

선형구조(2)

1-1 큐(queue) – FIFO(First In First Out)

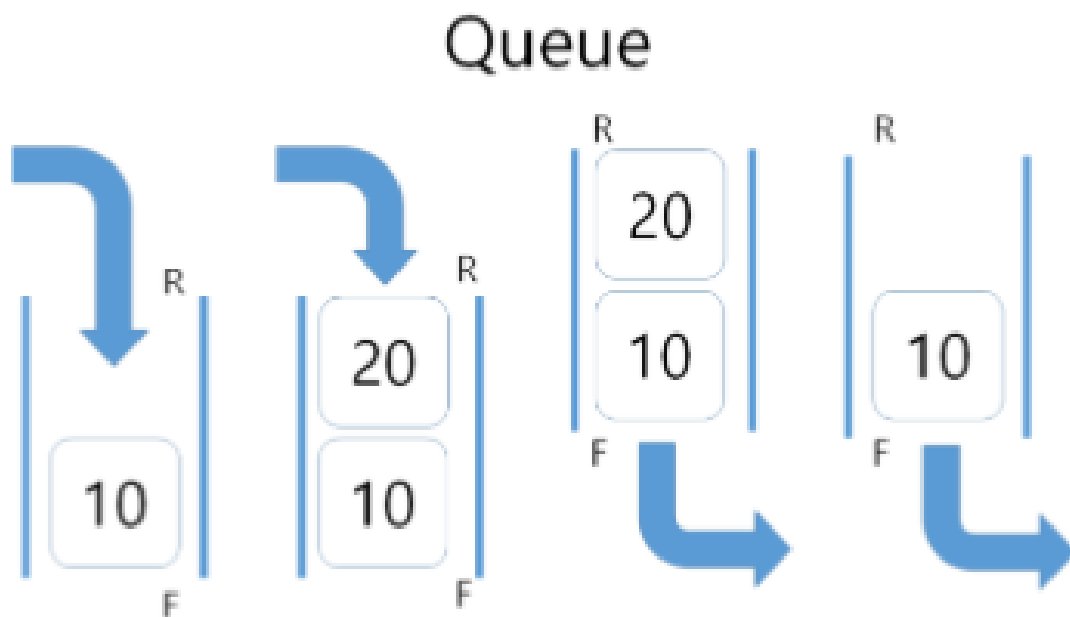


큐(queue)는 한쪽 방향으로 데이터가 삽입되고 반대 방향으로 데이터가 삭제되는 먼저

들어온 데이터가 먼저 나가는 선입 선출(First In First Out) 구조이다.

ex) 계산대에서 계산대에 먼저 도착한 고객이 먼저 계산하고 나간다.

버스 승강장에서 앞에 선 사람이 먼저 승차한다.



- 아무 것도 없는 큐에서 데이터 삭제가 일어나면 Under Flow Error 발생하고
- 반대로 큐에 다 채워져 있는 상태에서 삽입 시 Over Flow Error 가 발생한다.
- **프런트(F, Front)**

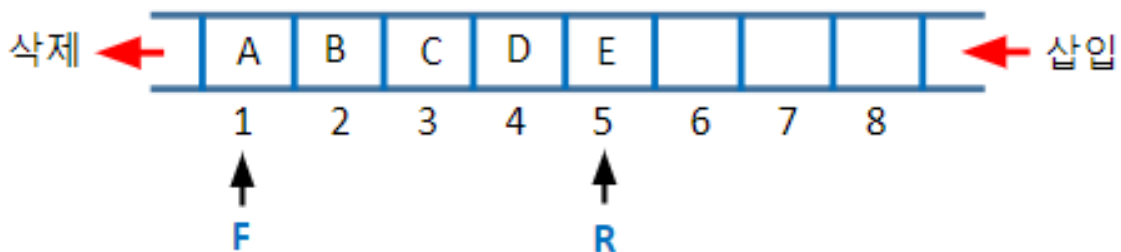
- ? -가장 먼저 삽입된 자료의 기억 공간을 가리키는 포인터이다.
-삭제 작업을 할 때 사용

- 리어(R, Rear)

- ? - 가장 마지막에 삽입된 자료가 위치한 기억 장소를 가리키는 포인터이다.
- 삽입 작업을 할 때 사용

- Queue의 응용 분야

- ? -창구 업무나 택시 정거장처럼 서비스순서를 기다리는 등의 대기 행렬의 처리에 사용한다.
-운영 체제의 작업 스케줄링에 사용



자바로 기초틀만 잡은 코드입니다.

```
public class IntQueue {
    private int[] que; //큐용 배열
    private int capacity; //큐의 크기
    private int front; //맨 처음 요소 커서
    private int rear; //맨 끝 요소 커서
    private int num; //현재 데이터 개수

    // 실행시 예외: 큐가 비어있음
    public class EmptyIntQueueException extends RuntimeException{
        public EmptyIntQueueException(){}
    }
}
```

```

// 실행시 예외: 큐가 가득 찼음
public class OverflowIntQueueException extends RuntimeException{
    public OverflowIntQueueException(){}
}

//생성자(constructor)
public IntQueue(int maxlen){
    num = front = rear = 0; //세개의 변수를 초기화
    capacity = maxlen;
    try{
        que = new int[capacity]; // 큐 본체용 배열을 생성
    }catch (OutOfMemoryError e){ //생성할 수 없음
        capacity = 0;
    }
}

//큐에 데이터를 인큐
public int enqueue(int x) throws OverflowIntQueueException{
    if (num>=capacity) //현재 배열의 크기보다 데이터의 개수가 많다면?
        throw new OverflowIntQueueException(); //예외처리
    que[rear++] = x; //맨 끝값에 값을 삽입
    num++; //데이터의 수를 1증가
    if(rear == capacity) //끝값이 배열의 크기와 같다면?
        rear = 0; //끝값을 0으로 초기화
    return x; //x의 값을 반환
}

//큐에서 데이터를 디큐
public int dequeue()throws EmptyIntQueueException{
    if (num == 0) //데이터의 크기가 0이라면
        throw new EmptyIntQueueException(); //예외처리
    int x = que[front++]; //첫번째 값을 증가
    num--; //데이터의 크기를 1 감소
    return x; //x를 반환
}
}

```

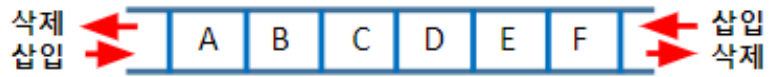
데크(Deque) – Double Ended Queue



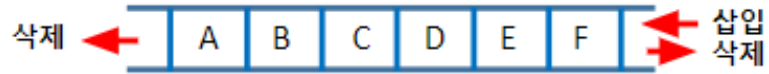
삽입과 삭제가 양쪽 끝에서 모두 발생할 수 있는 자료 구조이다.

특징

Stack과 Queue의 장점만 따서 구성한 것



- 입력 제한 덱 : Scroll



- 출력 제한 덱 : Shelf

