# CS 115 - Assignment 2
# Dr Malek Mouhoub

Write a program that does the following operations:

1. read a list of naturals into an array,

2. sort the naturals using the *bubble sort* algorithm described in project 4 page 483,

3. sort the naturals using the *insertion sort* algorithm described in project 13 page 487,

4. and return the position of a natural in the array, given its value. You should use here the *binary search* procedure defined in project 12 page 486.

A sample run follows.

```
Enter the number of naturals: 10
Enter the naturals to sort:
12 34 4 0 22 3 6 18 1 11
Naturals sorted in increasing order (using bubble sort):
0 1 3 4 6 11 12 18 22 34
Naturals sorted in increasing order (using insertion sort):
0 1 3 4 6 11 12 18 22 34
Enter the natural to search: 3
Number 3 is found at position 2
Do you want to continue [Y/N]: Y
Enter the natural to search: 5
Number 5 does not exist
Do you want to continue [Y/N]: N
Good bye !
```

# Hand In

1. Name the file containing your program "`assign2.cpp`". Your C++ program **SHOULD** compile using `CC` (Sun compiler) under Hercules.

2. Submit your `assign2.cpp` file through WebCT (webct.uregina.ca). You will then receive an acknowledgement email confirming your submission. You should save this email as a proof of submission. If you do not receive an email acknowledging your submission then you should promptly email the marker (mark115@cs.uregina.ca).

# Marking scheme : total = 100% + 10% (Bonus)

1. Readability (program style) : 10%

   - Program easy to read, well commented,
   - and good structured (layout, indentation, whitespace, . . .) and designed(following the top-down approach)

2. Compiling and execution process : 10%

   - program compiles w/o errors and warnings
   - robustness : execution w/o run time errors

3. Correctness : 80%

   - code produces correct results (output).
   - **output meets the initial requirements (see above for the output format)**.

4. Bonus : 10%

   - Features that increase functionality and/or presentation.