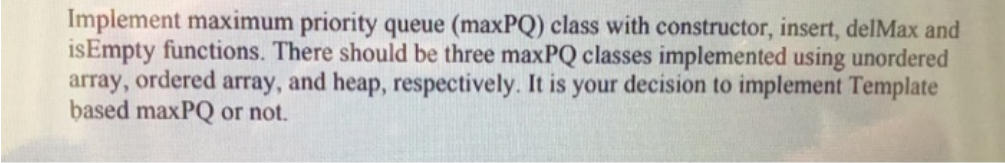


Question:

In c++



Implement maximum priority queue (maxPQ) class with constructor, insert, delMax and isEmpty functions. There should be three maxPQ classes implemented using unordered array, ordered array, and heap, respectively. It is your decision to implement Template based maxPQ or not.

Show transcribed image text

Answer

```
//main.cpp
#include "MaxPQ.h"
using namespace std;

int main()
{
    MaxPQ m(7);
    m.insert(3);
    m.insert(4);
    m.delMax();
    m.show();
    return 0;
}

-----
//MaxPQ.cpp
#include "MaxPQ.h"

MaxPQ::MaxPQ(int n)
{
    queue = new int[n];
    length = n;
    index = 0;
}

void MaxPQ::insert(int i)
{
    if(index < length)
    {
        queue[index] = i;
        index++;
    }
    else
    {
        printf("Ran out of spaces\n");
    }
}

int MaxPQ::delMax()
{
    int largest = 0;
    int li = 0;
    for(int i = 0; i <= index; i++)
    {
        if(queue[i] > largest){
            largest = queue[i];
            li = i;
        }
    }
    queue[li] = 0;
    return largest;
}

void MaxPQ::show()
{
    for(int i = 0; i < length; i++){
        printf("%d\n", queue[i]);
    }
}

bool MaxPQ::isEmpty()
{
}
```

```

bool foundNZ = false;
for(int i = 0; i < length; i++) // length not defined
{
    if (i != 0) foundNZ = true;
}
return foundNZ;
}

```

```

//MaxPQ.h
#ifndef MAXPQ_H
#define MAXPQ_H

```

```

#include <stdio.h>

```

```

using namespace std;

```

```

class MaxPQ
{
private:
    int *queue;
    int index;
    int length;
public:
    MaxPQ(int n);
    void insert(int i);
    int delMax();
    bool isEmpty();
    void show();
};

```

```

#endif

```

```

1  #include "MaxPQ.h"
2  using namespace std;
3
4  int main()
5  {
6      MaxPQ m(7);
7      m.insert(3);
8      m.insert(4);
9      m.delMax();
10     m.show();
11     return 0;
12 }

```

Run: MaxPriorityQueue

```

/Users/swapnil/CLionProjects/MaxPriorityQueue/cmake-build-debug/MaxPriorityQueue
3
0
0
-1610612736
-1748566000
32767
1590607165
Process finished with exit code 0

```