This week the basic programming concepts have been introduced. In this portfolio, you are required to develop programs for solving simple problems using loops, control structures, logical conditions and basic calculations.

Exercises

EX1:
Write a program that read a number from the console and check the given number is EVEN or ODD. For example, one would see the bellow message in the terminal when you run your code.

```
Enter a number to check ODD or EVEN:
21
21 is an ODD number
Press any key to continue . . .
```

EX2:
Extend the previous program to check whether the entered number is a prime number. A prime number is a number which is greater than 1 and has no positive divisors other than 1 or itself.

EX3:
Write a program that read an odd number from console and display a X shape of stars in the console. The maximum number of lines of stars from the top to bottom in the X shape is the entered odd number. For example, one would see the bellow message in the terminal when you run your code.

```
Enter the maximum number of lines of *:
5
  *   *
   * *
    *
   * *
  *   *

Press any key to continue . . .
```

EX4:
Write a program that read a sentence from console and display the sentence from lower case to upper case character **without using** any library functions (e.g. String.ToUpper). For example, one would see the bellow message in the terminal when you run your code.

```
Enter a sentence:
Our tutor name is Ardhendu
Upper case string:
OUR TUTOR NAME IS ARDHENDU
Press any key to continue . . .
```

EX5:

Write a program to check whether given input string is a palindrome or not. Your code should be able to read a string from the console and provide the yes/no decision as an output. Palindrome is a word, phrase, or sequence that reads the same backwards as forwards. For example, one would see the bellow message in the terminal when you run your code.

```
Please enter a word:
madam

madam is a palindrome.
Press any key to continue . . .
```

EX6 (Optional):
Create a program that will generate a random number between 0-100 and keep it as a secret number. This can be generated using:

```
Random r = new Random();
int r_num = r.Next(100);//random number between 0-100
```

`r_num` is the secret number and you will be given 5 attempts to guess this number. Each time you guess you will get one of the bellow messages based on the difference between the secret number and the guessed number:

1. Very high or very low (>= 50)
2. High or low (< 50 and >= 20)
3. Moderately high or moderately low (<20 and >= 10)
4. Somewhat high or somewhat low (<10)

For example, let's assume the secret number is 72, if you enter the choice as a 20, then you will get the message of "Very Low". This means you have to increase your guess by at least 50. Let's say the second guess is 70, then the message will be "Somewhat Low". Then you can be sure that the secret number is in between 70-80. Then your third choice would be 75 and the message will be "Somewhat High" then you can conclude that the secret number is in between 70 -75. Then if you enter your 4^{th} choice as 72 then you will win the game in 4^{th} attempt.