|  |  |  |
| --- | --- | --- |
| Module Leader: Mo Rezai | | Level: 7 |
| Module Name: Computer Programming | | Module Code: 55-700724 |
| Assignment Title: Binary class | | |
| Individual / Group: Group | Weighting: 50% | Magnitude: *wordcount/length of…*  *2000 Words* |
| Submission date/time: See MyHallam Site | Blackboard submission Y/N: Yes Turnitin submission Y/N: No | Format: Vistual Studio 2022 solution folder |
| Planned feedback date: 3 Weeks fom submission point | Mode of feedback: BB MyGrades | In-module retrieval available: No |
| **Module Learning Outcomes**   * Articulate key concepts that relate to designing and implementing small applications. * Apply programming structures and language syntax such as sequence, iteration, selection and functions to design software programs. * Produce a software application using appropriate programming tools and environments for a specified case study. | | |

# Assessment Brief

* This is group work. This is individually assessed.
* You must organise as a group of 2 (3 is too many).
* One representative of the group must email the module leader ([m.j.rezai@shu.ac.uk](mailto:m.j.rezai@shu.ac.uk) ).
  + Email must be copied to the 2nd member of the group.
  + Email must contact full names and student numbers.
  + Deadline for the email is 11/11/2022. Following this date students without group will be grouped randomly.

Introduction

This assignment provides you with the opportunity to consolidate your skills, design, and implement the above concepts in a working program.

*[int](https://docs.microsoft.com/en-us/dotnet/api/system.collections.generic.list-1?view=net-5.0)* class is a much-used and well-documented .NET class. The operations that are performed on objects of this class are often arithmetic. The object of this exercise is to implement a new class, the Binary class. This class will do operations on binary representations of integers.

In the course of the module, you will learn to ..

* Build C# console application using .NET Framework.
* Implement your own *Binary* class:
  + You will implement not only to mimic the behaviour of *int* class, but also to extend operations to include some that are specific to Binary representation.
  + You will extend the behaviour of Binary to extend operations to include some that are specific to Binary representation of integers.
  + You must work with 16 bits binary representations of integers. You must implement the following operations:
    - Binary operators: Ones’ complement (~), twos’ complement or negator (-), Shift to the left (<<), Shift to the right (>>)
    - Binary arithmetic operators and Logical operators: +, -, \*, /, ==, !=, <, >, <=, >=
  + All algorithms must perform in binary. Algorithms that convert to denary, perform in denary, and convert back to binary carry 0 marks.
* Construct code: use types, create variables, design methods and code constructs, and design class.

Tasks and Submission:

* + You are provided with a Console solution (*Binary.zip*). Download the solution and implement ‘*Binary.cs*’ class to pass tests in ‘*Program.cs*’.
  + ‘*Binary.cs*’: This class requires your implementation. Regions have been defined in the class to help you with the requirements. In implementing this class, you are encouraged to consider good practice for code design. Good consideration needs to be given to method design, sensible use of types and variables, and good code construct. Good and sensible OO practice is encouraged.
  + ‘*Program.cs*’: This file contains tests that your implementation must pass with expected outcomes. You are not to make changes to ‘*Program.cs*’. Note that ‘*Program.cs*’ is not intended to be rigorously testing all possible sensible practices. These tests must be used as helpful guidance and you are expected to apply results of your own investigations and enquiries, over and above these tests.
  + ‘*ExpectedOutcome.txt*’: This is the expected output of application of tests and execution of your implementation. Use this as reference.
  + Submit, only, the class file *‘Binary.cs’* to the submission point on BB. One member submits on behalf of the group.

# Assessment Criteria

Note that this assignment is Group work / Individually assessed

|  |  |  |
| --- | --- | --- |
| **Binary Class Implementation** | | **Mark** |
| Fields | | /4 |
| Properties | | /2 |
| index operator : Binary[index] | | /3 |
| Implicit convertors | Binary to int | /5 |
| int to Binary | /5 |
| Methods | ToDecimal | /4 |
| ToString | /4 |
| Shift Operators | << | /4 |
| >> | /4 |
| Binary Operators | Ones’ complement | /3 |
| Twos’ complement (Negation) | /2 |
| Binary Arithmetic Operators | + | /5 |
| - | /5 |
| \* | /5 |
| / | /10 |
| % | /5 |
| Logical Operators | == | /5 |
| != | /5 |
| < | /5 |
| > | /5 |
| <= | /5 |
| >= | /5 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Presentation/Demonstration Mark Scheme 50% | | | | |
| **Class** | **Category** | **Grade** | **%** | **General Characteristics** |
| Distinction  (exceptional) | Exceptional Distinction | 16 | 82 | Exceptional breadth and depth of knowledge and understanding evidenced by own independent insight and critical awareness of  concepts at the forefront of the discipline; evidence of extensive and appropriate independent inquiry operating with advanced concepts, methods, and techniques to solve problems in unfamiliar contexts; |
| Distinction (Excellent) | High Distinction | 15 | 78 | Excellent knowledge and understanding evidenced by some clear independent insight and critical awareness of relevant concepts some of which are at the forefront of the discipline; evidence of appropriate independent inquiry operating with core concepts, methods, and techniques to solve complex problems in mostly familiar contexts; |
| Mid Distinction | 14 | 75 |
| Low Distinction | 13 | 72 |
| Merit (Good) | High Merit | 12 | 68 | Very good knowledge and understanding is evidenced as the student is typically able to independently relate taught facts/concepts together some of which are at the forefront of the discipline; evidence of some competent independent inquiry operating with core concepts, methods, and techniques to solve familiar problems; |
| Mid Merit | 11 | 65 |
| Low Merit | 10 | 62 |
| Pass (Sufficient) | High Pass | 9 | 58 | Satisfactory knowledge and understanding of the area of study balanced towards the descriptive rather than critical or analytical and mostly confined to concepts that are not at the forefront of the discipline; evidence of some independent reading and research to advance work and inform arguments and approaches; Arguments and explanations are limited in range and depth although some are adequately supported by the literature albeit descriptively rather than critically; |
| Mid Pass | 8 | 55 |
| Low Pass | 7 | 52 |
| FAIL (Insufficient) | Borderline Fail | 6 | 48 | Knowledge and understanding is insufficient as the student only evidences an understanding of small subset of the taught concepts and techniques; fails to make sufficient links between known concepts and facts to adequately solve relevant aspects of the brief/problem; |
| Mid Fail | 4 | 45 |
| Low Fail | 2 | 42 |
| FAIL (Incompetent) | Very Low Fail | 1 | 38 | Knowledge and understanding is highly insufficient as the student is unable to evidence any meaningful understanding of taught concepts or methods; very limited evidence of reading and research to advance work; inadequate technical and practical skills as the student is unable to use and apply such skills to address problems or make judgements; limited or lack of understanding of the boundaries of the discipline and does not question received wisdom; approach to learning lacks autonomy and approach to tasks is not sustained; inability to communicate coherently. |
| ZERO | Zero | 0 | 0 | Work not submitted, work of no merit, penalty in some misconduct cases. |