**60-266 Lab 5**

NAME: STUDENT NUMBER:

OBJECTIVES:

i) To learn how to to write a simple program using loops.

ii) To learn to manipulate arrays using indexed addressing.

iii) To learn to use different data-related operators

iv) To learn how to to link to an external library.

v) To use some of the procedures provided in irvine32.lib.

1. **Problem**: Create and initialize an array of at least 8 unsigned doubleword integers (myArray). Find the number of elements in the array and the size of the array in Bytes. Find the location (offset) of the first array element and store the offset in edx register. Calculate the sum of alternate elements of the array, using indexed addressing. Store the result in an unsigned word integer variable (mySum). You can assume the value of mySum can fit in a 16-bit integer

**Hints**:

* Use LENGTHOF operator to obtain the length of the array.
* Use OFFSET operator to obtain the address (offset) of the array.
* Update the value of ecx (loop counter) appropriately.
* Use edi as array index.
* Inside loop, add alternate elements.
* Use register window of the debugger to see how register contents are modified after each instruction.

1. **Problem**: Randomly generate a positive integer *n* (2< *n* < 5). Read *n* positive decimal integers from the standard input, store the *n* integers in an array and display each integer on a separate line in hexadecimal format. Use *DumpMem* function to display the contents of the corresponding memory locations.

**Hints**:

* Use *Randomize* and *RandomRange* to generate a random integer *n*.
* In a loop, use *ReadInt* to read each integer.
* Use *WriteHex* to write each integer on the screen.
* Use *DumpMem* to display memory contents.