Computer Science III	Lab 14D – PQ MONSTER LAB	Lab Value - 100

Lab Goal : This lab was designed to teach you more about using a Priority Queue.

Lab Description : Create a Priority Queue of Monster and store x Monster. The Java PriorityQueue is a minimum heap where the smallest value is always the first one removed.

Sample Data and Output :

```
How many monsters did you wish to enter? :: 8

Enter the ht :: 1
Enter the wt :: 1
Enter the age :: 1


Enter the ht :: 2
Enter the wt :: 2
Enter the age :: 2


Enter the ht :: 1
Enter the wt :: 2
Enter the age :: 3


Enter the ht :: 3
Enter the wt :: 2
Enter the age :: 1


Enter the ht :: 2
Enter the wt :: 1
Enter the age :: 3


Enter the ht :: 2
Enter the wt :: 3
Enter the age :: 1


Enter the ht :: 3
Enter the wt :: 1
Enter the age :: 2


Enter the ht :: 3
Enter the wt :: 3
Enter the age :: 3

[1 1 1, 2 1 3, 1 2 3, 3 2 1, 2 2 2, 2 3 1, 3 1 2, 3 3 3]

getMin() - 1 1 1
removeMin() - 1 1 1

[1 2 3, 2 1 3, 2 3 1, 3 2 1, 2 2 2, 3 3 3, 3 1 2]

getMin() - 1 2 3
```

Files Needed ::

MonsterPQ.java
Lab14d.java

The Lab14 folder has Monster.class

```
class Monster implements Comparable<Monster>
{
   //code for instance variable not shown
   //code for methods not shown

   public Monster()

   public Monster(int ht)

   public Monster(int ht, int wt)

   public Monster(int ht, int wt, int age)

   public void setWeight(int wt)

   public void setHeight( int ht)

   public void setAge( int age)

   public Object clone()

   public int getWeight()

   public int getHeight()

   public int getAge()

   public boolean equals( Object o )

   public int compareTo( Monster rhs )

   public String toString( )
}
```

```
removeMin() - 1 2 3

[2 1 3, 2 2 2, 2 3 1, 3 2 1, 3 1 2, 3 3 3]
```