VISA NCG

.NET and C# Course

Assignments

Version 1.0 | 24th September, 2020

Revision History

Please keep the latest version on top

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version** | **Change Description** | **Sections** | **Date** | **Author** | **Reviewer** |
| 1.0 | Initial Version | All | 24th September, 2020 | Joydip Mondal | Visa SME |

References

Reader should read this document in conjunction with the following documents

|  |  |
| --- | --- |
| **No** | **Document Name** |
|  | VISA-.NET for NCGs 2020-C# Bootstrap Course\_Content.docx |

Contents

[Revision History 2](#_Toc51873596)

[References 2](#_Toc51873597)

[1. Fibonacci series 4](#_Toc51873598)

[Objective: 4](#_Toc51873599)

[Problem Statement: 4](#_Toc51873600)

[2. Compound Interest 4](#_Toc51873601)

[Objective: 4](#_Toc51873602)

[Problem Statement: 4](#_Toc51873603)

[3. Credit card bill 5](#_Toc51873604)

[Objective: 5](#_Toc51873605)

[Problem Statement: 5](#_Toc51873606)

## Fibonacci series

In mathematics, the Fibonacci numbers are the numbers in the following integer sequence:

0,1,1,2,3,5,8,13,21,34,…

By definition, the first two Fibonacci numbers are 0 and 1, and each subsequent number is the sum of the previous two.

In mathematical terms, the sequence Fn of Fibonacci numbers is defined by the recurrence relation

Fn = Fn-1 + Fn-2; with initial values F0 = 0 and F1 = 1.

### Objective:

Understand how to write a recursive function.

### Problem Statement:

Write a method **Fibonacci** that takes some integer has a parameter and returns the nth Fibonacci number, where we think of the first 1 as the first Fibonacci number. Thus, an invocation of **Fibonacci (6)** should return 8, and in invocation of **Fibonacci (10)** should return 55

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Nth Fibonacci | 0 | 1 | 1 | 2 | 3 | 5 | 8 | 13 | 21 | 34 | 55 |

## Compound Interest

**Compound interest** arises when interest is added to the principal, so that from that moment on, the interest that has been added also it earns interest. This addition of interest to the principal is called compounding. A bank account, for example, may have its interest compounded every year: in this case, an account with $1000 initial principal and 20% interest per year would have a balance of $1200 at the end of the first year, $1440 at the end of the second year, and so on.

### Objective:

Understand which type of loop to use.

### Problem Statement:

You deposit INR. 1000.00 in the bank account at 5% annual compound interest. The interest is paid once at the end of the year. What if you are interested in becoming a millionaire (Rs 1000000.00)? How long will it take to reach one million?

|  |  |  |
| --- | --- | --- |
| Year | Interest for the year | End of the year amount |
| 1 | 1000\*0.05=50 | 1050.00 |
| 2 | 1050\*0.05=52.5 | 1102.50 |
| 3 | 1102.50\*0.05=55.125 | 1157.625 |

## 

## Credit card bill

### Objective:

Understand which type of loop to use.

### Problem Statement:

You owe the credit card company 1000.00 rupee. The company charges you 1.5% per month on the unpaid balance. You have decided to stop using the card and to pay off the debt by making a monthly payment of N rupee a month. Write a program that asks for the monthly payment, the program writes out the balance and total payments so far for every succeeding month until the balance is zero or less.

Sample Input:

Enter the monthly payment: 100

Sample Output:

Month: 1 balance: 915.0 total payments: 100.0

Month: 2 balance: 828.725 total payments: 200.0

Month: 3 balance: 741.155875 total payments: 300.0

## Mileage:

### Objective:

Understand which control flow statements and operators.

### Problem Statement:

Write a program to evaluate the fuel consumption of a car. The mileage at the start and end of the journey should be read, and also the fuel level in the tank at the start and end of the journey.

Calculate fuel used, kilo meters travelled, and hence the overall fuel consumption in kilo meters travelled per liter of fuel.

|  |  |  |
| --- | --- | --- |
| UTC | Sample Input | Sample Output |
| 01 | Fuel Level at start of journey: 30 Liters  Fuel Level at end of journey: 20 Liters  Mileage reading at Start of journey: 24,500  Mileage reading at end of journey: 24,650 | Total fuel used : 10 Liters  Distance covered: 150 Km  Mileage / Liter: 15 Km |
| 02 | Fuel Level at start of journey: 12 Liters  Fuel Level at end of journey: 4 Liters  Mileage reading at Start of journey: 12,000  Mileage reading at end of journey: 12,100 | Total fuel used : 8 Liters  Distance covered: 100 Km  Mileage / Liter: 12.5 Km |
| 03 | Fuel Level at start of journey: 30 Liters  Fuel Level at end of journey: 29 Liters  Mileage reading at Start of journey: 42,000  Mileage reading at end of journey: 42,009 | Trip too short for ac-curate results |

The message “**Trip too short for accurate results**” should be printed if the mileage used to calculate the fuel consumption is less than 10 kilo meters

## Student Results:

### Objective:

Understand which control flow statements and operators.

### Problem Statement:

Percentage marks attained by a student in three exams are to be entered to a computer. An indication of Pass or Fail is given out after the three marks are entered.

The criteria for passing are as follows:

A student passes if all three examinations are passed.

Additionally a student may pass if only one subject is failed and the overall average is greater than or equal to 50.

The pass mark for an individual subject is 40

|  |  |  |
| --- | --- | --- |
| UTC | Sample Input | Sample Output |
| 01 | Marks in First Subject: 89  Marks in Second Subject: 55  Marks in third subject: 99 | PASS |
| 02 | Marks in First Subject: 32  Marks in Second Subject: 45  Marks in third subject: 90 | PASS |
| 03 | Marks in First Subject: 22  Marks in Second Subject: 65  Marks in third subject: 90 | PASS |
| 04 | Marks in First Subject: 22  Marks in Second Subject: 25  Marks in third subject: 20 | FAIL |

## Matrix Arrangement:

### Objective:

Understand how to use arrays.

### Problem Statement:

Write a function to return an array where the midst position contains the smallest value followed by the next smallest value on the right of the midst position, followed by the next smallest value to the left of the midst position and the rest of numbers continue in this format

Function signature:

public static int[ ] ArrangeElements(int[,] inputArray)

{

// write the code here

}

|  |  |  |
| --- | --- | --- |
| UTC | Sample Input | Sample Output |
| 01 | |  |  |  | | --- | --- | --- | | 4 | 2 | 13 | | 3 | 8 | 5 | | 9 | 6 | 17 | | {17,9,6,4,2,3,5,8,13} |
| 02 | |  |  |  | | --- | --- | --- | | 4 | 1 | 3 | | 3 | 8 | 5 | | 4 | 16 | 17 | | {17,8,4,3,1,3,4,5,16} |