

자료구조 실습과제 06 - 큐 응용

- 솔루션 및 프로젝트 명칭 : Proj_06_이름이니셜
- 제출방법 : 아래 문제를 해결하기 위한 프로그램을 구현한 후 컴파일 및 실행한 후, 오류가 없으면 메뉴에서 솔루션 정리를 수행한 후 윈도우 탐색기에서 솔루션 폴더를 찾아 압축하여 E-class에 올림

3개의 문제를 하나의 프로젝트에 포함시켜 수행하기 위해 다음의 구조를 이용한다.

```
#define _CRT_SECURE_NO_WARNINGS

#define PROB 1 // 각각의 문제를 구현하고 해당 문제 번호를 변경하여 테스트

#if PROB == 1
// 1번 문제를 해결하기 위한 프로그램 작성

#elif PROB == 2
// 2번 문제를 해결하기 위한 프로그램 작성

#elif PROB == 3
// 3번 문제를 해결하기 위한 프로그램 작성

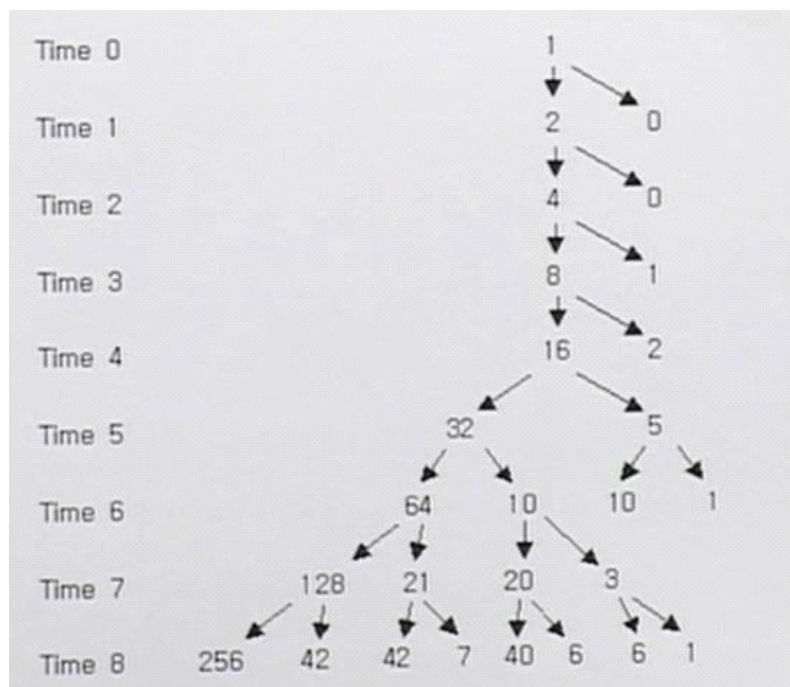
#endif
```

문제 1) 바이러스 생성 문제

바이러스 1 마리가 있다. 이 바이러스의 수는 1초 후에 2배로 늘어날 수도 있고 1/3(소숫점 이하 버림)로 줄 수도 있다. 현재 몇 마리의 바이러스가 존재하는지 주어질 때 1마리의 바이러스에서부터 최소 몇 초의 시간이 흘러 현재 상태가 되었는지를 구하는 프로그램을 작성하시오.

예로 현재 바이러스가 7마리 있다면, 1 -> 2 -> 4 -> 8 -> 16-> 32 -> 64 -> 21 -> 7 이 보다 더 빠른시간 안에 7마리가 될 수는 없다. 따라서 답은 8초이다.

<시간대 별로 존재할 수 있는 바이러스의 수>



알고리즘

바이러스의 수가 N마리, 지난 시간이 T초라고 할 때 이를 (N, T)라고 하자.

step 1 : 초기 상태 (1, 0)을 큐에 넣는다.

step 2 : 큐에서 하나를 끄집어 낸다. 그 값이 (N, T)라면 그 다음 상태인 (2*N, T+1) 과 ({N/3}, T+1)을 큐에 집어 넣는다.

step 3 : 큐에서 하나를 끄집어 냈을 때 바이러스의 수가 구하고자 하는 수라면 그 때의 시간을 출력한다.

참고 : 탐색 공간을 줄이기 위해서 체크 배열을 하나 두어, 새롭게 구한 바이러스 수가 이미 한 번 나왔다면 다음 상태로 진행시키지 않고 넘어갈 수 있다. 아주 큰 숫자의 바이러스 생성을 위해 큐의 데이터를 담는 공간을 동적으로 할당함. 큐의 크기를 `#define MAX_QUEUE_SIZE 1000000`로 정의하여 사용함

실행 예 (Queue의 크기를 #define MAX_QUEUE_SIZE 1000000로 했을 때 실행결과이다. 다만 큐가 가득찬 경우 에러를 출력하고 프로그램을 종료한다. Queue의 크기에 따라 오류를 다르게 발생할 수 있음.)



문제 2) 살아남기 게임

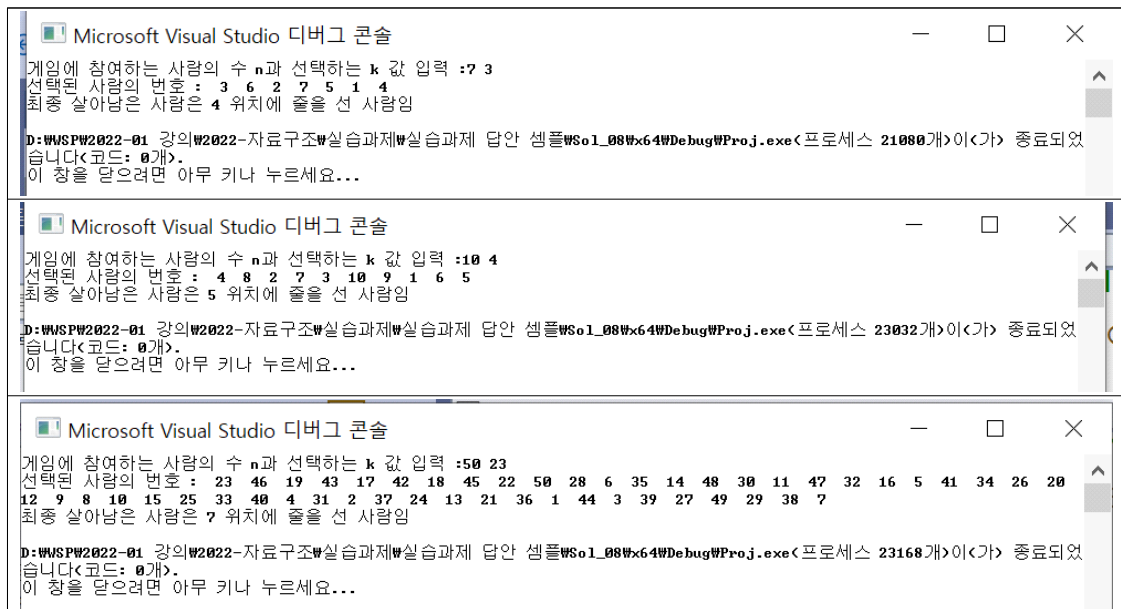
n명의 포로를 일렬로 세워놓고 1부터 순서를 세어 k번째 포로를 죽이는 오징어 게임이 시작되었다. k-1번째까지의 포로는 같은 순서로 뒤로 가서 다시 줄을 서게 된다. 지속적으로 k번째 포로를 죽일 때 최종까지 살아남으려면 첫 번째 줄을 설 때 몇 번째 위치에 줄을 서야 하는지 알 수 있는 프로그램을 작성하시오. 즉, n명 중 k번째 사람을 죽이는 경우의 순서를 출력하는 프로그램을 작성하면 된다.

입력은 처음 줄을 서게 되는 n값과 k 값이다. 단, $n \geq 1$, $1 \leq k \leq n$ 임.

출력은 k번째 죽게되는 사람의 처음 줄을 선 위치 정보, 즉 n명이 줄을 서게 되면 각 사람의 위치를 1, 2, ..., n으로 할당하고, 연속적으로 k번째 선택되었을 때 선택된 번호를 출력하면 된다.

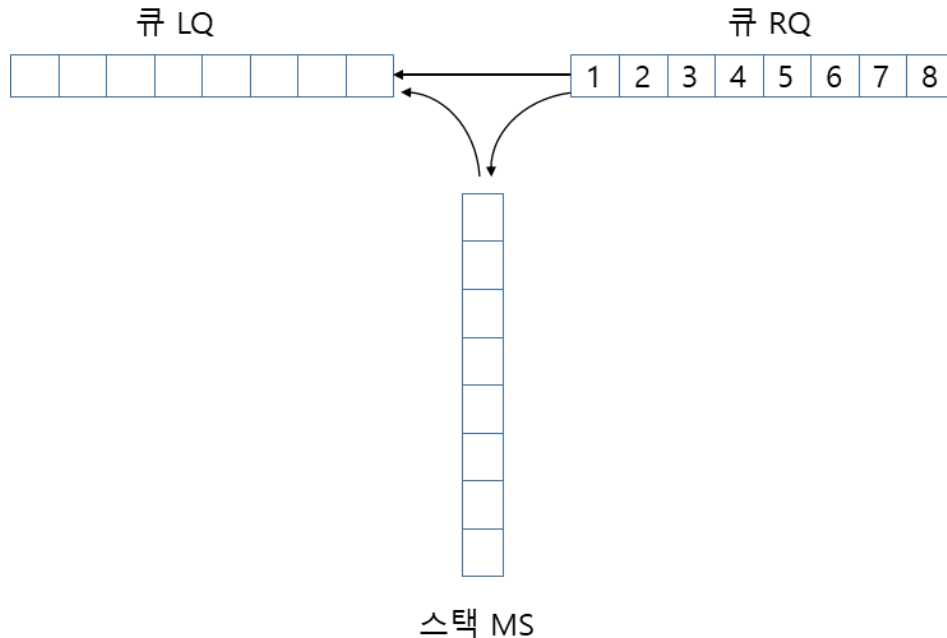
예 : n=7, k=3이면 출력은 3, 6, 2, 7, 5, 1, 4로 되어 처음에 4번째 줄을 서면 최종 살아남게 된다.

실행 예 : 입력값은 n과 k 값임



문제 3) 스택 수 판별하기

스택 수란 아래 그림과 같은 장치를 가지고 만들어 낼 수 있는 수열을 의미한다. 장치는 1 부터 n까지 순서대로 저장되어 있는 큐 RQ와 만들어낼 수열을 저장하는 큐 LQ가 있으며, 임시로 데이터를 저장하기 위한 스택 MS가 있다. 이 장치는 오른쪽에 있는 RQ에서 하나의 데이터를 읽어와서 스택 MS에 저장하거나 큐 LQ에 바로 저장할 수 있다. 또한 스택 MS에 있는 데이터를 큐 LQ로도 이동할 수 있다.



스택 수를 생성하는 장치를 이용하여 입력받은 수열이 생성 가능한지(LQ에 같은 수열이 만들어지는지) 점검하여 가능 여부를 출력하시오.

입력 : RQ에 들어가는 데이터의 크기 n, 테스트할 수열(n개로 구성됨)

출력 : 이 장치로 생성가능한지 여부(가능, 불가능)

예 : 입력 3

출력 : 가능

1 3 2

테스트를 위하여 main 함수 내에서 n값과 테스트 수열을 다음과 같이 초기화하여 사용한다.

```
int n = 10;
```

```
int testNumberSequence[MAX_QUEUE_SIZE] = { 3, 4, 6, 7, 8, 5, 10, 2, 1, 9 };
```

실행 예

