

BOJ 9019

# DSLR

소프트웨어학과 201921017 이지우

# 1. 접근

## 문제

$$\begin{array}{cccc} 0-1 & 1000 \times 2 & d_1=1 & 0 \\ = 999 & n & d_2=0 & 0 \end{array}$$

네 개의 명령어 D, S, L, R 을 이용하는 간단한 계산기가 있다. 이 계산기에는 레지스터가 하나 있는데, 이 레지스터에는 0 이상 10,000 미만의 십진수를 저장할 수 있다. 각 명령어는 이 레지스터에 저장된  $n$  을 다음과 같이 변환한다.  $n$  의 네 자릿수를  $d_1, d_2, d_3, d_4$  라고 하자(즉  $n = ((d_1 \times 10 + d_2) \times 10 + d_3) \times 10 + d_4$  라고 하자)

1. D: D 는  $n$  을 두 배로 바꾼다. 결과 값이 9999 보다 큰 경우에는 10000 으로 나눈 나머지를 취한다. 그 결과 값( $2n \bmod 10000$ )을 레지스터에 저장한다.
2. S: S 는  $n$  에서 1 을 뺀 결과  $n-1$  을 레지스터에 저장한다.  $n$  이 0 이라면 9999 가 대신 레지스터에 저장된다.
3. L: L 은  $n$  의 각 자릿수를 왼편으로 회전시켜 그 결과를 레지스터에 저장한다. 이 연산이 끝나면 레지스터에 저장된 네 자릿수는 왼편부터  $d_2, d_3, d_4, d_1$  이 된다.
4. R: R 은  $n$  의 각 자릿수를 오른편으로 회전시켜 그 결과를 레지스터에 저장한다. 이 연산이 끝나면 레지스터에 저장된 네 자릿수는 왼편부터  $d_4, d_1, d_2, d_3$  이 된다.

위에서 언급한 것처럼, L 과 R 명령어는 십진 자릿수를 가정하고 연산을 수행한다. 예를 들어서  $n = 1234$  라면 여기에 L 을 적용하면 2341 이 되고 R 을 적용하면 4123 이 된다.

여러분이 작성할 프로그램은 주어진 서로 다른 두 정수 A와 B( $A \neq B$ )에 대하여 A를 B로 바꾸는 최소한의 명령어를 생성하는 프로그램이다. 예를 들어서  $A = 1234, B = 3412$  라면 다음과 같이 두 개의 명령어를 적용하면 A를 B로 변환할 수 있다.

$$A \rightarrow B \quad pp \quad b$$

$$\rightarrow 1234 \rightarrow_L 2341 \rightarrow_L 3412$$

$$\rightarrow 1234 \rightarrow_R 4123 \rightarrow_R 3412$$

따라서 여러분의 프로그램은 이 경우에 LL 이나 RR 을 출력해야 한다.

$n$  의 자릿수로 0 이 포함된 경우에 주의해야 한다. 예를 들어서 1000 에 L 을 적용하면 0001 이 되므로 결과는 1 이 된다. 그러나 R 을 적용하면 0100 이 되므로 결과는 100 이 된다.

# 1. 접근

## 입력

프로그램 입력은 T개의 테스트 케이스로 구성된다. 테스트 케이스 개수 T는 입력의 첫 줄에 주어진다. 각 테스트 케이스로는 두 개의 정수 A와 B( $A \neq B$ )가 공백으로 분리되어 차례로 주어지는데 A는 레지스터의 초기 값을 나타내고 B는 최종 값을 나타낸다. A와 B는 모두 0 이상 10,000 미만이다.

## 출력

A에서 B로 변환하기 위해 필요한 최소한의 명령어 나열을 출력한다. 가능한 명령어 나열이 여러가지면, 아무거나 출력한다.

### 예제 입력 1 복사

```
3
1234 3412
1000 1
1 16
```

### 예제 출력 1 복사

```
→ LL
→ L
→ DDDD
```

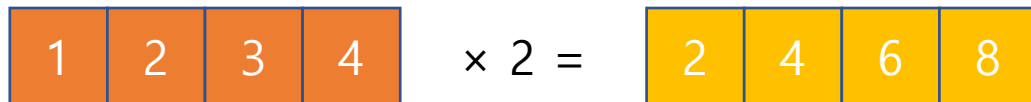
1 2 3 4

2 3 4 1

3 4 1 2

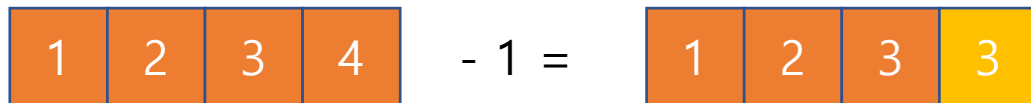
# 1. 접근

**D**



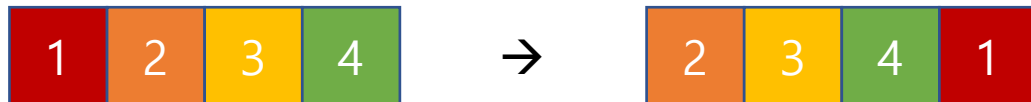
결과값 > 9999 → 결과값 %= 10000

**S**

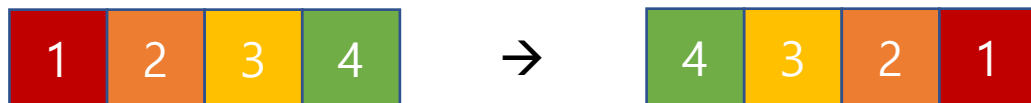


0000 - 1 → 9999

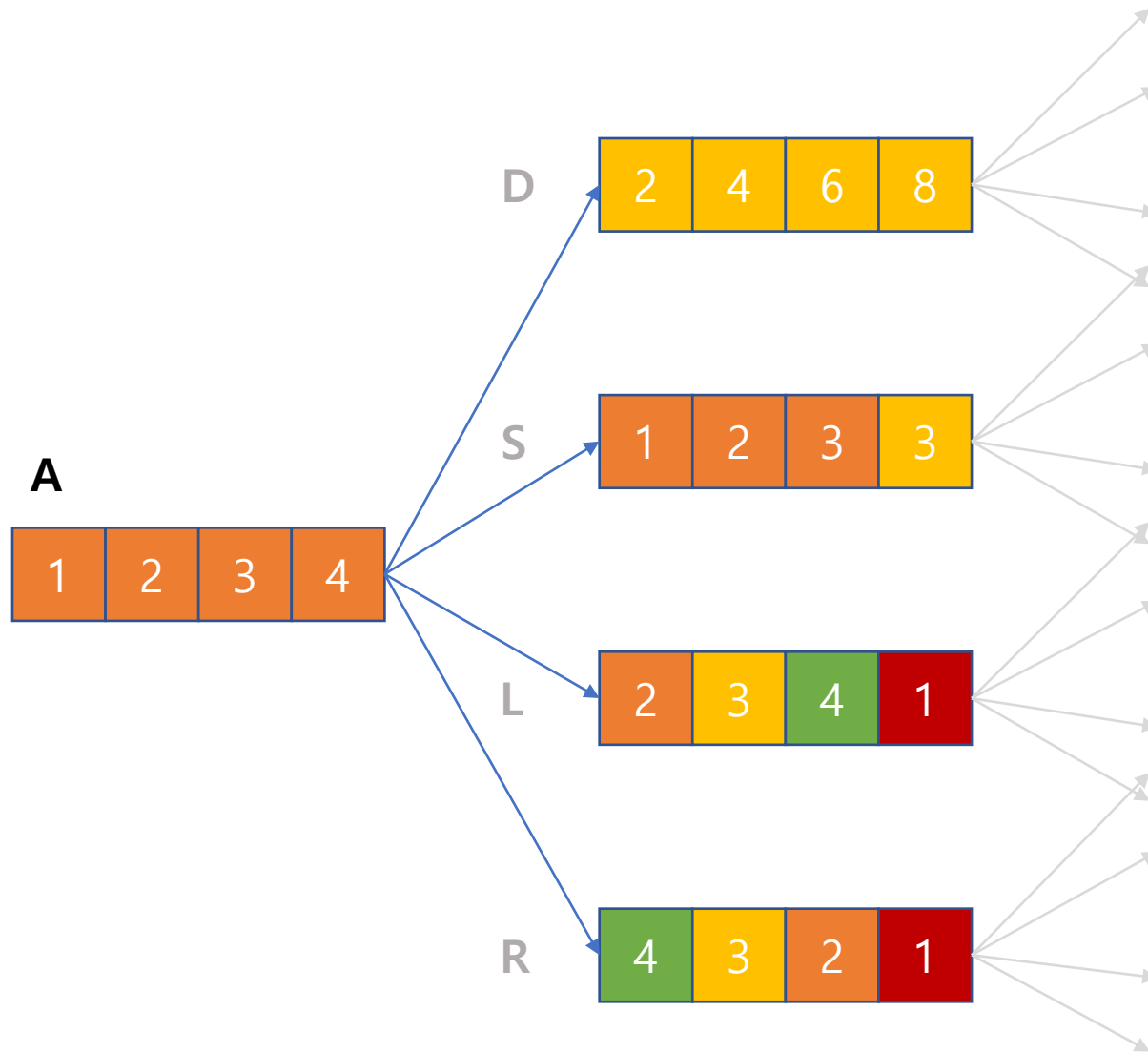
**L**



**R**



## 1. 접근



## 가능한 네가지의 길

1. 조건에 맞게 4가지 경우에 대하여 BFS 탐색
2. 변환된 **값**과,  
그 값을 얻기까지의 **명령어**를 쌍으로 저장
3. 큐에서 뽑아낸 **값** == B 이면,  
해당 **명령어** 출력

## 2. 구현

```
string convertTo4DigitNumber(int num) {  
    string numStr = to_string(num);  
  
    while (numStr.length() != 4) {  
        numStr = "0" + numStr;  
    }  
  
    return numStr;  
}
```

정수를 길이가 4인 문자열로  
변환해주는 함수

12 → "0012"

```
string toLeft(string numStr) {  
    char temp = numStr[0];  
  
    numStr[0] = numStr[1];  
    numStr[1] = numStr[2];  
    numStr[2] = numStr[3];  
    numStr[3] = temp;  
  
    return numStr;  
}  
  
string toRight(string numStr) {  
    char temp = numStr[3];  
  
    numStr[3] = numStr[2];  
    numStr[2] = numStr[1];  
    numStr[1] = numStr[0];  
    numStr[0] = temp;  
  
    return numStr;  
}
```

4자리수 문자열을

각각 왼쪽, 오른쪽으로

회전시키는 함수

"1234" → "4123"

"1234" → "2341"

## 2. 구현

```
int main()
{
    ios_base::sync_with_stdio(false);
    cin.tie(NULL);
    cout.tie(NULL);

    int t;
    cin >> t;

    while(t--) {

        int a, b;
        cin >> a >> b;

        queue<pis> q;
        q.push({a, ""});

        pis curr;

        while (true) {

            curr = q.front(); q.pop();

            if (curr.first == b) break;

            int nextNum;
```

```
            // D
            nextNum = curr.first << 1;
            if (nextNum > 9999) nextNum %= 10000;
            q.push({nextNum, curr.second + "D"});

            // S
            if (curr.first == 0) {
                nextNum = 9999;
            } else {
                nextNum = curr.first - 1;
            }
            q.push({nextNum, curr.second + "S"});

            // L
            nextNum = stoi(toLeft(convertTo4DigitNumber(curr.first)));
            q.push({nextNum, curr.second + "L"});

            // R
            nextNum = stoi(toRight(convertTo4DigitNumber(curr.first)));
            q.push({nextNum, curr.second + "R"});


        }

        cout << curr.second << endl;

    }

    return 0;
}
```

### 3. 결과

제출 번호	아이디	문제	결과	메모리	시간	언어	코드 길이
52294871	january	 9019	메모리 초과			C++17 / 수정	2003 B