**RICHFIELD**

**FACULTY OF INFORMATION TECHNOLOGY**

**PROGRAMMING 622 – C++**

**4TH SEMESTER ASSIGNMENT**

**Name & Surname: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ICAS No: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Qualification: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Semester: \_\_\_\_\_ Module Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Date Submitted: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

|  |  |  |  |
| --- | --- | --- | --- |
| **ASSESSMENT CRITERIA** | **MARK**  **ALLOCATION** | **EXAMINER MARKS** | **MODERATOR MARKS** |
| **MARKS FOR CONTENT** | | |  |
| **QUESTION ONE** | **50** |  |  |
| **QUESTION TWO** | **20** |  |  |
| **QUESTION THREE** | **30** |  |  |
|  |  |  |  |
| **TOTAL MARKS** | **100** |  |  |
| **Examiner’s Comments:** | | |  |
|  | | |  |
|  | | |  |
| **Moderator’s Comments:** | | |  |
|  | | |  |
|  | | |  |
| **Signature of Examiner: Signature of Moderator:** | | |  |

**QUESTIONS ONE [50 MARKS]**

Write a C++ program that implements a linked list as an abstract data type. The program must:

· Be type flexible, that is, could be a list of integers, string, or any type.

· allow the user to add members into the list

· delete members from the list

· make sure the list is still sorted after all the deletion and addition

. Show all the function implementation.

**QUESTIONS TWO [20 MARKS]**

Write a program to convert a number from a binary representation to a decimal format—that is, from base 2 to base 10. Your program should make use of recursion to do the conversion.

**QUESTIONS THREE [30 MARKS]**

A number of the form a + ib, in which i2 = -1 and a and b are real numbers, is called a complex number. We call the real part and b the imaginary part of a + ib. Complex numbers can also be represented as ordered pairs (a, b). The addition and multiplication of complex numbers are defined by the following rules:

(a + ib) + (c + id) = (a + c) + i(b + d )

(a + ib) \* (c + id) = (ac - bd) + i(ad + bc)

Using the ordered pair notation, these rules are written as:

(a, b) + (c, d) = ((a + c), (b + d ))

(a, b) \* (c, d) = ((ac - bd ), (ad + bc))

C++ has no built-in data type that allows us to manipulate complex numbers. Construct a data type, complex Type, that can be used to process complex numbers. Overload the stream insertion and stream extraction operators for easy input and output. We will also overload the operators + and \* to perform addition and multiplication of complex numbers. If x and y are complex numbers, we can evaluate expressions such as x + y and x \* y.

**TOTAL: 100 MARKS**