

Data Structure and Algorithm Analysis---COP3530

Program – Unit 8

NO LATE ASSIGNMENTS WILL BE ACCEPTED!!

Objective:

In this assignment you will do the following:

(1) manipulate pointers, (2) allocate memory dynamically, (3) implement a default constructor, copy constructor and destructor, (4) **use only one pointer to add to the back and to dequeue from the front of the queue.**

Assignment Description:

You will implement a doubly-linked circular queue of integers.

Consider the following class declarations when implementing BQUEUE. As always, you must comment your declaration and implementation files, "BQUEUE.h" and "BQUEUE.cpp", respectively.

```
class bqnode
{
    public:
        int time;
        bqnode *prev, *next;
};

class BQUEUE
{
    public:
        BQUEUE( );
        ~BQUEUE( );
        BQUEUE(const BQUEUE &);
        void Enqueue(int);
        void Dequeue( );
        void Print( );
    private:
        bqnode *front;//use ONLY one pointer
};
```

Use the following driver called "BQUEUE_driver.cpp" to test your code:

```
#include <iostream>
#include "BQUEUE.h"

using namespace std;

int main( )
{
```

```
BQUEUE k;
```

```
k.Enqueue(60);  
k.Print();  
k.Enqueue(20);  
k.Enqueue(30);  
k.Print();  
k.Enqueue(10);  
k.Print();  
k.Enqueue(50);  
k.Enqueue(40);  
k.Print();
```

```
BQUEUE j = k;
```

```
j.Dequeue();  
j.Print();  
j.Dequeue();  
j.Dequeue();  
j.Dequeue();  
j.Print();  
j.Dequeue();  
j.Dequeue();  
j.Print();  
j.Dequeue();  
j.Dequeue();
```

```
return 0;
```

```
}
```

Good luck...

Submit the header file "BQUEUE.h", "BQUEUE.cpp", and "BQUEUE_driver.cpp" in a zip file called **“program6_queue” to Blackboard before the due date and time.**

Good Luck....