Assignment Objective: Build skills on C class creation and integration while implementing a **sorting** utility utilizing the **Heapsort** algorithm.

Requirements:

* Make a copy of your Program 7 (Quad Sorts) code and add the following methods to it. The class will still be named intList. Be sure to update the namespace from pq:: to intList:: on the P6 functions that are copied into this. Adjust the variable names from the p6 components as required to work with p8; for example, qCount becomes heapSize.
  + Public method:
    - void heapSort(); // see lecture notes
  + Private methods:
    - void buildHeap(); // see lecture notes
    - From Program 6:
      * void heapify(int index);
      * void swap(int &x, int &y); // note the change; use this
      * int left(int index) const;
      * int right(int index) const;
  + Private variable:
    - int heapSize; // to track the size of the superimposed heap
* You must not use any other data structure, whether built-in or otherwise.
* Demonstrate that the integer list and sorting work, by using the supplied p8m.cpp.
  + Run your program as follows:

g++ -O2 p8.cpp p8m.cpp -o p8

./p8 10 > p8output.txt

./p8 30 >> p8output.txt

./p8 40000 >> p8output.txt

* + Compare your p8output.txt to the posted p8correctOutput.txt file
  + Into D2L, put a zip file containing:
    - A p8.h file for your **intList** ADT
    - A p8.cpp file for your **intList** ADT implementation
    - A p8output.txt text file with your output
    - DO NOT put a project into D2L
  + Turned into class, a hardcopy of your p8.h, p8.cpp, and p8output.txt files in that order.