|  |  |  |  |
| --- | --- | --- | --- |
| **Name:** | **Lee Yan Cheng** | **Lab Group#:** | **B10** |
| **Student Id:** | |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **A** | **0** | **1** | **9** | **9** | **1** | **4** | **1** | **B** | |  | |

**Page 1**

1. Give the modified C statement in report. **[1 pt]**

|  |
| --- |
| display(ageArray[0]); |

1. Find out the purpose of the C operator **sizeof()**? Briefly describe. **[2 pt]**

|  |
| --- |
| 1. It can be used to find the amount of memory allocated to a data type or variable e.g. 4 for int |

**Page 2**

2. Give the function body. **[3 pts]**

|  |
| --- |
| **void printFractionArray( struct Fraction fArray[], int size)**  **{**      for (int i = 0; i < size; i++) {          printFraction(&(fArray[i]));          printf(", ");      }      printf("\n");  **}** |

3. Can you print out the number of fractions**? [3 pts]**

|  |
| --- |
| In main(): **Yes** using the same method in part a Reason (only for ***no***):  In printFractionArray(): **No**  Reason (only for ***no***):  When an array is passed to a function as a parameter, it decays into a pointer to the first element. So using sizeof(arr) in the method would only return the size of the pointer instead. |