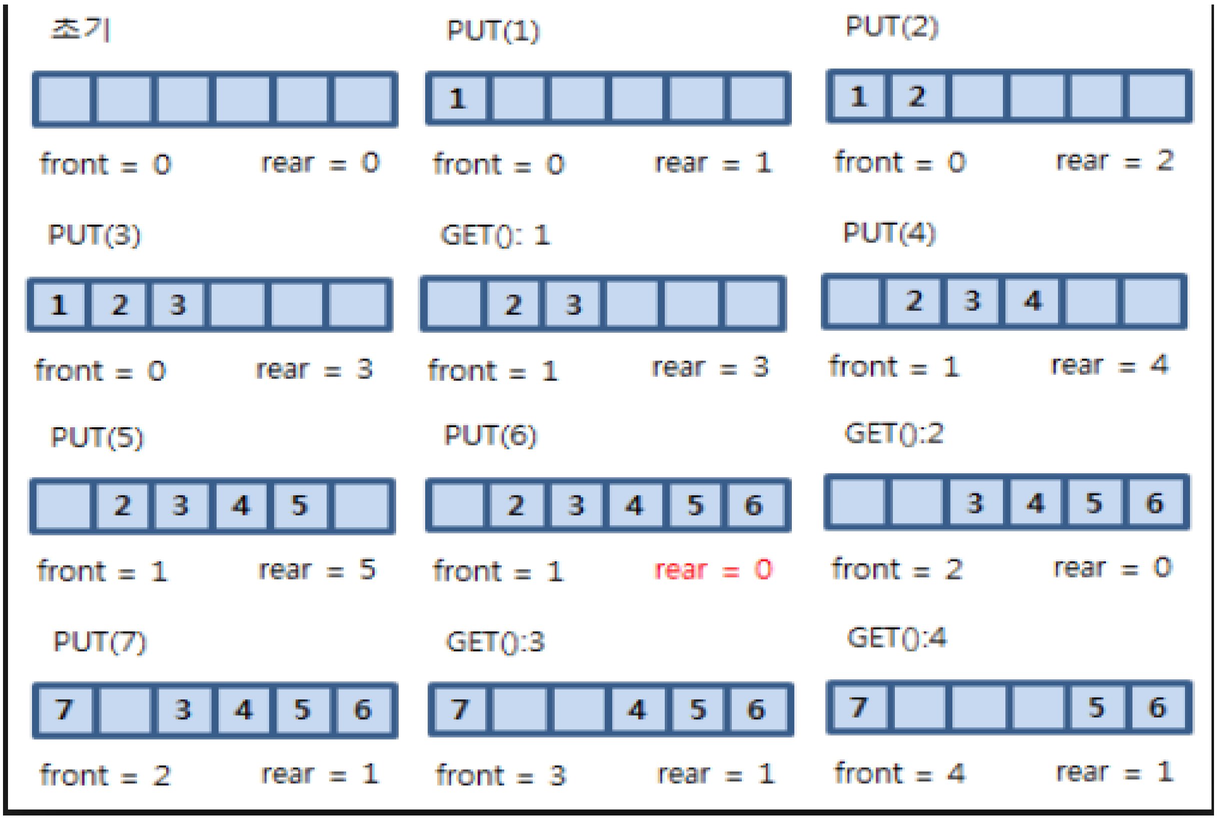
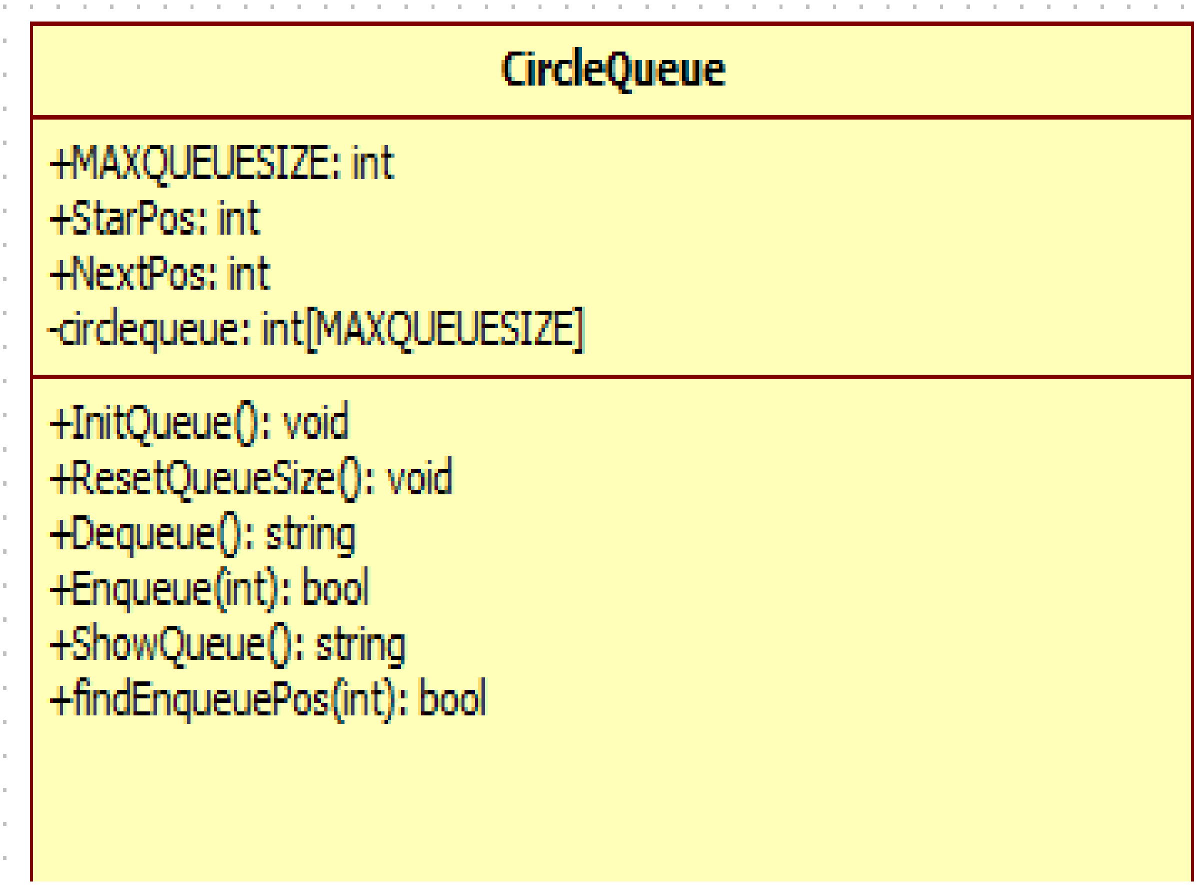
원형 큐 구현

이재건



구현한 원형 큐 모델

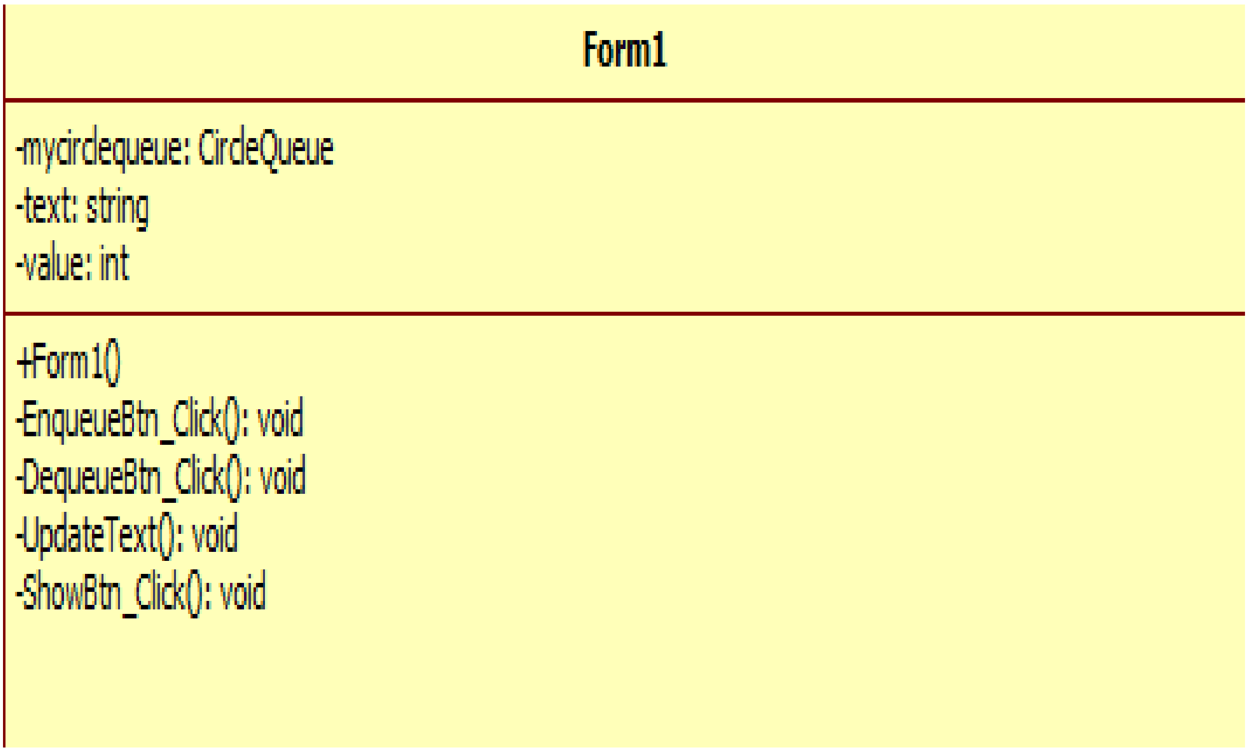
구현한 원형 큐 UML



CircleQueue

+MAXQUEUESIZE: int
+StarPos: int
+NextPos: int
-circlequeue: int[MAXQUEUESIZE]

+InitQueue(): void
+ResetQueueSize(): void
+Dequeue(): string
+Enqueue(int): bool
+ShowQueue(): string
+findEnqueuePos(int): bool

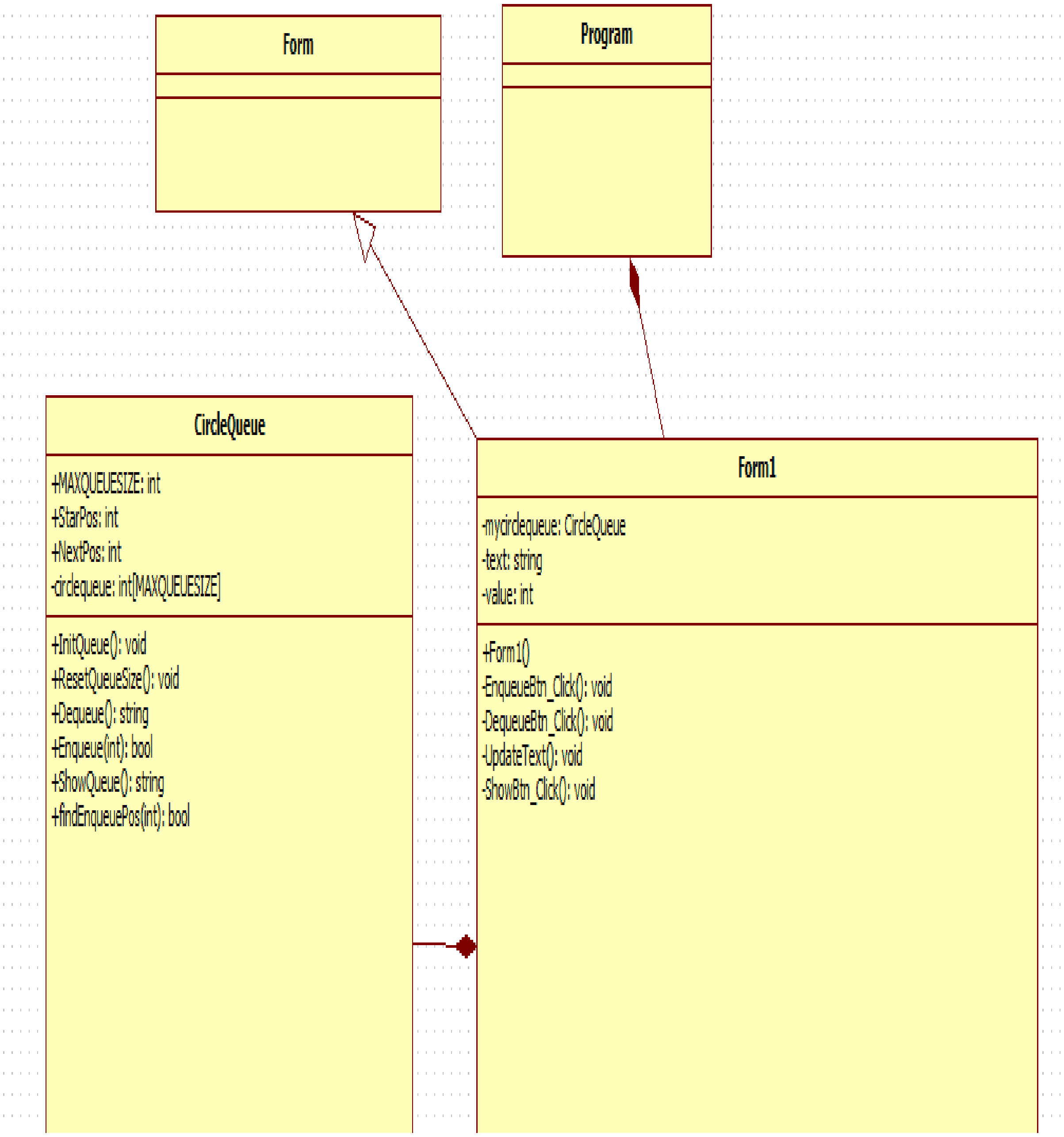


Form1

-mycirclequeue: CircleQueue
-text: string
-value: int

+Form1()
-EnqueueBtn_Click(): void
-DequeueBtn_Click(): void
-UpdateText(): void
-ShowBtn_Click(): void

전체적인 UML


Form
Program
CircleQueue
+MAXQUEUESIZE: int
+StarPos: int
+NextPos: int
-circlequeue: int[MAXQUEUESIZE]
+InitQueue(): void
+ResetQueueSize(): void
+Dequeue(): string
+Enqueue(int): bool
+ShowQueue(): string
+findEnqueuePos(int): bool
Form1
-mycirclequeue: CircleQueue
-text: string
-value: int
+Form1()
-EnqueueBtn_Click(): void
-DequeueBtn_Click(): void
-UpdateText(): void
-ShowBtn_Click(): void

Main Code 설명

Data를 큐에 insert 하는 함수

```
        public bool Enqueue(int value)
        {
            if (circlequeue[Nextpos] == 0)
            {
                circlequeue[Nextpos] = value;
                if (Nextpos + 1 < MAXQUEUESIZE)
                    Nextpos++;
                return true;
            }
            else
                return findEnqueuePos(value);
```

배열의 startpos 부분부터 data를 제거하는 부분

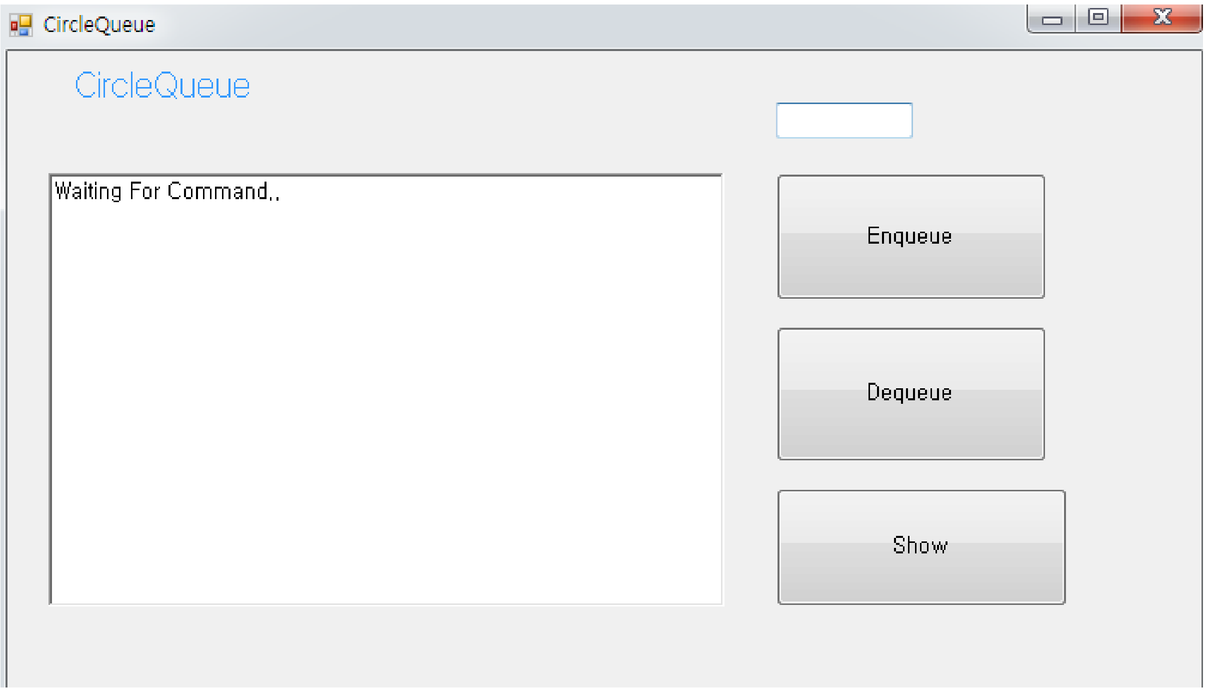
```
public string Dequeue()
        {
            string temp = "";
            if (circlequeue[Startpos] != 0)
            {
                temp = "Dequeue Button Clicked- " +
GetStartposValue() + " deleted! and Current StartPosPosition " +
CircleQueue.Startpos;
                circlequeue[Startpos] = 0;

                Startpos++;
                if (Startpos == MAXQUEUESIZE)
                    Startpos = 0;
            }
            else
            {
                temp = "There is no data for delete...";
            }
          //  setEnqeuePosition();
            return temp;
```

dequeue를 통해 큐 내부에서 비어있는 공간을 찾는 함수

```
public bool findEnqueuePos(int value)
{
    bool exitpoiont = false;
    int position = Nextpos;

    while (!exitpoiont)
    {
        position++;
        if (position >= MAXQUEUESIZE)
            position = 0;

        if (circlequeue[position] == 0)
        {
            Nextpos = position;
            exitpoiont = true;
            circlequeue[Nextpos] = value;
            Nextpos++;

        }
        else if (position == Nextpos)
        {
            Nextpos = position;
            break;
        }
    }
    return exitpoiont;
```

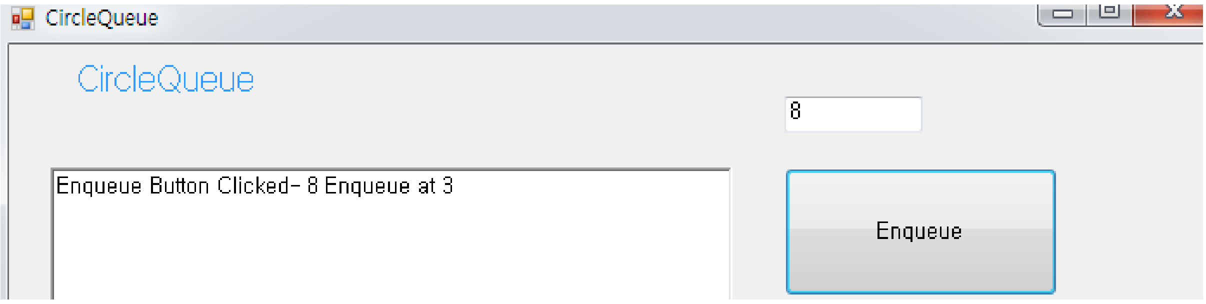
실행화면

1) 초기

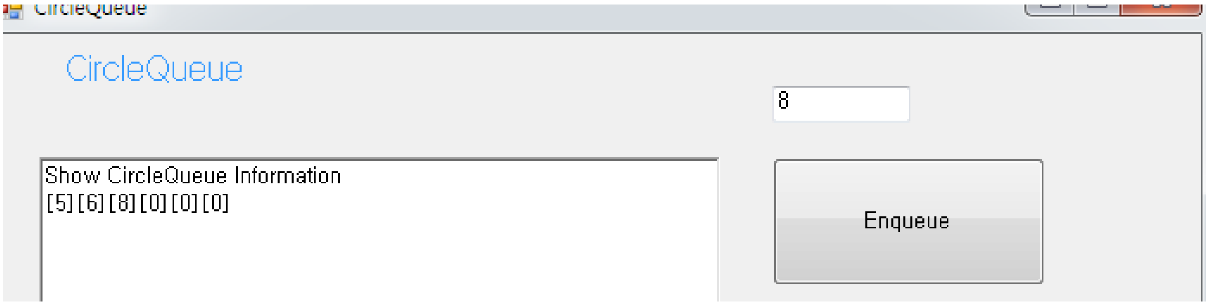


Waiting For Command..

2)Enqueue 후 show

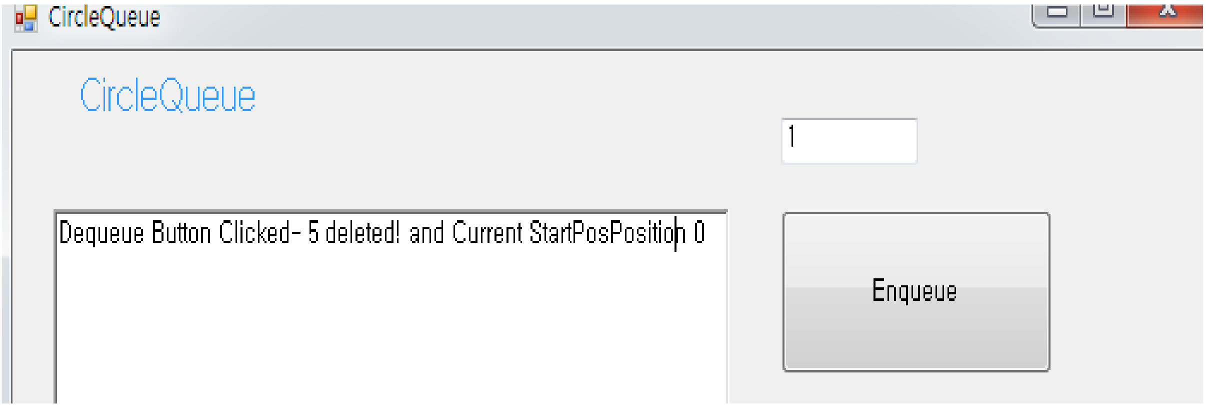


Enqueue Button Clicked- 8 Enqueue at 3

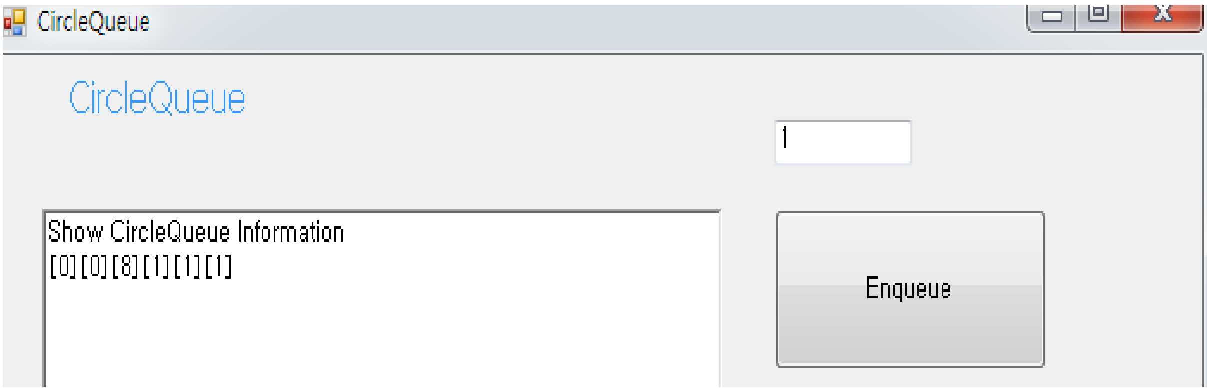


```
Show CircleQueue Information
[5][6][8][0][0][0]
```

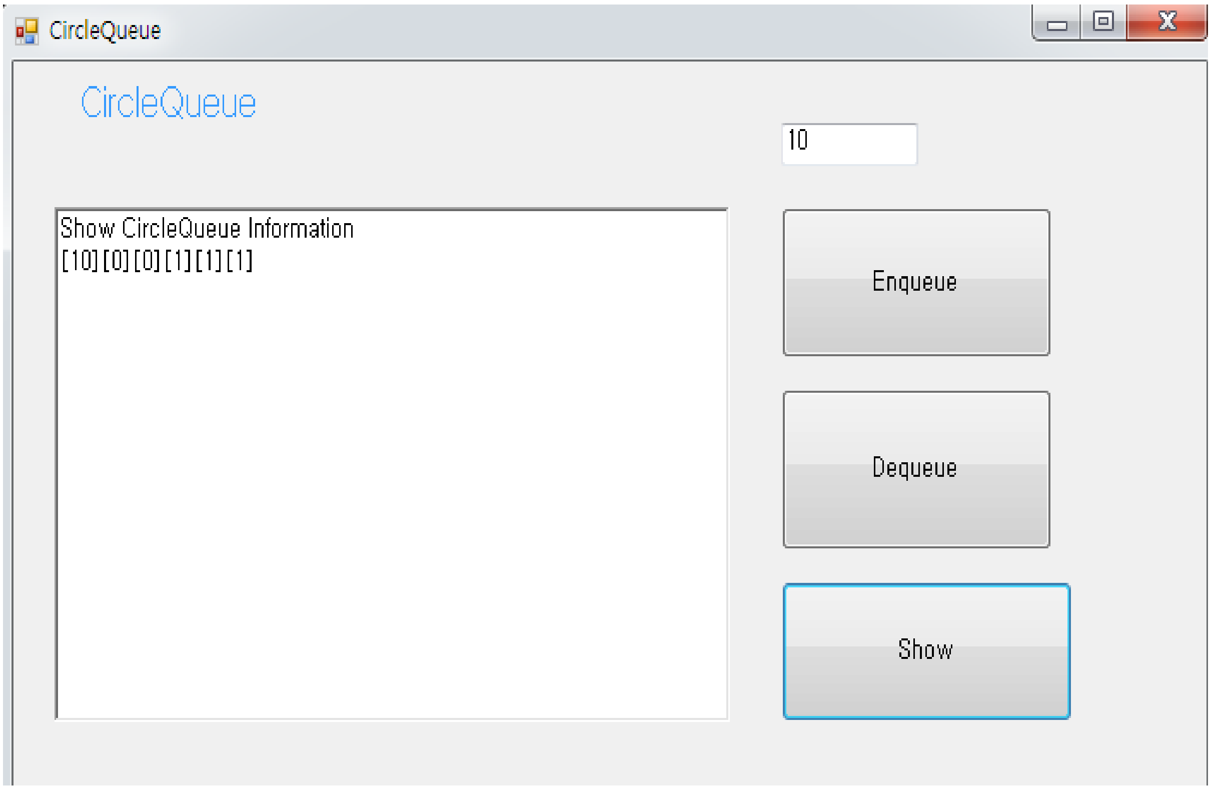
3)dequeue &enqueue 후 show



Dequeue Button Clicked- 5 deleted! and Current StartPosPosition 0



```
Show CircleQueue Information
[0][0][8][1][1][1]
```



```
Show CircleQueue Information
[10][0][0][1][1][1]
```