Lab7 Assignment – Binary Heaps

Do these problems in order. While implementing/debugging, hard-code the program's input in your Python file. Once your code is working you can prompt the user for input.

Problem 1: Max-Heap

Implement the Priority Queue ADT operations using **Binary Heaps**. You will have implement the following methods:

`insert, heapify, maximum, extract-max, build-heap`.

You can define a class BinaryHeap that implements these operations. This class will need to have a field *elements* which is the array (a list in Python) of the elements in the heap. You can implement the **build-heap** operation in the constructor itself, by passing a list of unordered elements and building a heap out of them.

Problem 2: Heapsort

Implement the in-place Heapsort using the code written in problem 1. Make sure that the sorting does not use additional space other than the input (unordered) list and maybe a few additional variables.

Problem 3:

Write a function **isBinaryHeap** that takes an arbitrary BinaryTree (not an array/list) as an input and checks if that tree is a Binary Heap or not.

Problem 4: Min-Heap

Build a Binary Min Heap similar to the MaxHeap of Problem 1. You will have to implement the following operations:

`insert, heapify, minimum, extract-min, build-heap`.