FACULTY: INORMATION & COMMUNICATION TECHNOLOGY
SUBJECTS: PRINCIPLES OF PROGRAMMING A & INTRODUCTION TO PROGRAMMING I
PPA115D & TRO115D

WEB/SEMESTER TEST 3

Tshwane University of Technology

We empower people

WEB/SEMESTER TEST 3
Duration: 12 hrs
Date: 27 October 2020
Total: 100
Pages: 23

1st Examiner
V MEMANI

"Code and test one functionality at a time"
- **Vathiswa Booi**

Instructions

- All the questions must be done on **EC** (**1**, **2**, **3, 4**, **5** and **6**).
- To provide answers on **EC**, please logon **EC** and go to **Student → Tests → Web Test 4**.
- Enter the **subject code** as **PPA115D** or **TRO115D, intake** is **20201** and **test number R.**
- Provide your answers under the following sections:
  - ✓ Question **1** under section **1**.
  - ✓ Question **2** under section **2**.
  - ✓ Question **3** under section **3**.
  - ✓ Question **4** under section **4**.
  - ✓ Question **5** under section **5**.
  - ✓ Question **6** under section **6**.

Question 1: Theory **//10//**

Questions

*For each of the following statements (**1.1 – 1.4**), state whether the given statement is **True** or **False**.*

1.1 Java is a high-level, object-oriented programming language. [1]

1.2 Static members of a class can be accessed through the class name using the dot notation. [1]

1.3 Iteration control structures are used to select statements for execution depending on the evaluation outcome of a condition. [1]

1.4 Company **ABC** gives bursaries to ladies above **16** years who want to further their studies at tertiary institutions. The applicants must at least have a level **6** Mathematics achievement at Matric. If Thandi, a **16** year old lady, applies with a level **7** Mathematics, she will definitely get the bursary. [2]

*For each of the following statements (**1.5 – 1.8**), choose a letter (**A, B, C or D**) that corresponds to the correct answer.*

1.5 What is the best approach to problem solving? [1]

A. Write code.
B. Analyze problem.
C. Write code after analyzing the problem.
D. Write code after analyzing and designing a solution to the problem.

1.6 What is the end product of the compilation phase? [1]

A. Source code.
B. Byte code.
C. Executable code.
D. None of the above.

1.7 What is the purpose of code commenting? [1]

A. Make my code look neat.
B. Make my code readable.
C. Make my code executable.
D. Make my code compilable.

1.8 Institution **XYZ** is increasing the salaries of its employees by **10%.** On top of the increase, Heads of Departments (HODs) are getting a once-off appreciation token of **R2000**. How much is **Prof B** going to get as an HOD, if her current salary is **R40 000.00**? [2]

A. R44000.

B. R44200

C. R46000

D. R45000

Question 2: Code reading **//10//**

Problem statement

2.1 Tendani is running a small cinema business in Sosh. He only allows people who are **18** years and above entry into his cinema. Looking at the given code snippet below, which code is missing at **A**? [2]

```
if(    A    ) {
    System.out.println("Admission granted");
} else {
    System.out.out.print("Admission refused.");
}
```

Choose either **A**, **B**, **C** or **D**.

```
A. age = 18
B. age == 18
C. age > 18
D. age >= 18
```

Your choice: _____

2.2 Institution **XYZ** wants to increase the salary of all **Computer Science** lecturers involved in programming 1 by **10%.** The code of the department is **C** or **c** and the code for programming subjects is **1**. Looking at the given code snippet below, which code is missing at **A,** and **B**? [2]

```
char deptCode;
int subjectCode;
Scanner sc = new Scanner(System.in);
double salary;

System.out.print("Please enter your department code: ");
deptCode = sc.next().charAt(0);
System.out.print("Please enter your subject code: ");
subjectCode = sc.nextInt();

if(____A____) {
    System.out.println("You deserve a raise");
    System.out.print("Please enter your current salary: R");
    salary = sc.nextDouble();
    salary = _________B_________;
} else {
    System.out.println("Sorry, no raise for you.");
}
```

Choose either **A, B, C** or **D**.

```
A. The missing code is:
   at A: (deptCode == 'C' || deptCode == 'c') && (subjectCode == 1)
   at B: salary + 0.1 * salary
B. The missing code is:
   at A: (deptCode == 'C' || deptCode == 'c') & (subjectCode == 1)
   at B: salary + 0.1 * salary
C. The missing code is:
   at A: (deptCode == 'C' || deptCode == 'c') && (subjectCode == 1)
   at B: salary + 10% * salary
D. The missing code is:
   at A: (deptCode == 'C' || deptCode == 'c') && (subjectCode == 1)
   at B: salary + 10 * salary
```

Your choice: _____

2.3 Thandeka wants to create an application that will accumulate the sum of numbers as entered by the user. When the user decides to end the program, the sum of the numbers must be displayed. Given below is a partial code in fulfillment of Thandeka's requirements. Which code is missing at **A**? [2]

```
Scanner sc = new Scanner(System.in);
int option, num, sum = 0;

System.out.println("Please enter an option" + "\n" +
    "1 - add numbers" + "\n" +
    "2 - exit");
option = sc.nextInt();

while(_____A_____) {
    if(option == 1) {
        System.out.print("Please enter number: ");
        num = sc.nextInt();

        sum += num;
    } else {
        System.out.println(option + " is invalid. Please enter 1 or 2.");
    }

    System.out.println("Please enter an option" + "\n" +
                       "1 - add numbers" + "\n" +
                       "2 - exit");
    option = sc.nextInt();
}

System.out.println("The sum is " + sum);
```

Choose either **A**, **B**, **C** or **D**.

```
A. option != 2
B. option == 2
C. option = 2
D. option =! 2
```

Your choice: _____

2.4 Munganga wants to determine the sum and average of the first five numbers, **1 – 5**. Looking at the given code snippet below, which code is missing at **A** and **B**? [2]

```
52
53  int sum = 0;
54  double average;
55
56  for(int i = 0; i < 5; i++){
57      sum = sum + ______A_______;
58  }
59
60  average = ______B_______;
61
62  System.out.println("The sum of the first five numbers, 1 to 5 is " + sum);
63  System.out.println("The average of the first five numbers, 1 to 5 is " + average);
64
```

Choose either **A**, **B**, **C** or **D**.

```
A. The missing code is:
   at A: (1 + 2 + 3 + 4 + 5)
   at B: sum / 5
B. The missing code is:
   at A: i
   at B: sum / 5
C. The missing code is:
   at A: (i + 1)
   at B: sum / 5
D. The missing code is:
   at A: 5
   at B: sum / 5
```

Your choice: _____

2.5 Pule wants to create a static method that will display a message on the screen. Which of the following method headers is the most appropriate to use? [2]

Choose either **A**, **B**, **C** or **D**.

```
A. public static void displayMessage(String message);
B. public void displayMessage(String message);
C. public static int displayMessage(String message);
D. public int displayMessage(String message);
```

Your choice: _____

Question 3: IPO chart creation **//10//**

Problem statement

A parallelogram is one of the most basic geometric shapes. It has the geometric property of opposite sides and angles being equal. **Figure 1** below shows a typical parallelogram.

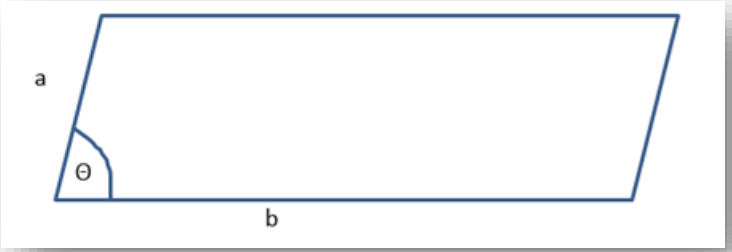


Figure 1: A parallelogram

We can determine both the **area** and **perimeter** of the shape using the following respective formulae:

- area = a * b * sin Θ, where Θ is an angle in degrees.
- perimeter = 2 * (a + b)

In this section, we want you to create an IPO chart that will determine and display both the **area** and **perimeter** of a **parallelogram**. Given to you in the next page is an incomplete IPO chart that must be filled with answers from the provided table.

To do

Create an IPO chart that will determine and display both the **area** and **perimeter** of a **parallelogram**. Given to you below is **Table 1.** The table has possible answers that you can use to complete the given IPO chart. Each possible answer is represented by a letter on its left, e.g **angle** is represented by **R**.

A	Ask user to enter side a	**D**	Display perimeter	**G**	Enter a	**J**	Determine perimeter	**M**	area	**P**	a
B	Ask user to enter side b	**E**	Display area	**H**	Enter b	**K**	Determine area	**N**	perimeter	**Q**	b
C	Ask user to enter angle	**F**	O	**I**	Enter angle	**L**	Determine angle	**O**	I	**R**	angle

Table 1: Possible answers for completing IPO chart.

Use the table to complete the given IPO chart below. Fill-in the chart with letters corresponding to possible answers.

Incomplete IPO chart

1._____	**P**	**15.**_____
2. a 3. _____ 4. angle	5. Ask user to enter side a 6._____ 7. Ask user to enter side b 8. _____ 9. _____ 10. _____ 11. _____ 12. Determine perimeter 13. _____ 14. Display perimeter	16. area 17. _____

Question 4: Flowchart implementation **//15//**

Problem statement

Even numbers have a property of being divisible by **2**. This means when divided by **2**, they produce a remainder of **0**. For example, **8** is an even number because its remainder is **0** when divided by **2**.

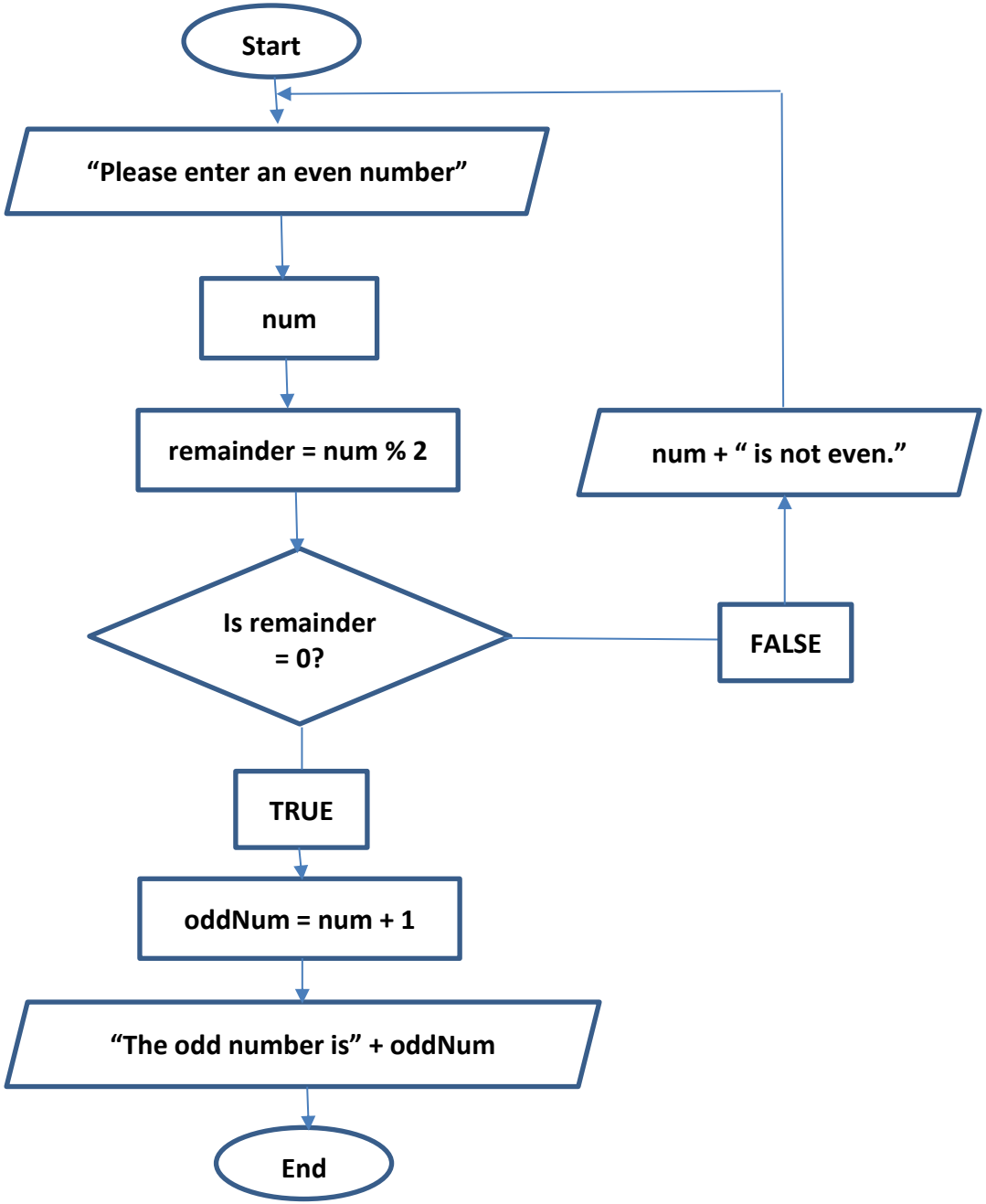
In this section you are given a flowchart that converts an even number into an odd number. The flowchart allows a user to enter a number, verifies that the number is even, and converts the verified even number into an odd number and thereafter display the odd number. Adding a **1** to an even number results in an odd number.

Consequently in this section you are required to convert the flowchart given in the next page into a working Java program.

To do

Create a Java program called **EvenNumbersConverterApp** that will convert a verified even number into an odd number using the provided flowchart in the next page. The odd number must be displayed.

Flowchart


Start
"Please enter an even number"
num
remainder = num % 2
Is remainder = 0?
FALSE
num + " is not even."
TRUE
oddNum = num + 1
"The odd number is" + oddNum
End

Question 5: Problem solving **//15//**

Problem statement

The Fibonacci series has an interesting pattern. The **next** number in the series is calculated from the sum of the **current** and **previous** number. The first two numbers in the series are assumed to be **1** and **2**, and the series gets developed from there. In general, a number in the Fibonacci series is determined by using the following formula:

nextNumber = currentNumber + previousNumber

Let's take an example.

Example

Say we have the following Fibonacci series of **5** numbers:

1, 2, 3, 5, 8

Using the formula provided above, the series is built or developed as follows:

nextNumber = currentNumber + previousNumber

3 = 2 + 1

5 = 3 + 2

8 = 5 + 3

We can see that the pattern of the series is as follows: **3** (**next number**) is the sum of **2** (**current number**) and **1 (previous number)**, the next number **5** is the sum of **3** and **2**, and **8** is the sum of **5** and **3**.

In this section, we want you to create a program that will generate and display a Fibonacci series of **15** numbers.

To do

Create an application called **FibonacciSeriesApp** that will generate and display a Fibonacci series of **15** numbers.

Question 6: Project coding **//40//**

Problem statement

Number guessing is a very interesting game played between computers and humans. In this game a computer generates a random number in the range of **1** to **5**, both numbers inclusive. After generation of the random number, a placeholder in the form of an **x** is displayed to represent the random number. This is done to hide the random number from the user. The user is then asked to guess the number. The guess is compared to the computer generated number. If the two match, the user wins and her/his score is incremented by 1, otherwise the score of the computer is the one that gets incremented with a 1. After the score update, the counter that tracks the number of games played is also incremented by 1. The user is then asked if she/he wants to play the game again. If the user says yes, the game starts again. If the user says no, the game summary is displayed and the game comes to an end. The following game summary report is displayed on the screen:

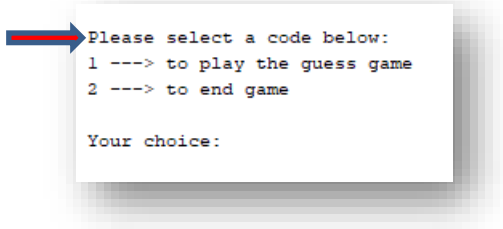
- The number of games played.
- The number of games won by the user.
- The number of games won by the computer.

Consequently, in this section we want you to create an application called **NumberGuessingApp** that will simulate the number guessing game. In the next pages we demonstrate how the application should work.

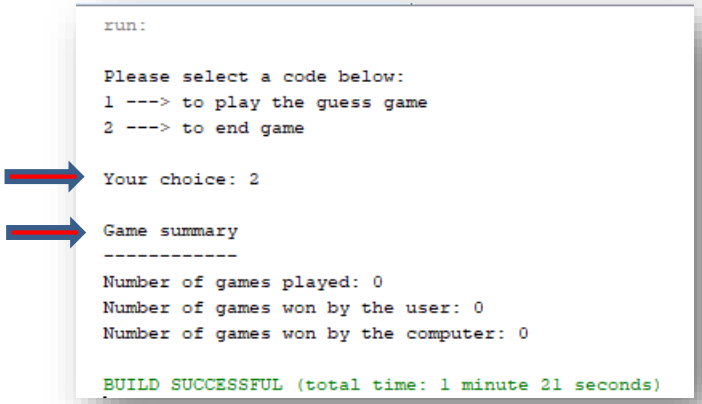
How should the program work?

The program should work as follows:

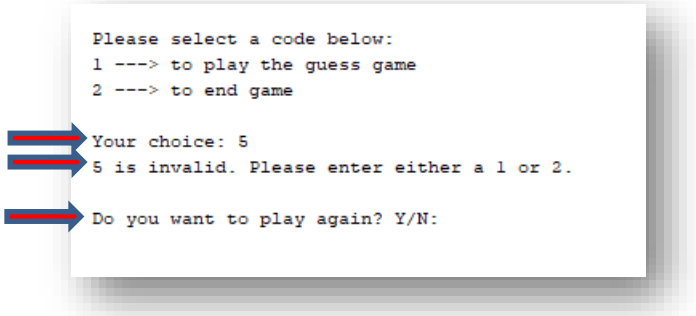
- Display the game options.



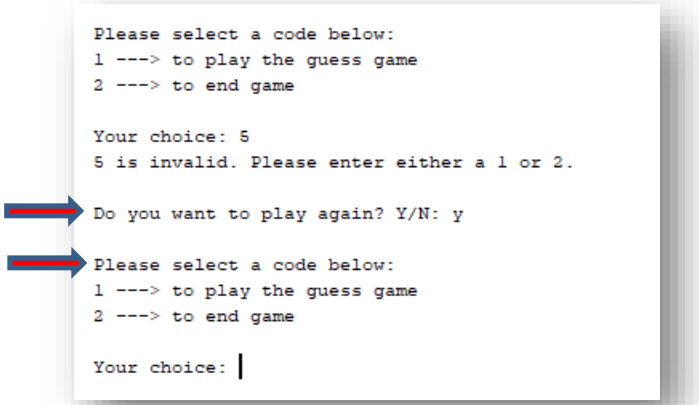
- End the game if the user enters a **2**. The game summary must be displayed.



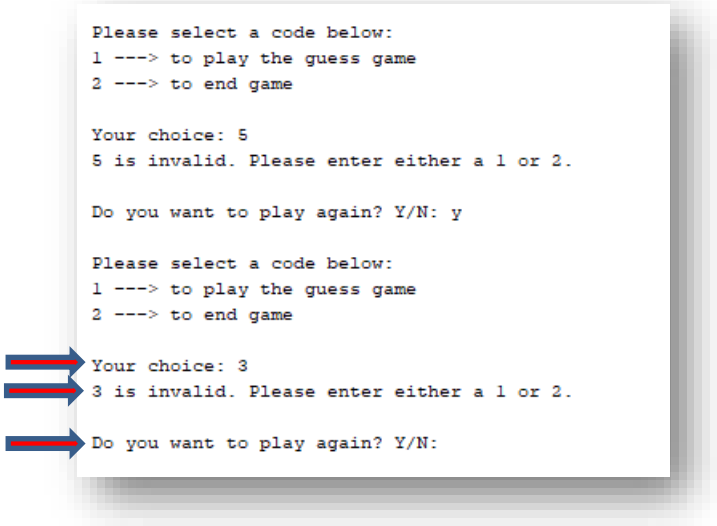
- Display an error message if the user enters an invalid number, and ask if the user wants to play again.



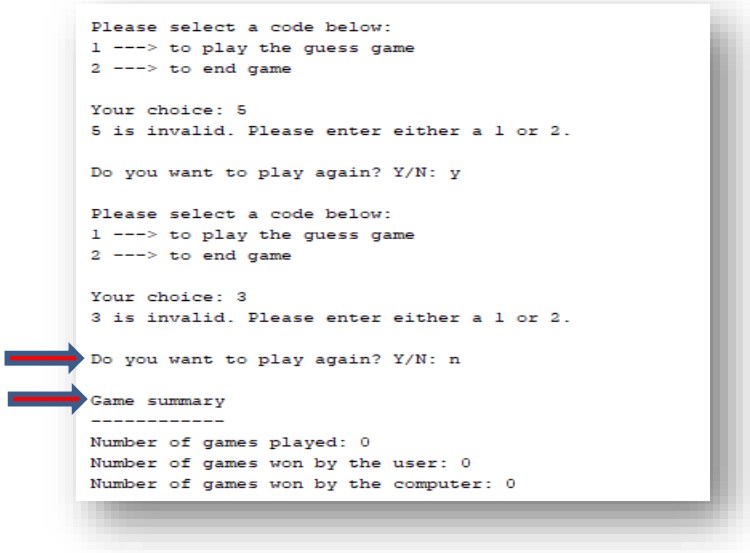
- If the user says **yes**, display the menu options again.



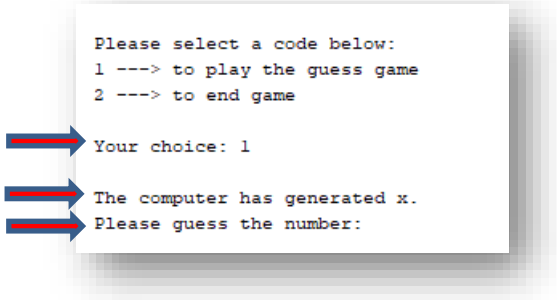
- If the user enters an invalid number, display error message and ask if they want to play again.



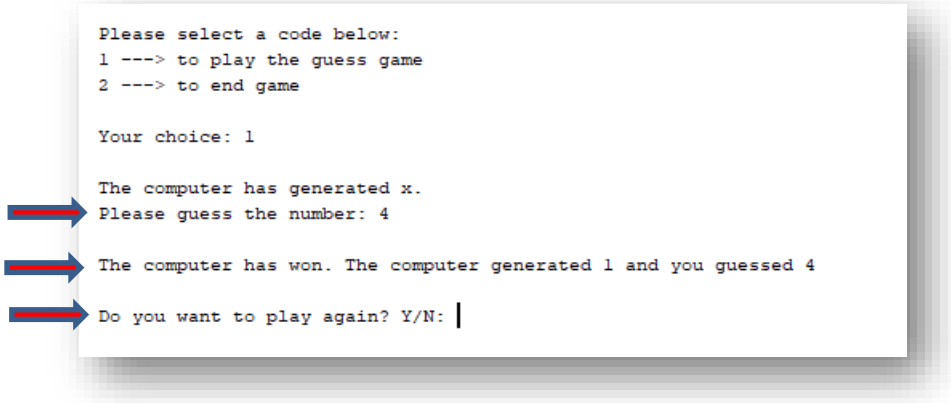
- If the user says **no**, display the game summary and end the game.



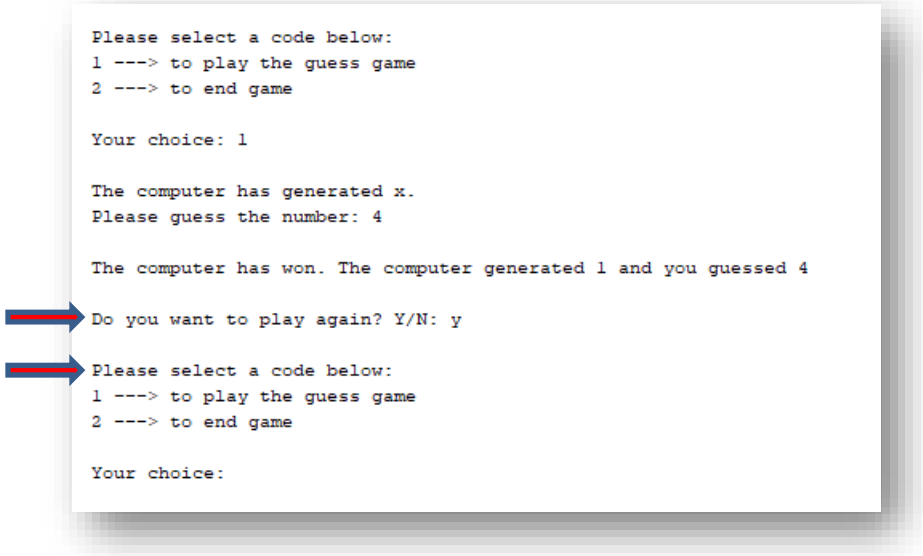
- If the user enters a 1 from the menu, display the placeholder and ask the user to guess the hidden number.



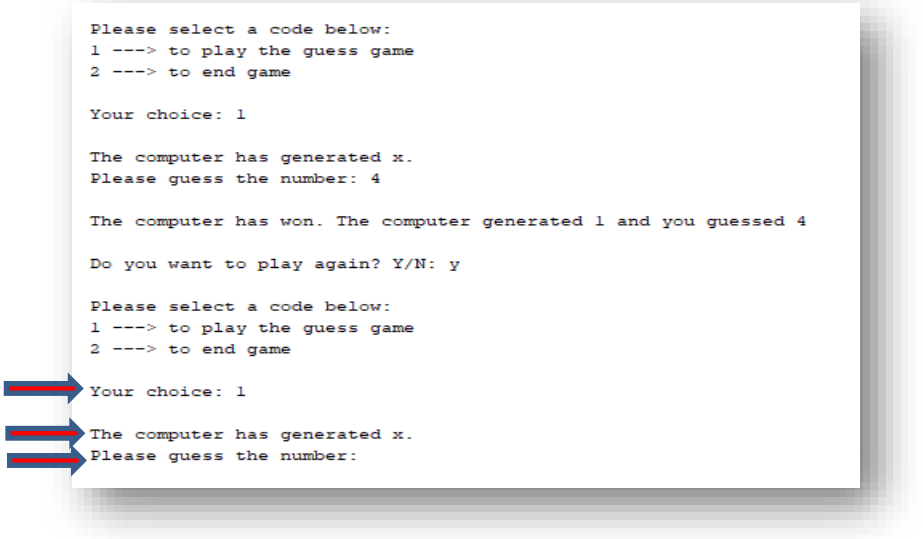
- After the user has entered the guess, display outcome and ask if the user wants to play again.



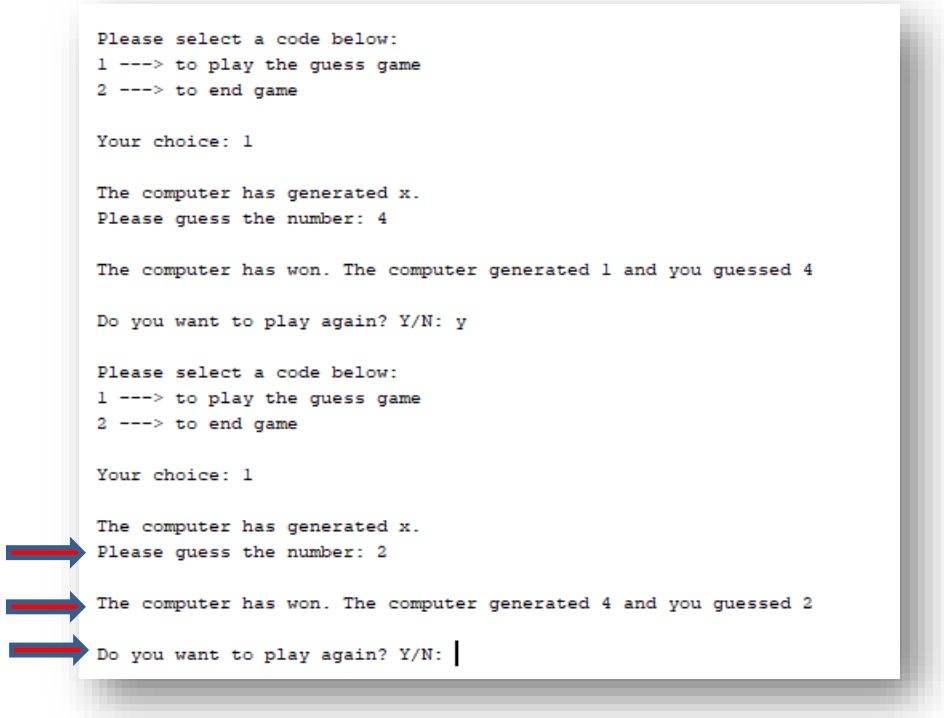
- If the user wants to play again, display the menu options.



- If the user enters **1**, display placeholder and ask user to guess the hidden number.



- After the user has entered the guess, display outcome and ask if the user wants to play again.



- If the user says no, display the game summary and end program.

```
Please select a code below:
1 ---> to play the guess game
2 ---> to end game

Your choice: 1

The computer has generated x.
Please guess the number: 4

The computer has won. The computer generated 1 and you guessed 4

Do you want to play again? Y/N: y

Please select a code below:
1 ---> to play the guess game
2 ---> to end game

Your choice: 1

The computer has generated x.
Please guess the number: 2

The computer has won. The computer generated 4 and you guessed 2

Do you want to play again? Y/N: n

Game summary
------------
Number of games played: 2
Number of games won by the user: 0
Number of games won by the computer: 2
```

To do

Create an application called **NumberGuessingApp** that will simulate the number guessing game played between a computer and human as described in the **"Problem statement"** part and demonstrated in the **"How should the program work?"** section. Your application must consist of static methods that perform specific tasks. Below is the description of each static method.

Description of static methods

1. **public static int generateRandomNumber()** [5]

   This method must generate and return a random number in the range of **1 – 5**.

2. **public static int getUserOption()** [3]

   This method must return an option chosen by a user. A user must be given two options, namely:

   - Enter **1** to play the game.
   - Enter **2** to end the game

3. **public static void displayGameSummary(int numGamesPlayed, int numGamesWonByUser, int numGamesWonByComputer)** [3]

   This method must display the game summary statistics. The game summary involves the number of games played, the number of games won by the user and the number of games won by the computer.

4. **main()** [29]

   The main method must operate as per the description given in the problem statement and shown in the **"How should the program work?"** section. Code written in other static methods must not be repeated in the main method.