

Sogang Programming Contest 2019

Master Division Problems

22nd November, 2019
Sogang University

대회 규칙

- 대회 중 제출한 소스코드는 채점 서버에 의해 자동으로 채점되며, 실시간으로 결과를 알 수 있습니다. 문제를 제출하였을 때 '맞았습니다!!'를 받으면 문제를 푼 것으로, 이외의 결과를 받으면 틀린 것으로 생각합니다.
- 문제를 풀 때마다 패널티 점수가 누적됩니다. 패널티 점수는 모든 맞은 문제에 대해, 대회 시작 시간부터 그 문제를 풀기까지 걸린 시간을 t 분, 처음으로 문제를 맞기 직전까지 제출한 횟수를 w 번이라고 할 때 $(t + 20w)$ 점입니다.
- 순위는 푼 문제가 많은 순서대로, 푼 문제가 같을 경우에는 패널티 점수의 합이 적은 순서대로 결정됩니다.
- 사용 가능 언어는 C, C++, Java, Python 3, Kotlin입니다. 모든 문제는 출제진이 C++과 Java(혹은 Kotlin)으로 정답을 작성했음이 보장됩니다. 각 언어의 컴파일 옵션과 컴파일러 버전은 아래와 같습니다.

C11. gcc (GCC) 8.3.0

```
컴파일 gcc Main.c -o Main -O2 -Wall -lm -static -std=c99 -DONLINE_JUDGE -DBOJ
실행 ./Main
```

C++17. gcc (GCC) 8.3.0

```
컴파일 g++ Main.cc -o Main -O2 -Wall -lm -static -std=gnu++17 -DONLINE_JUDGE -DBOJ
실행 ./Main
```

Java. OpenJDK Runtime Environment (build 1.8.0_222-8u222-b10-lubuntu1 16.04.1-b10)

```
컴파일 javac -J-Xms1024m -J-Xmx1024m -J-Xss512m -encoding UTF-8 Main.java
실행 java -Xms1024m -Xmx1024m -Xss512m -Dfile.encoding=UTF-8 Main
```

Java 11. OpenJDK Runtime Environment (build 13+33)

```
컴파일 javac -J-Xms1024m -J-Xmx1024m -J-Xss512m -encoding UTF-8 Main.java
실행 java -Xms1024m -Xmx1024m -Xss512m -Dfile.encoding=UTF-8 Main
```

Python 3. Python 3.7.4

```
컴파일 python3 -c "import py_compile; py_compile.compile(r'Main.py')"
실행 python3 Main.py
```

PyPy3. PyPy 7.0.0 with GCC 6.2.0 20160901 (Python 3.5.3)

```
컴파일 python3 -c "import py_compile; py_compile.compile(r'Main.py')"
실행 pypy3 Main.py
```

Kotlin/JVM. kotlinc-jvm 1.3.50 (JRE 1.8.0_201-b09)

```
컴파일 kotlinc-jvm -J-Xms1024m -J-Xmx1024m -J-Xss512m -include-runtime -d \
Main.jar Main.kt
실행 java -Xms1024m -Xmx1024m -Xss512m -jar Main.jar
```

Kotlin/Native. kotlinc-native 1.3.50 (JRE 1.8.0_201-b09)

```
컴파일 kotlinc-native -o Main -opt Main.kt
실행 ./Main.kexe
```

- 네트워크 사용은 금지됩니다. 단, 솔루션을 제출하거나 언어 레퍼런스를 확인하는 것은 가능합니다. 접속이 허용된 사이트의 예는 다음과 같습니다.

C/C++. <https://en.cppreference.com/w/>

Java. <https://docs.oracle.com/javase/8/docs/api/>

Python. <https://docs.python.org/3/>

Kotlin. <https://kotlinlang.org/docs/reference/>

- 대회 종료 전에 퇴실할 수 없습니다.
- 책이나 개인이 준비한 인쇄된 참고자료를 열람할 수 있습니다. 참고자료의 양에는 제한이 없습니다.
- 대회 중에 휴대폰 및 전자기기는 사용할 수 없습니다.

문제 목록

인쇄되어 있는 문제가 총 8문제가 맞는지 확인하시기 바랍니다.

- A 1998년생인 내가 태국에서는 2541년생?!
- B 도깨비불
- C 블랙 프라이데이
- D 이진수 게임
- E 그르다 김가눔
- F 7-세그먼트 디스플레이
- G 단어 암기
- H 색깔 하노이 탑

모든 문제의 메모리 제한은 1GB로 동일합니다.

A. 1998년생인 내가 태국에서는 2541년생?!

시간 제한: 1초

ICPC Bangkok Regional에 참가하기 위해 수완나폼 국제공항에 막 도착한 팀 레드시프트 일행은 눈을 믿을 수 없었다. 공항의 대형 스크린에 올해가 2562년이라고 적혀 있던 것이었다.

불교 국가인 태국은 불멸기원(佛滅紀元), 즉 석가모니가 열반한 해를 기준으로 연도를 세는 불기를 사용한다. 반면, 우리나라는 서기 연도를 사용하고 있다. 불기 연도가 주어질 때 이를 서기 연도로 바꿔 주는 프로그램을 작성하시오.

입력

서기 연도를 알아보고 싶은 불기 연도 y 가 주어진다. ($1000 \leq y \leq 3000$)

출력

불기 연도를 서기 연도로 변환한 결과를 출력한다.

예제

입력	출력
2541	1998

B. 도깨비불

시간 제한: 1초

	Q	W	E	R	T	Y	U	I	O	P				
	ㅅ	ㅈ	ㅊ	ㅋ	ㆁ	ㆆ	㆏	㆑	㆓	㆔				
A	S	D	F	G	H	J	K	L						
	ㅏ	ㅑ	ㅓ	ㅕ	ㅗ	ㅛ	ㅝ	ㅟ	ㅛ	ㅜ				
Z	X	C	V	B	N	M								
	ㅞ	ㅘ	ㅚ	ㅜ	ㅠ	ㅡ								

한글 두벌식 자판을 사용하다 보면, 다른 입력 방식에서는 찾아보기 힘든 도깨비불 현상이라는 특이한 현상을 경험할 수 있다. 예를 들어 “서강대학교”를 입력할 때 글쇠 입력 하나하나를 추적해 보자:

```

ㅅ
서
석
서가
서강
서강ㄷ
서강대
서강땡
서강대하
서강대학
서강대학ㄱ
서강대학교

```

“석”에서 “서가”로 변할 때, 다음 글자의 초성이 될 자음이 입력하고 있는 글자의 종성 위치에 먼저 붙는 현상이 일어난다. 이를 ‘도깨비불 현상’이라고 한다.

“서강대학교”를 입력할 때는 도깨비불 현상이 2회 발생했다. 다른 예시도 보자.

입력	영문 모드 입력	도깨비불 횟수
이보세요	dlqhtpdy	3회
코딩하기 싫다	zheldgkrl tlfgek	2회
마춤삐비 틀려써요	akcnaQjql xmfFuTjdy	4회

- 이보세요: (“입” → “이보”), (“이봣” → “이보세”), (“이보쌩” → “이보세요”)
- 코딩하기 싫다: (“꼰” → “코딩”), (“코딩학” → “코딩하기”)
- 마춤삐비 틀려써요: (“맏” → “마추”), (“마춤뻐” → “마춤삐비”), (“마춤삐비 틀렸” → “마춤삐비 틀려써”), (“마춤삐비 틀려쌩” → “마춤삐비 틀려써요”)

문자열이 주어지면 그 문자열을 키보드로 입력했을 때 도깨비불 현상의 횟수를 계산해보자.

입력

첫 번째 줄에 공백 문자()와 완성형 한글들로 이루어진 한글 문자열을 영문 입력 모드에서 타이핑했을 때 입력 되는 문자열이 주어진다. 문자열의 길이는 1 이상 10^4 이하이다.

출력

입력으로 주어진 문자열을 한글 입력 모드로 타이핑했을 때 발생하는 도깨비불 현상의 횟수를 출력하라.

예제 1

입력	출력
dlqhtpdy	3

“이보세요”이다.

예제 2

입력	출력
zheldgkrl tlfgek	2

“코딩하기 싫다”이다.

예제 3

입력	출력
akcnaQjql xmffuTjdy	4

“마춤빠비 틀려써요”이다.

노트

초성	ㄱ ㅋ ㄴ ㄷ ㄸ ㄹ ㅁ ㅂ ㅃ ㅅ ㅆ ㅈ ㅊ ㅌ ㅍ ㅑ ㅒ ㅓ ㅔ ㅕ ㅖ ㅗ ㅛ ㅜ ㅠ ㅎ
영문 입력	r R s e E f a q Q t T d w W c z x v g
중성	ㅏ ㅙ ㅛ ㅝ ㅑ ㅓ ㅕ ㅗ ㅛ ㅜ ㅝ ㅑ ㅓ ㅕ ㅗ ㅛ ㅜ ㅝ ㅑ ㅓ ㅕ ㅗ ㅛ ㅜ ㅝ ㅑ ㅓ ㅕ ㅗ ㅛ ㅜ ㅝ
영문 입력	k o i O j p u P h hk ho hl y n nj np nl b m
중성	ㅡ ㅣ
영문 입력	m l ㄴ
중성	ㄱ ㅋ ㄴ ㄷ ㄸ ㄹ ㅁ ㅂ ㅃ ㅅ ㅆ ㅈ ㅊ ㅌ ㅍ ㅑ ㅒ ㅓ ㅔ ㅕ ㅖ ㅗ ㅛ ㅜ ㅠ ㅎ
영문 입력	r R rt s sw sg e f fr fa fq ft fx fv fg a q qt t
중성	ㅆ ㅑ ㅒ ㅓ ㅔ ㅕ ㅖ ㅗ ㅛ ㅜ ㅠ ㅎ
영문 입력	T d w c z x v g

입력 자판은 앞서 첨부한 두벌식 자판 이미지를 참고한다. (한글 자판은 두벌식, 영문 자판은 QWERTY이다.)

쌍자음 ㄴ, ㄸ, ㅃ, ㅆ, ㅌ은 각각 R, E, Q, T, W에 대응되며, rr, ee, qq, tt, ww와 같은 방법으로는 입력할 수 없다.

초성은 19개, 중성은 21개, 종성은 27개가 있다.

C. 블랙 프라이데이

시간 제한: 1초

서강 백화점이 블랙 프라이데이를 맞아 특별 이벤트를 진행한다. 백화점에서 제시하는 양의 정수의 무게 C 에 딱 맞게 물건들을 가져오면 전부 만 원에 판매하는 이벤트이다.

선택할 수 있는 물건은 최대 3개까지이고, 같은 물건을 중복 선택하는 것은 불가능하다. 그리고 백화점에서 판매하는 물건들의 무게는 모두 다르다.

예를 들어, 백화점에서 판매하고 있는 물건 5개의 무게가 각각 1, 2, 3, 4, 5일 때, C 가 5라면 $\{2, 3\}$ 또는 $\{5\}$ 에 해당하는 물건의 조합을 만 원에 구매할 수 있다.

판매하는 물건 N 개의 양의 정수의 무게가 각각 주어질 때, 만 원에 구매할 수 있는 조합이 있는지 출력하라.

입력

첫 번째 줄에 물건의 개수 N 과 제시하는 무게 C 가 공백으로 구분되어 주어진다. ($1 \leq N \leq 5,000, 1 \leq C \leq 10^8, N$ 과 C 는 양의 정수)

다음 줄에는 N 개의 물건 각각의 무게 w 가 공백으로 구분되어 주어진다. ($1 \leq w \leq 10^8, w$ 는 양의 정수)

출력

문제의 조건을 만족하는 조합이 있으면 1, 그렇지 않으면 0을 출력한다.

예제 1

입력	출력
5 5 1 2 3 4 5	1

예제 2

입력	출력
3 13 3 7 8	0

D. 이진수 게임

시간 제한: 1초

이진수 게임은 주어진 ‘시작 이진수’를 몇 가지 동작으로 ‘목표 이진수’로 바꾸는 게임이다.

이 게임에서 가능한 동작들은 다음과 같다.

1. 한 자리 숫자를 보수로 바꾸기. 단, 맨 앞 숫자(Most Significant Digit)는 바꿀 수 없다.

$$101_2 \rightarrow 111_2$$

2. 현재 수에 1 더하기.

$$11_2 \rightarrow 100_2$$

3. 현재 수에서 1 빼기. 단, 현재 수가 0이라면 빼기가 불가능하다.

$$110_2 \rightarrow 101_2$$

‘시작 이진수’와 ‘목표 이진수’가 주어질 때, ‘시작 이진수’를 ‘목표 이진수’로 만들기 위한 최소 동작 횟수를 출력하라. 주어지는 이진수들의 맨 앞 숫자는 항상 1이다.

입력

첫 번째 줄에 길이 L 의 ‘시작 이진수’가 주어진다. 두 번째 줄에 길이 K 의 ‘목표 이진수’가 주어진다. ($1 \leq L, K \leq 10$)

출력

‘시작 이진수’를 ‘목표 이진수’로 만들기 위한 최소 동작 횟수를 출력한다.

예제 1

입력	출력
110 1000	2

예제 2

입력	출력
101 111	1

E. 그르다 김가눔

시간 제한: 1.5초

정래는 김밥가게 “그르다 김가눔”에 납품할 김밥을 만드는 김밥 공장을 운영한다. 정래는 김밥 양쪽 끝을 “꼬다리”라고 부른다. 그리고 꼬다리를 잘라낸 김밥을 “손질된 김밥”이라고 부른다.

공장에서는 김밥 N 개에 대해서, 김밥 꼬다리를 잘라내고 손질된 김밥을 김밥조각으로 만드는 작업을 한다. 꼬다리를 잘라낼 때에는 양쪽에서 균일하게 K cm만큼 잘라낸다. 만약 김밥의 길이가 $2K$ cm보다 짧아서 한쪽밖에 자르지 못한다면, 한쪽만 꼬다리를 잘라낸다. 김밥 길이가 K cm이거나 그보다 짧으면 그 김밥은 폐기한다.

손질된 김밥들은 모두 일정한 길이 P 로 잘라서 P cm의 김밥조각들로 만든다. P 는 양의 정수여야 한다. 정래는 일정한 길이 P cm로 자른 김밥조각을 최소 M 개 만들고 싶다. P 를 최대한 길게 하고 싶을 때, P 는 얼마로 설정해야 하는지 구하시오.

입력

첫 번째 줄에 손질해야 하는 김밥의 개수 N , 꼬다리의 길이 K , 김밥조각의 최소 개수 M 이 주어진다. ($1 \leq N \leq 10^6$, $1 \leq K, M \leq 10^9$, N, K, M 은 정수)

두 번째 줄부터 김밥의 길이 L 이 N 개 주어진다. ($1 \leq L \leq 10^9$, L 은 정수)

출력

김밥조각의 길이 P 를 최대로 할 때, P 를 출력한다. 만족하는 P 가 없는 경우, -1 을 출력한다.

예제 1

입력	출력
3 6 4 20 10 3	2

예제 2

입력	출력
3 8 1 16 7 8	-1

F. 7-세그먼트 디스플레이

시간 제한: 3초

7-세그먼트 디스플레이란 아래와 같이 7개의 선분으로 글자를 표시할 수 있는 장치를 말한다.

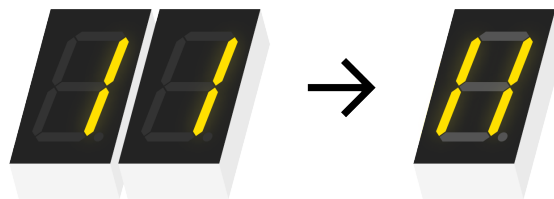


최근 시프트는 디지털 회로에 관한 강의를 듣고 있다. 시프트가 이번 주에 받은 과제의 내용은 아래와 같다.

7-세그먼트 디스플레이 n 개를 사용한 회로를 구성한다. 양의 정수 m 을 입력받아, n 개의 7-세그먼트 디스플레이에 표현할 수 있는 가장 큰 m 의 배수를 표시하시오.

다행히도 입출력은 강의자료에 전부 나와 있어서 그대로 따라 하면 됐기 때문에 별로 문제가 될 건 없었다. 더군다나, n 자리 수로 만들 수 있는 가장 큰 m 의 배수를 계산하는 건 너무 쉽다.

하지만 시프트는 놀라운 발견을 하게 되는데, 7-세그먼트 디스플레이 하나에는 0-9의 숫자뿐만 아니라 11도 표시할 수 있다는 사실이었다.



두 개의 7-세그먼트 디스플레이로 1을 표시하는 대신 한 개의 디스플레이로 11을 표시하면 한 자리를 아낄 수 있고, 이 방법으로 더 큰 m 의 배수를 표현할 수 있을지도 모른다. 예를 들어 4자리 수 중 가장 큰 3의 배수는 9999이지만, 4개의 디스플레이로 만들 수 있는 가장 큰 3의 배수는 **9 11 11 11**이다.

이 방법을 활용해, 정수 n 과 m 에 대해 시프트가 n 개의 7-세그먼트 디스플레이에 표현할 수 있는 가장 큰 m 의 배수를 출력하라.

입력

첫 번째 줄에 테스트 케이스의 개수 t 가 주어진다. ($1 \leq t \leq 25$)

각 테스트 케이스는 한 줄로 이루어져 있으며, 두 개의 수 n 과 m 이 주어진다. ($1 \leq n \leq 9, 1 \leq m \leq 10^5$)

출력

각 테스트 케이스마다 n 개의 7-세그먼트 디스플레이에 표현할 수 있는 가장 큰 m 의 배수를 한 줄에 하나씩 출력한다.

예제

입력	출력
3	9111111
4 3	1111116
4 4	1111115
4 5	

G. 단어 암기

시간 제한: 4초

준석이는 영어 단어를 외우려고 한다. 사전에는 N 가지 단어가 적혀 있다. 모든 단어는 소문자이다. 단어 안에 있는 모든 알파벳을 알 때, 그 단어를 완전히 안다고 한다.

다음과 같은 쿼리들이 주어진다.

- 1 x : 알파벳 x 를 잇는다.
- 2 x : 알파벳 x 를 기억해 낸다.

처음에 모든 알파벳을 기억하는 상태고, 모음은 완벽하게 외웠기 때문에 절대 잊지 않는다.

각 쿼리마다 완전히 알고 있는 단어의 개수를 출력하여야.

입력

첫 번째 줄에는 정수 N ($1 \leq N \leq 10^4$)과 M ($1 \leq M \leq 5 \times 10^4$)이 주어진다.

다음 N 개의 줄에는 문자열이 하나씩 주어진다. 문자열의 길이는 10^3 을 넘지 않는다.

다음 M 개의 줄에는 정수 o 와 문자 x 가 한 줄씩 주어진다. o 는 1, 2 중 하나이고, x 는 알파벳 소문자이다.

o 가 1이면 x 를 잇는다는 뜻이고, o 가 2면 x 를 기억해낸다는 뜻이다. o 가 1일 때는 x 를 기억하고 있었음이 보장되고, o 가 2일 때는 x 를 잇고 있었음이 보장된다.

출력

각 쿼리마다 정수 하나를 출력한다.

예제

입력	출력
5 10	3
apple	1
actual	0
banana	0
brick	1
courts	1
1 l	1
1 b	3
1 c	4
1 n	5
2 l	
2 b	
1 s	
2 c	
2 s	
2 n	

H. 색깔 하노이 탑

시간 제한: 1초

하노이 탑 게임은 3개의 기둥과 여러 장의 크기가 모두 다른 원판을 이용한 게임이다. 처음에는 1번째 기둥에 크기가 큰 원판이 밑에 오도록 모든 원판이 크기 순서대로 놓여 있고, 다음 규칙을 지키면서 모든 원판을 3번째 기둥으로 옮겨야 한다.

- 작은 원판 위에 큰 원판이 올라올 수 없다.
- 한 번에 1개의 원판만 옮길 수 있다.

이 규칙을 토대로 원판의 색깔이 두 개인 하노이 탑 게임을 만들려고 한다.



처음에는 1번째 기둥에 크기 1인 빨간 원판, 크기 1인 검은 원판, 크기 2인 빨간 원판, 크기 2인 검은 원판, ..., 크기 N 인 빨간 원판, 크기 N 인 검은 원판이 위에서부터 차례대로 놓여 있다.

위의 규칙을 지키면서 원판을 움직여 3번째 기둥에 크기 1인 빨간 원판, 크기 1인 검은 원판, 크기 2인 빨간 원판, 크기 2인 검은 원판, ..., 크기 N 인 빨간 원판, 크기 N 인 검은 원판이 위에서부터 차례대로 오도록 놓아야 한다면, 원판을 최소로 이동시킬 때 총 몇 번 이동해야 할까?

입력

첫 번째 줄에 정수 N ($1 \leq N \leq 10^6$)이 주어진다.

출력

첫 번째 줄에 원판의 최소 이동 횟수를 출력하여야 한다. 수가 커질 수도 있으니, 최소 이동 횟수를 $10^9 + 7$ 로 나눈 나머지를 출력하여야 한다.

예제

입력	출력
1	3

노트

원판의 크기가 같다면, 색과 상관없이 어떤 원판을 위에 올려놓아도 무방하다. 단, 최종 상태는 위에서부터 빨강 원판 - 검정 원판 - 빨강 원판 - 검정 원판 - ... 순이어야 한다.