Exercise 2 – Implementing a Heap

Due Friday 21st August 2020 by 23:59. (2 marks)

Marking for this exercise is based mainly on correct implementation and code readability. You should implement your code in one file (e.g. ex2.cpp, ex2.c, ex2.java, ex2.py). Make sure your program has a header comment block containing the name of the exercise, your name and your student login (e.g. jfk01). You may implement your solution in C, C++, java or Python.

For this exercise you are to implement a heap, using an array as shown in the lectures. The heap will be built by repeatedly reading the values in the file into sequential locations in the array and then converting the array into a heap using makeheap() as seen in lectures.

Once you have built the heap you are to print out the first five elements of the heap array. Your heap should be able to hold 100 integers. A pseudo-code outline for the program is given below:

```
Begin main
       display a prompt for the file name
       read in the file name
       try to open the file
       if the file fails to open
              print an error message on the screen and exit
       fi
       while we can read an int from the file
              insert the int into the array
       elihw
       close the file
       makeheap()
       for i = 1 to 5
              print the ith element of the heap
       rof
End main
```

Do not implement the heap using a class or with STL. The heap must be implemented using a fixed size array of integers (100 entries should be enough). The heap array and the index to the last item in the heap should be global variables.

Information on makeheap and the functions it uses are available in the week 2 lecture notes.

Submit your program source as ex2.*ext* via moodle where *ext* is on of c, cpp, java or py.