```

5. Write down the syntax for entering the value from keyboard

   ```
   System.out.print("Enter a score: ");
   score = input.nextInt();
   ```

6. Add the following selection structure

   ```
   if (score >= 100) {
       score += 10;
   } else {
       score -= 10;
   }
   System.out.println("The final score is " + score);
   ```

7. Compile and run the program. Observe the results!

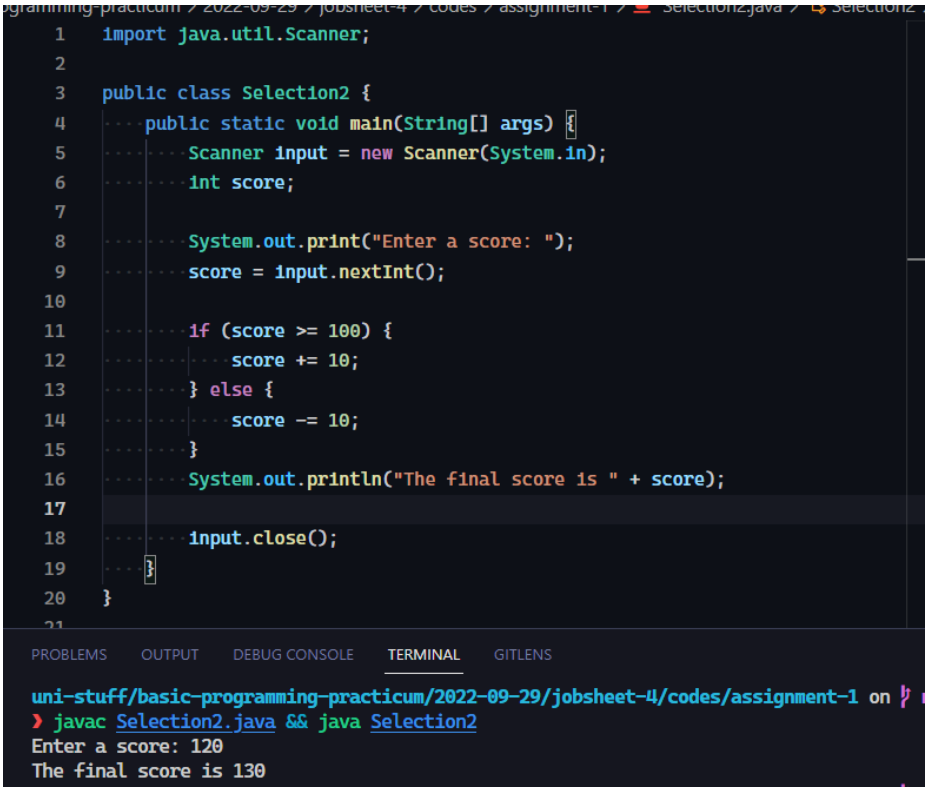


Figure 4: Selection 2 implementation

Questions

1. Describe the function of the following program code:

```
score += 10;
score -= 10;
```

They are called addition assignment and subtraction assignment. They are used to add or subtract and assign the new value add the same time. It is a shorthand for doing `score = score + 10` and `score = score - 10`

2. Modify the program so that only one input becomes two (for example: `score1` and `score2`). Then calculate the average of two values, if the average value is more than equal to 100 then subtract 5, whereas if the average value is less than 100 it will be printed immediately.

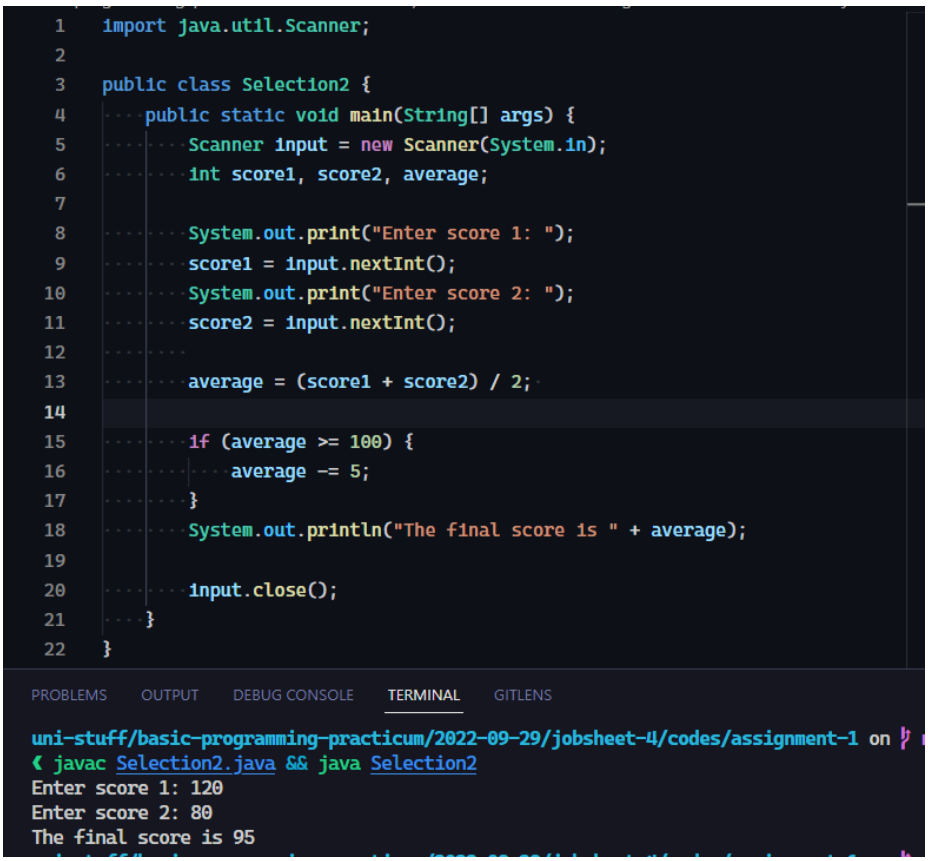


Figure 5: Selection 2 with average

1.3 Experiment 3

1. Open a text editor. Create a new file. Name it **Selection3.java**
2. Write the basic structure of Java programming language which contains the `main()` function
3. Add the **Scanner** library. Write the following code at the top **outside the class**

   ```
   import java.util.Scanner;
   ```

4. Make a Scanner declaration. Write the following code in the `main()` function

```
Scanner input = new Scanner(System.in);
```

5. Create an `int` variable with the name `age`

```
int age;
```

6. Write down the syntax for entering the value from keyboard:

```
System.out.print("Enter your age: ");
age = input.nextInt();
```

7. Add the following selection structure to check the age category

```
if (age > 65) {
    System.out.println("Elderly");
} else if (age > 18) {
    System.out.println("Adults");
} else if (age > 12) {
    System.out.println("Teen");
} else if (age > 5) {
    System.out.println("Children");
} else {
    System.out.println("Toddler");
}
```

8. Compile and run the program. Observe the results!

basic-programming-practicum > 2022-09-29 > jobsheet-4 > codes > assignment-1 > Selection3.java >

```java
import java.util.Scanner;

public class Selection3 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int age;

        System.out.print("Enter your age: ");
        age = input.nextInt();

        if (age > 65) {
            System.out.println("Elderly");
        } else if (age > 18) {
            System.out.println("Adults");
        } else if (age > 12) {
            System.out.println("Teen");
        } else if (age > 5) {
            System.out.println("Children");
        } else {
            System.out.println("Toddler");
        }

        input.close();
    }
}
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL GITLENS

```
uni-stuff/basic-programming-practicum/2022-09-29/jobsheet-4/codes/assignment-1 on
❮ javac Selection3.java && java Selection3
Enter your age: 18
Teen
```

Figure 6: Selection 3

Questions

1. How many conditions exist in experiment 3? Mention what conditions are!

   There are 5 conditions:

   - If the age is greater than 65, then print Elderly
   - If the age is greater than 18, then print Adults
   - If the age is greater than 12, then print Teen
   - if the age is greater than 5, then print Children
   - Otherwise, print Toddler

2. Modify the program so that if the age entered is 0 years or less than 0 it will display the output "Sorry, the age you entered is wrong"!

Figure 7: Selection 3 with less than equal 0 condition

1.4 Selection4

1. Open a text editor. Create a new file. Name it **Selection3.java**

2. Write the basic structure of Java programming language which contains the `main()` function

3. Add the **Scanner** library. Write the following code at the top **outside the class**

```
import java.util.Scanner;
```

4. Make a Scanner declaration. Write the following code in the `main()` function

```
Scanner input = new Scanner(System.in);
```

5. Create the following variables

```
double number1, number2, result;
char operator;
```

6. Write down the syntax for entering values from keyboard

```
System.out.print("Enter the first number: ");
number1 = input.nextDouble();
System.out.print("Enter the second number: ");
number2 = input.nextDouble();
System.out.print("Enter an operator (+ - * /): ");
operator = input.next().charAt(0);
```

7. Add the following selection structure

```
switch (operator) {
    case '+':
        result = number1 + number2;
        System.out.println(number1 + " + " + number2 + " = " + result);
        break;
    case '-':
        result = number1 + number2;
        System.out.println(number1 + " + " + number2 + " = " + result);
        break;
    case '*':
        result = number1 + number2;
        System.out.println(number1 + " + " + number2 + " = " + result);
        break;
    case '/':
        result = number1 + number2;
        System.out.println(number1 + " + " + number2 + " = " + result);
        break;
    default:
        System.out.println("The operator you entered is wrong");
}
```

8. Compile and run the program. Observe the results!

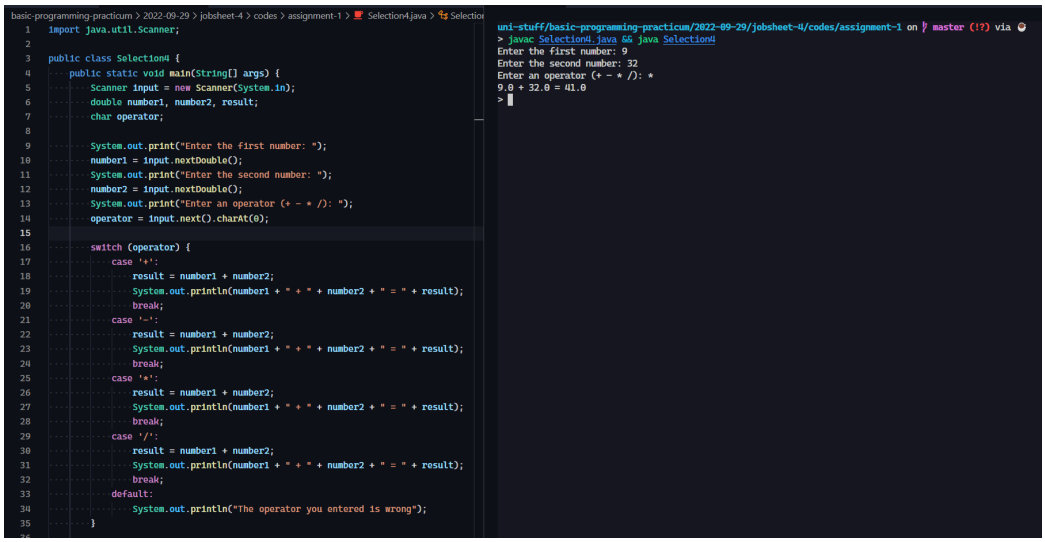


Figure 8: Selection 4

Questions

1. Explain the function of break and default in experiment 4!

   The `break` keyword is used to stop the switch statement. If we don't use it, it will just passthrough the next case. The `default` keyword is used as the default condition. It's like an `else` keyword for a `switch` statement.

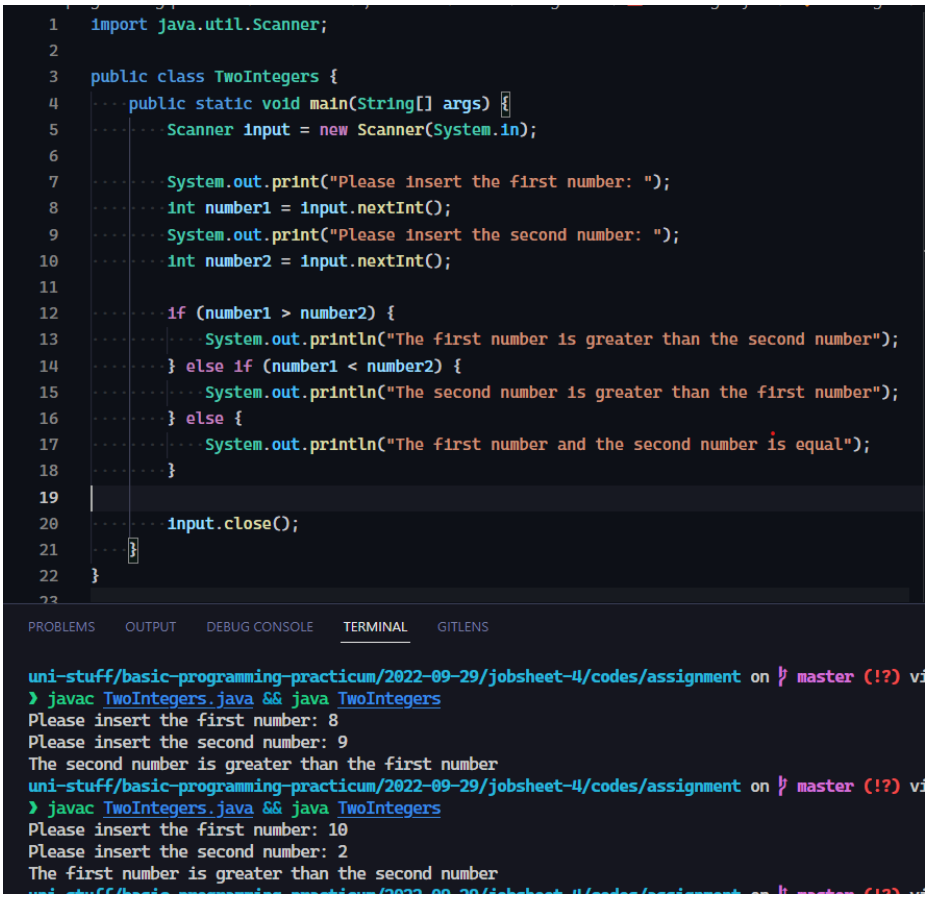
2. Explain the function of the following program code commands!

   ```
   operator = input.next().charAt(0);
   ```

   It is used to get the first character of a String since `input.next()` will take a String until the next newline. In case someone inserted `*asdasd`, we will only get the `*` thanks to the `.charAt(0)` method.

2 Assignment

1. Create a program to input two integers, then print on with the largest value!



```java
import java.util.Scanner;

public class TwoIntegers {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        System.out.print("Please insert the first number: ");
        int number1 = input.nextInt();
        System.out.print("Please insert the second number: ");
        int number2 = input.nextInt();

        if (number1 > number2) {
            System.out.println("The first number is greater than the second number");
        } else if (number1 < number2) {
            System.out.println("The second number is greater than the first number");
        } else {
            System.out.println("The first number and the second number is equal");
        }

        input.close();
    }
}
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  GITLENS

uni-stuff/basic-programming-practicum/2022-09-29/jobsheet-4/codes/assignment on  master (!?) v
❯ javac TwoIntegers.java && java TwoIntegers
Please insert the first number: 8
Please insert the second number: 9
The second number is greater than the first number
uni-stuff/basic-programming-practicum/2022-09-29/jobsheet-4/codes/assignment on  master (!?) v
❯ javac TwoIntegers.java && java TwoIntegers
Please insert the first number: 10
Please insert the second number: 2
The first number is greater than the second number
```

Figure 9: Comparing two integers

2. Observe the following flowchart!

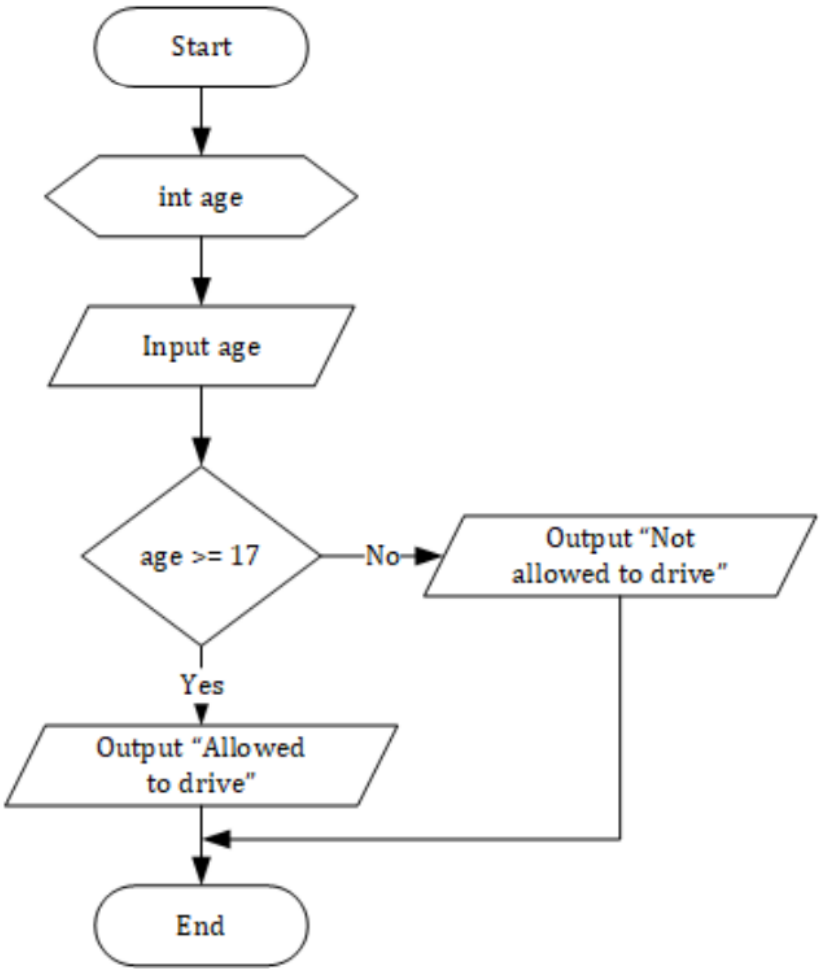


Figure 10: Flowchart

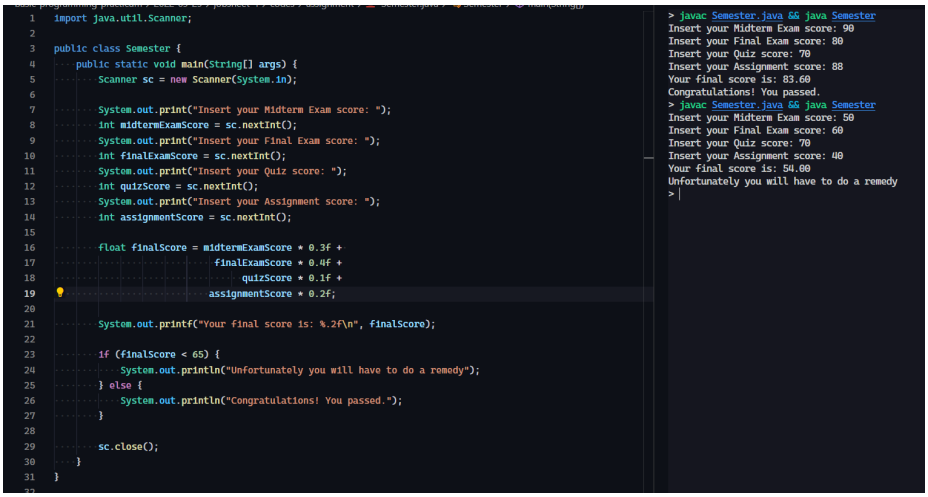
Write program code according to the flowchart!

```java
import java.util.Scanner;

public class FlowchartAge {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int age;

        System.out.print("Please insert your age: ");
        age = input.nextInt();

        if (age >= 17) {
            System.out.println("Allowed to drive");
        } else {
            System.out.println("Not allowed to drive");
        }

        input.close();
    }
}
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  GITLENS
uni-stuff/basic-programming-practicum/2022-09-29/jobsheet-4/codes/assignment on  master (!?) vi
❮ javac FlowchartAge.java && java FlowchartAge
Please insert your age: 16
Not allowed to drive
uni-stuff/basic-programming-practicum/2022-09-29/jobsheet-4/codes/assignment on  master (!?) vi
❯ javac FlowchartAge.java && java FlowchartAge
Please insert your age: 18
Allowed to drive
```

Figure 11: Implementation based on the Flowchart

3. At the end of the semester a lecturer calculates the final score of students which consists of midterm exam score, final exam score, quiz scores, and assignment scores. The final score is obtained from 30% of midterm exams core, 40% of final exam score, 10% of quiz scores, and 20% of assignment scores. If the final score of the student is less than 65, then the student will get a remedy. Create a program to help determine which students get remedies based on the final score they received!



```java
import java.util.Scanner;

public class Semester {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Insert your Midterm Exam score: ");
        int midtermExamScore = sc.nextInt();
        System.out.print("Insert your Final Exam score: ");
        int finalExamScore = sc.nextInt();
        System.out.print("Insert your Quiz score: ");
        int quizScore = sc.nextInt();
        System.out.print("Insert your Assignment score: ");
        int assignmentScore = sc.nextInt();

        float finalScore = midtermExamScore * 0.3f +
                           finalExamScore * 0.4f +
                           quizScore * 0.1f +
                           assignmentScore * 0.2f;

        System.out.printf("Your final score is: %.2f\n", finalScore);

        if (finalScore < 65) {
            System.out.println("Unfortunately you will have to do a remedy");
        } else {
            System.out.println("Congratulations! You passed.");
        }

        sc.close();
    }
}
```

```
> javac Semester.java && java Semester
Insert your Midterm Exam score: 90
Insert your Final Exam score: 80
Insert your Quiz score: 70
Insert your Assignment score: 88
Your final score is: 83.60
Congratulations! You passed.
> javac Semester.java && java Semester
Insert your Midterm Exam score: 50
Insert your Final Exam score: 60
Insert your Quiz score: 70
Insert your Assignment score: 40
Your final score is: 54.00
Unfortunately you will have to do a remedy
>
```

Figure 12: Semester score calculation implementation