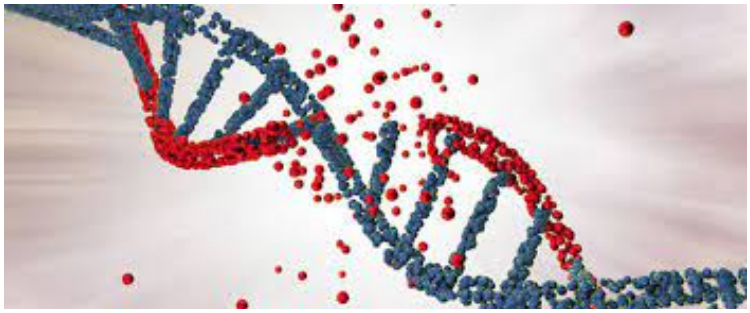

GDBC[1)]

[문제] 인간의 유전자는 대략 2.5만개 정도로 추정되고 있다. 인간의 모든 생리적 특성은 이 유전자들의 상호 작용에 의해서 결정된다. 따라서 질병도 특정 유전자의 유무에 따라서 결정되는데 유전인자의 유무, 그리고 그 발현 정도에 따라서 나타나는 질병을 보통 유전성 질환이라고 부른다. 다운증후군, 백색증, 난포성 섬유종 등은 모두 유전성 질환이며 그 중 대표적인 것은 BRCA 유전자의 변형으로 인한 유방암이다. 그러나 DNA 서열에 암유전자가 포함되어 있다고 해서 암에 걸리는 것은 아니며 그 유전자가 발현(express)될 때 발생한다.[2)]

이를 위하여 유전자 빅데이터 센터(GDBC)에서는 세계 모든 병원의 의료진과 과학자들로부터 자신이 취득한 유전성 질환 정보를 보고받아 유전자에 따른 발현 질병에 관한 정보를 수집하여, 검색 요청을 받은 특정 유전자 집합에 어떤 유전성 질환이 연관되어 있는지를 최대한 빠르게 처리하는 서비스를 구축하려고 한다. 여러분은 이를 도와주는 검색 프로그램을 구현해야 한다. 따라서 단순하게 STL vector나 배열과 같은 iteratable container가 아닌 map이나 python의 dict와 같은 associative memory를 활용해야 한다.

GDBC은 두 가지 동작을 지원한다. 하나는 연구자들의 실험 결과를 등록하는 작업으로 특정 유전자와 연관된 질환에 관한 내용을 등록(Registration)한다. 다른 하나는 관심 유전자와 관련된 질병이 어떤 것인지를 선행 연구자들이 이미 등록한 결과를 찾아보는 질의 모드(Query Mode)이다. 등록(Registration)의 한 예는 위와 같다.

```
> R  34  561 123  87 -34
```

첫 문자 R은 Registration Mode를 의미한다. 이어 같은 줄에 나타나는 양의 정수는 유전자 id를 나타낸다. 그리고 마지막에 나타난 음수는 해당 유전자를 가진 사람에게 나타난 질병의 코드값 GD

1) Genetic Diseases Bigdata Center(GDBC). 이 과제는 2023년 자료구조의 마지막 과제물입니다. 끝까지 최선을 다한 모든 수강생들에게 감사 인사를 전합니다. 정말 모두 고생했습니다.
2) 대부분의 경우 흡연, 음주, 마약류 등을 동반한 방탕한 생활을 하면 암관련 유전자가 "켜(on)"지게 되며, 화학물질이나 특히 방사성 관련 물질은 암 유전자를 깨우는 가장 대표적이며 치명적인 물질이다.

CSED를 의미한다. 각 R 모드에서 관련 유전자는 1개 이상이며 질병은 1개만 표시된다.

그리고 질의 모드(query Mode)의 형식과 예는 다음과 같다.

```
> Q  34  561 123  87 0
```

첫 문자(character)는 Q이며 이어서 관찰된 유전자 ID의 집합이 나타나고 그 끝은 숫자 zero(0)으로 표시된다. GDBC의 서버에서는 Q mode에서 질의에 입력된 유전자 집합과 동일한 집합의 유전자에서 발현된 모든 질병을 찾아서 출력한다. 즉, 여러분은 질의 유전자 집합 { g_i }와 연관되어 있다고 이미 보고된 모든 유전 질병을 찾아 내림차순으로 stdout에 한 줄로 출력해야 한다.

만일 질의한 유전자 집합과 관련된 질병이 GDBC 서버에 보고가 된 적이 없을 경우에는 "None"이라고 출력해야 한다. 즉 여러분은 Query mode 만큼의 올바른 결과를 출력해야 한다.

단, 유전자와 해당 증상(syndrome)은 정확하게 일치하는 경우에만 검색된다. 만일 유전자가 `{1,2,3}`일때 -10인 질병이 존재한다는 보고만 GDBC에 등록되었다고 가정하자. 이 경우 Query로 들어온 어떤 사람의 유전자 집합이 `{1,2,3,4}`라면 -10과는 전혀 다른 생리적 현상이 나타날 수 있으므로 "None"이 출력된다. 왜냐하면 새로 추가된 유전자 4가 다른 유전자 `{1,2,3}`을 조절(enhancing 또는 suppressing)할 수 있으므로 앞서의 '-10'에 해당되는 병리적 현상이 전혀 나타나지 않을 수 있다.

[입출력] 입출력은 stdout과 stdin을 사용한다. 입력의 끝은 문자 `'$'`로 표시된다. 즉 해당 줄(line)의 첫 문자가 `'$'`이면 작업을 종료해야 한다. 특정 유전자 집합에 대하여 하나 이상의 질병이 보고될 수 있기 때문에 질의 이전에 보고된 모든 관련된 질병 코드를 내림차순으로 출력해야 한다. $를 제외한 R과 Q모드를 합한 입력 파일의 전체 줄의 수는 **최대 100,000(십만)**이다. 유전자로 표시된 번호는 100,000 이하의 숫자이며 질병 코드 GD의 범위는 $-100 \le GD \le -1$ 의 정수이다.

[예제]

stdin		stdout	
R 1 2 3 -10		-10 -77	//질의 1
R 1 5 6 9 -45		None	//질의 2
R 3 2 1 -77			
R 55 66 77 -3			
Q 3 2 1 0	//질의 1		
R 1 2 3 6 -45			
Q 6 3 2 0	//질의 2		
$	// end marker		

stdin		stdout	
R 1 2 3 -10		-10	//질의 1
R 1 5 6 9 -45		None	//질의 2
Q 3 2 1 0	//질의 1	-3 -9	//질의 3
R 55 66 77 -9			
R 3 2 1 -77			
R 55 77 66 -3			
R 1 2 3 6 -45			
Q 77 66 0	//질의 2		
Q 77 55 66 0	//질의 3		
$	// end marker		

[조건] 프로그램의 이름은 **GDBC.{c, cpp, java, py}**, 제출 횟수는 30회, 수행 제한시간은 1초, 허용되는 Token의 수는 500이다. 이 문제는 반드시 **STL map** 또는 Python의 **dict, hash, set** 등을 활용해서 해결해야 한다. map의 key에 vector나 set이 활용될 수 있음을 이해하는데 도움이 되는 중요한 과제이다. 제한 사항을 지키지 않으면 점수는 50%로 처리된다.

Python은 set을 dict의 key로 사용하지 못하므로, 이를 대신할 수 있는 "냉동집합"-frozen_set이나 tuple을 사용해야 한다.