DSA Lab

Mr. ALEEM AHMAD

Bahria University

Lab # 4
Queue Implementation

LAB Journal

Asim Ali (01-131232-015)

Lab 4: Queue Implementation

TASK:

Queue Implementation.

Lab Task GitHub Link:

Link

CODE:

```
#include<iostream>

using namespace std;

class DynamicQuene {
private:
    int* arr;
      int size;
      int currentSIZE;
      int front;
      int rear;
      int* value;

public:
      //s is the size of the array
      DynamicQuene(int s) {
            arr = new int[s];
            size = s; //size of the array
            currentSIZE = 0; //current size of the array
            front = -1;
            rear = -1;
            value = new int[s];
      }

      void insert(int value) {
            //when the quene capacity gets full
            if (currentSIZE == size) {
                  cout << "Quene Overflow" << endl;
                  return;
            }
            rear = (rear + 1) % size;
            arr[rear] = value;
            currentSIZE++;
            if (front == -1)
                  front = rear;
```

```
	}
	void remove() {
		value = &arr[front];
		front = (front + 1) % size;
		currentSIZE--;

	}
	void show() {
		for (int i = 0; i< currentSIZE ; i++)
		{
			int index = (front + i) % size;
			if (index >rear)
				return;
			cout << arr[index] << endl;

		}
	}

};

int main() {
	DynamicQuene s(5);
	cout << "Before Removing From Queue" << endl;
	s.insert(1);
	s.insert(2);
	s.insert(3);
	s.show();
	cout << "After Removing(1,2) From Queue" << endl;
	s.remove();
	s.remove();
	s.show();
	cout << "After Inserting Two Values in Queue." << endl;
	s.insert(1);
	s.insert(2);
	s.show();
	s.insert(6);


 }
```

OUTPUT:

```
Before Removing From Queue
1
2
3
After Removing(1,2) From Queue
3
After Inserting Two Values in Queue.
3
1
2
```