CIS 291 – Data Structures in C++ - Spring 2005
Homework #10
HW #10 due: THURSDAY, April 14th, BEGINNING of lecture

Purpose: implementing heapsort

How this will be turned in:
Use **~st10/291submit**, called from the directory on cs-server where the files you wish to submit are stored.

Homework #10

At the course **Moodle** site, you will find the files:

complete_tree.h
complete_tree.template
heap.h
heap.template

complete_tree is a template class that uses an array to implement a complete binary tree;
heap is a template class that uses a **complete_tree** instance within its implementation of a heap.

Write a function **heapsort** (in file **heapsort.cpp**) that expects an array of **int** and its size as its parameters; it modifies the passed array as a result, using the **heapsort** algorithm described in lecture to modify the passed array so that its contents are in ascending sorted order. (This is a function, so the full function opening comment block --- complete with a suitable collection of examples! --- is expected, of course.)

(remember the basic pseudocode for heapsort of an array's elements ---
add all of the elements of the array into a heap,
while the heap is not empty, remove the heap's root, reheap, and add it to the array in the "next" spot)

Of course, you'll have to **adapt** the basic pseudocode based on the operations of this particular **heap** class.
And, your function **must** use the provided **heap.h** and **heap.template**.

Write a **test_heapsort.cpp** that tests your function (including testing all of your examples, following the expected course testing standards...), and when you are happy with your code run:

```
test_heapsort > 291hw10_out
```

...and submit your resulting **heapsort.template**, **test_heapsort.cpp**, and **291hw10_out**.