

문제 1\_1: 문자열뒤집기(난이도 하)(연습)

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <string.h>

void reverse(char src_s[], char s[])
{
    // 코드 작성
}

// main은 given
int main(void)
{
    char src_s[101];
    char s[101];

    scanf("%s", src_s);
    reverse(src_s, s);
    printf("%s\n", s);
}
```

How many carry operations? 과제는 다음과 같다. 두 숫자의 합을 구할 때, 오른쪽부터 차례로 한 자리씩 더한다. 이 때 그 합이 10보다 크면 왼쪽 자리에 1을 더하며 그것을 carry라 한다. 두 숫자를 더할 때 carry가 몇 번 발생하는지 구하는 것이다.

i) 입력 받은 숫자 두 개를 모두 역순으로 만든다. 이렇게 하는 이유는 1의 자리 수를 문자열의 가장 앞쪽으로 만들기 위해서이다.

1 → 두 번째 숫자

25 → 25번의 carry가 발생

## 문제 2: 9배수 하나씩(난이도 중하)(프로그래머스 제출)

어떤 정수  $n$ 을 읽어서 그의 9배되는 수에 대해서 숫자 하나씩 공백을 두고 출력하는 프로그램을 작성하라.

즉 10을  $n$ 으로 읽으면 그의 9배되는 수인 90를 다음과 같이 출력한다. ( $n < 100000000$ )

9 0

실행예1

99 → 입력:  $n$

8 9 1 → 출력:  $99 * 9$ 는 891이므로

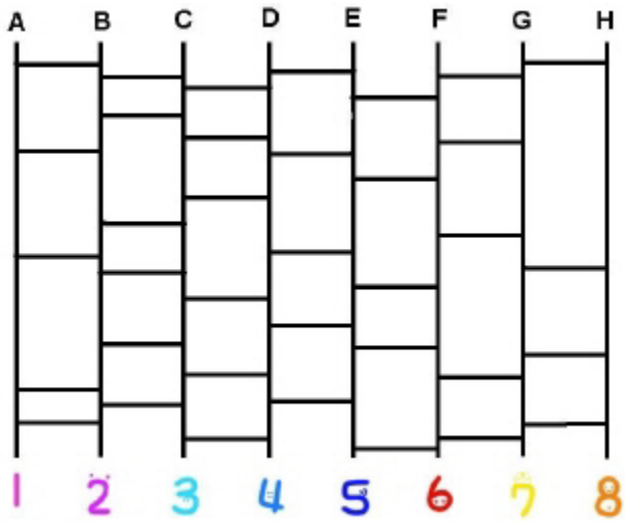
실행예2

7 → 입력:  $n$

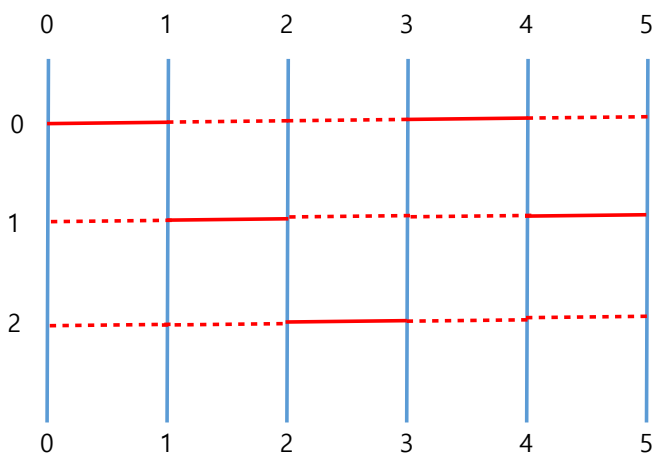
6 3 → 출력:  $7 * 9$ 는 63이므로

### 문제 3: 사다리 타기(난이도 중상)(프로그래머스 제출)

사다리 타기. 사다리 타기는 사람 수만큼 세로줄을 긋고 한쪽 편에는 이름을 쓰고 반대쪽에는 상품이나 물건을 쓴 뒤, 세로줄 사이사이에 가로줄을 무작위로 그은 다음, 세로줄을 타고 아래로 내려가면서 가로줄을 만날 때마다 가로줄로 연결된 다른 세로줄로 옮겨가는 게임이다.

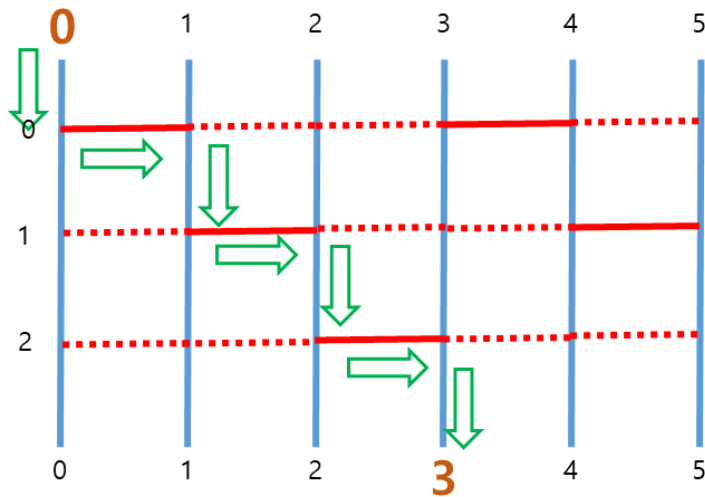


위 예시에서는 순서대로  $A \rightarrow 3$ ,  $B \rightarrow 6$ ,  $C \rightarrow 7$ ,  $D \rightarrow 1$ ,  $E \rightarrow 8$ ,  $F \rightarrow 2$ ,  $G \rightarrow 5$ ,  $H \rightarrow 4$  이런 결과가 나온다. 이번 문제는 사다리 타기를 구현하는 것이 목표이다. 사다리 타기를 프로그램으로 만들기 위해 사다리 타기의 세로 줄(사람이 고를 수 있는)  $N$ 개 그리고 가로줄을 넣을 수 있는 사다리의 높이를  $M$ 으로 고정한다.  $M$ 과  $N$ 의 최대는 각각 100이다.



위 그림은 사다리 타기의 세로 줄의 수가 총 6개이며, 사다리의 높이는 3이다. 사다리 타기의 가로줄은 점선의 경우는 그어지지 않은 경우이며 실선의 경우는 그어진 경우이다. 그어진 가로줄이 연속으로 나올 수는 없다 (이것은 사다리 그림의 원칙이다!)

이전 장의 예에서 0을 고르면 사다리를 타고 내려가다가 3의 위치로 나오게 된다.



여러분은 사다리 게임에서의 숫자로 된 시작 위치를 고르면 최종적으로 나오게 되는 숫자를 출력하는 프로그램을 작성해야 한다. 이를 위해 고를 수 있는 시작 위치와 최종 위치는 모두 0과 N-1 사이의 연속된 숫자로 되어 있다.

가로줄의 경우 1은 그어진 경우를 0은 그어지지 않은 경우를 표시하며 높이가 0인 줄부터 차례로 입력한다.

위의 예에서 사다리 타기 게임판은 아래와 같이 표시된다.

```
10010
01001
00100
```

여러분의 프로그램의 입력은 M+2 줄의 입력을 받는다. 첫 줄에는 사다리의 게임의 세로줄의 수(N)과 가로줄의 수(M)이 입력되며, 그 다음 줄은 시작줄의 위치(위의 예에서는 0), 그리고 그 다음 M줄은 사다리 타기의 모양이다.

위의 예의 경우에는 아래와 같은 입력을 여러분의 프로그램에게 준다.

실행예1:

입력)

6 3 (세로줄의 수 6, 가로줄의 수 3)

0 (시작 위치 0)

10010 → 사다리 타기의 모양

01001

00100

출력)

3 → 최종 위치

실행예2

입력)

6 3

1

10010

01001

00100

출력)

0 → 최종 위치