

CS1102S Data Structures and Algorithms

Assignment 03: Heaps

For this assignment, use the following submission procedure:

- Solve your assignment using the Java project given in the module home-page.
- Copy the file

`BinaryHeap.java`

to your account on `sunfire` to a folder (let's say `cs1102s/assignment3`).

- Go to that folder and execute the `submit` program:

```
% cd cs1102s/assignment3
% ls *.java
BinaryHeap.java
% /home/course/cs1102s/bin/submit
```

You will get a message that tells you whether the file has been submitted successfully. The script `submit` can be run several times, and you can remove submitted files later. For details on how to use the `submit` script, type

```
/home/course/cs1102s/bin/submit -h
```

Please note all file names are case sensitive and have to conform to the assignment questions.

Submission is activated 5 days before the submission deadline, which is on 5/3, 5:00pm and will be de-activated at that time.

1. (6 marks) Download the assignment project from <http://www.comp.nus.edu.sg/~cs1102s/java/Assignment3.zip>.

Extract the zip file and create a Java project in Eclipse, as usual.

The class `BinaryHeap` does not properly implement the following additional functions:

- `void replaceBy(int p, AnyType newValue)`: Replace the current value at array position `p` by `newValue`.

- void delete(int p): Delete the current value at array position p.

Both functions should throw an java.lang.IllegalArgumentException if the given index is not within the current range of legal positions in the heap.

2. (8 marks)

```
public static <AnyType extends Comparable<? super AnyType>>  
boolean isHeapImplementation( AnyType[] heapArray, int heapSize )
```

to BinaryHeap that returns true if and only if the given array represents a binary heap of size heapSize according to the description of array-based heap implementations in the textbook.

3. (no submission) Convince yourself of the following facts:

- The largest element of a heap must be contained in a leaf.
- There is in general no order among the leaves of a heap.

4. Binary heaps are optimized data structures for removing the smallest element. Implement a non-static method

```
public AnyType deleteMax( )
```

that deletes the **largest** element from a given heap.