MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND

159.201 Algorithms & Data Structures
S1 2022

Assignment 7

Write a C++ program that implements the **Heap sort** using a Heap class with a vector inside. The vector can be of type int. Use the STL vector (`#include <vector>`).

The objective is to show how the Heap sort copes with different initial order from the input set. Therefore, we should experiment using files with ***random***, ***reversed*** or ***sorted*** order. We should count the number of comparisons when *inserting elements* in the Heap, and then when *deleting elements* from the Heap. The only comparisons that should be considered are the ones that compare data directly. Comparisons for other operations (e.g., finding parents or children's index) should *not* be counted.

There are 5 sets of files to test your code, with 3 files per set. The program should accept three arguments in the form of the names of three files. The sample code already does that, so it is a good starting point. The difference between these sets is the total number of elements to sort. The input in the sample program is all based on these text files, three files at a time (random, reversed, sorted). The sample program accepts 3 arguments from command line, and your code should use the same input format. All numbers in the text files are *positive integers*.

The output of the assignment should be the ***state of the Heap before sorting***, the minimum ***number of comparisons*** used to sort that particular file, separated into the ***insertion*** and the ***deletion*** steps, and the state of the ***vector after sorting***. Each test set contains three text files. See an example of input and output for a test set using 5 elements data:
Input: *(all based on text files, download the files from Stream)*
random.txt: 3, 4, 2, 5, 1
reversed.txt: 5, 4, 3, 2, 1
sorted.txt: 1, 2, 3, 4, 5

Output: *(each element is separated by spaces, the name of the files are the arguments argv[])*

```
Heap before sorting: random_N_5.txt
5 4 2 3 1
InsertHeap: 5 comparisons
DeleteRoot: 6 comparisons
Vector after sorting:
1 2 3 4 5

Heap before sorting: reversed_N_5.txt
5 4 3 2 1
InsertHeap: 4 comparisons
DeleteRoot: 6 comparisons
Vector after sorting:
1 2 3 4 5

Heap before sorting: sorted_N_5.txt
5 4 2 1 3
InsertHeap: 6 comparisons
DeleteRoot: 6 comparisons
Vector after sorting:
1 2 3 4 5
```

Use our virtual machine to mark your submissions (host name **vm000296**). The input/output requirements are essential, please follow them carefully to avoid losing marks. Spaces matter and text is case sensitive.
After you are satisfied with the performance of your code as tested in the virtual machine, submit a **one source file** code on Stream by **Friday 3 of June 2022.** Your **name** and **ID number** must appear **on top of the file as comments.** If you are working in a group, then ***all*** names and IDs must appear on top of the file as comments, but you still need to **submit individually** in both the virtual machine and Stream.