

## 1번

XX쇼핑센터는 다음과 같이 회원 등급제를 운영하고 있습니다.

등급	기준
브론즈	최근 30일간 0원 이상 10,000원 미만 구매 고객
실버	최근 30일간 10,000원 이상 20,000원 미만 구매 고객
골드	최근 30일간 20,000원 이상 50,000원 미만 구매 고객
플래티넘	최근 30일간 50,000원 이상 100,000원 미만 구매 고객
다이아몬드	최근 30일간 100,000원 이상 구매 고객
등급은 브론즈 → 실버 → 골드 → 플래티넘 → 다이아몬드 순으로 높아집니다.	

고객의 2019년 1월 1일 ~ 2019년 12월 31일 기간 동안 구매 기록이 문자열 형태로 담긴 배열 purchase가 매개변수로 주어질 때, 각 등급별 유지 기간을 순서대로 배열에 담아 return 하도록 solution 함수를 완성해주세요.

## 1번

### 제한사항

purchase의 길이는 1 이상 365 이하입니다.

purchase의 원소는 2019/MM/DD X 형식입니다.

2019/MM/DD는 2019년 MM월 DD일을 의미하며, 2019/01/01에서 2019/12/31사이입니다.

구매 기록은 날짜 순으로 정렬되어 있습니다.

같은 날짜의 구매 기록이 중복해서 주어지지 않습니다.

'X'는 구매 금액을 나타내며 1,000 이상 200,000 이하인 자연수입니다.

구매 금액은 1,000원 단위로만 주어집니다.

정답 배열은 [브론즈 기간, 실버 기간, 골드 기간, 플래티넘 기간, 다이아몬드 기간] 순서로 채워서 return 해주세요.

purchase

["2019/01/01 5000", "2019/01/20 15000", "2019/02/09 90000"]

["2019/01/30 5000", "2019/04/05 10000", "2019/06/10 20000", "2019/08/15 50000", "2019/12/01 100000"]

result

[315, 9, 11, 20, 10]

[245, 30, 30, 30, 30]

## 2번

사이버 코딩 대학(Cyber Coding University, 이하 CCU)에서는 학생들을 대상으로 온라인 코딩 경진대회를 실시하였습니다. 심사위원 자격으로 학생들의 접속 로그를 살펴본 당신은, 같은 IP 주소가 여러 학생들에게 중복되는 것을 확인하였습니다. 하지만 같은 IP주소가 반복되었다고 해서, 무조건 부정 행위가 있었다고 단정할 수는 없습니다. 같은 방을 쓰는 학생들이 컴퓨터 한 대로 차례차례 경진대회에 응시했다면, 같은 IP를 사용했어도 부정 행위는 아니기 때문입니다. 따라서 당신은, 아래와 같은 3가지 기준을 만들어서 부정 참가자를 판별하기로 했습니다.

- [조건 1] 같은 IP주소로 접속한 학생이 4명 이상이면, 해당 IP로 접속한 학생은 모두 부정 참가자 입니다.
- [조건 2] 같은 IP주소로 접속한 학생이 3명이고, 3명이 모두 같은 언어군을 사용했다면 3명은 부정 참가자 입니다.
- [조건 3] 같은 IP주소로 접속한 학생이 2명이고, 2명이 같은 언어군을 사용했고, 성적도 동일하다면 2명은 부정 참가자 입니다.

경진대회에서 학생들이 사용할 수 있는 언어는 C, C++, C#, Java, JavaScript, Python3 6가지 입니다. 이 중에서, C, C++, C#은 서로같은 언어군으로 분류합니다. Java, JavaScript, Python3은 자기 자신을 제외하면 같은 언어군인 언어는 없습니다.

학생들의 IP주소를 담은 문자열 배열 `ip_addrs`, 사용 언어를 담은 문자열 배열 `langs`, 시험 점수를 담은 정수형 배열 `scores`가 매개변수로 주어집니다. 이 때, 전체 학생들 중에서 부정 참가자를 제외한 정당한 참가자의 수를 구해서 return 하도록 `solution` 함수를 완성해주세요.

## 2번

### 제한사항

ip\_addrs의 길이는 1 이상 100,000 이하입니다.

ip\_addrs의 원소는 숫자와 .로 이루어진 문자열로, IPv4 형식을 준수합니다.

IPv4 형식

문자열 형태로, 0부터 255까지의 숫자 4개를 .을 이용해 구분해서 표현합니다.

ex) 0.0.0.0 ~ 255.255.255.255

0을 제외한 다른 숫자는 0으로 시작하지 않습니다.

0은 0으로만 표시합니다. 즉 00과 같이 표시하지 않습니다.

ex) 127.0.0.1

1번 학생의 IP주소부터 차례대로 주어집니다.

langs의 길이는 id\_addrs의 길이와 동일합니다.

langs의 원소는 "C", "C++", "C#", "Java", "JavaScript", "Python3" 6가지 중 하나입니다.

1번 학생이 사용한 언어부터 차례대로 주어집니다.

scores의 길이는 id\_addrs의 길이와 동일합니다.

scores의 원소는 0이상 400 이하인 정수입니다.

1번 학생의 성적부터 차례대로 주어집니다.

ip_addrs	langs	scores	result
["5.5.5.5", "155.123.124.111", "10.16.125.0", "155.123.124.111", "5.5.5.5", "155.123.124.111", "10.16.125.0", "10.16.125.0"]	["Java", "C++", "Python3", "C#", "Java", "C", "Python3", "JavaScript"]		
["7.124.10.0", "7.124.10.0", "0.0.0.0", "7.124.10.0", "0.0.0.0", "7.124.10.0"]	["C++", "Java", "C#", "C#", "C", "Python3"]	[314, 225, 45, 0, 155, 400]	2

[294, 197, 373, 45, 294, 62, 373, 373] 3

### 3번

모든 아이디는 S+N 형식입니다.  
S는 알파벳 소문자(a ~ z)로 구성된 문자열로 길이는 3 이상 6 이하입니다.  
N는 숫자(0~9)로 구성된 문자열로 길이는 0 이상 6 이하입니다.  
N의 길이가 0이 될 수도 있다는 것은, N이 비어있는 널(null) 문자열이 될 수도 있다는 의미입니다.  
N의 길이가 1 이상이면, N의 첫자리는 0이 될 수 없습니다.  
즉, 034 , 06, 0, 09040, 000과 같은 문자열은 N이 될 수 없습니다.  
위의 규칙에 부합하는, 올바른 아이디의 예를 들어보면 다음과 같습니다.  
"ace", "hahaa512", "sunfri1", "aaaaaa900000", "abcde10101"

위의 규칙에 위배되는, 잘못된 아이디의 예를 들어보면 다음과 같습니다.

아이디	부적합 사유
"ac"	S의 길이가 3 미만입니다.
"Ange1004"	S에 알파벳 소문자가 아닌 문자가 포함되어 있습니다.
"1004angel"	S+N 형식이 아닙니다.(N+S형식은 허용하지 않습니다.)
"aaaaa1aaaaa"	S+N 형식이 아닙니다.(S+N+S 형식은 허용하지 않습니다.)
"triger0145"	N의 첫자리가 0이 될 수 없습니다.
"abcdefg733"	S의 길이가 6을 초과합니다.
"zzzzz4954951"	N의 길이가 6을 초과합니다.

이미 홈페이지에 가입된 아이디들의 목록(registered\_list)과, 신규회원이 사용하기를 원하는 아이디(new\_id)가 있다면, 아래와 같은 방법으로 신규회원에게 아이디를 추천 해주려고 합니다.

new\_id가 registered\_list에 포함되어 있지 않다면, new\_id를 추천하고 종료합니다.  
new\_id가 registered\_list에 포함되어 있다면,

- 2-1. new\_id를 두 개의 문자열 S와 N으로 분리합니다.
- 2-2. 문자열 N을 10진수 숫자로 변환한 값을 n이라고 합니다.(단, N이 비어있는 null 문자열이라면, n은 0이 됩니다.)
- 2-3. n에 1을 더한 값(n+1)을 문자열로 변환한 값을 N1라고 합니다.
- 2-4. new\_id를 S+N1로 변경하고 1.로 돌아갑니다.

이미 가입된 아이디들의 목록 registered\_list와 신규회원이 사용하기를 원하는 아이디 new\_id가 매개변수로 주어집니다. 이때 설명한 방법을 적용했을 때, 신규회원에게 추천되는 아이디를 return 하도록 solution 함수를 완성해주세요.

3번

제한사항  
new\_id는 올바른 S+N 형식의 아이디(문자열)입니다.  
registered\_list는 길이가 1 이상 100,000 이하인 문자열 배열입니다.  
registered\_list의 각 원소는 올바른 S+N 형식의 아이디(문자열)입니다.  
registered\_list에 중복된 원소는 없습니다.(즉, 모두 서로 다른 아이디입니다.)  
문제에서 설명된 방법을 적용하여 return 되는 추천 아이디는 항상 올바른 S+N 형식임이 보장됩니다.  
즉, 추천 아이디를 결정하는 방법을 수행하면서 N 부분의 자릿수가 6을 초과하는 경우의 입력은 주어지지 않습니다.

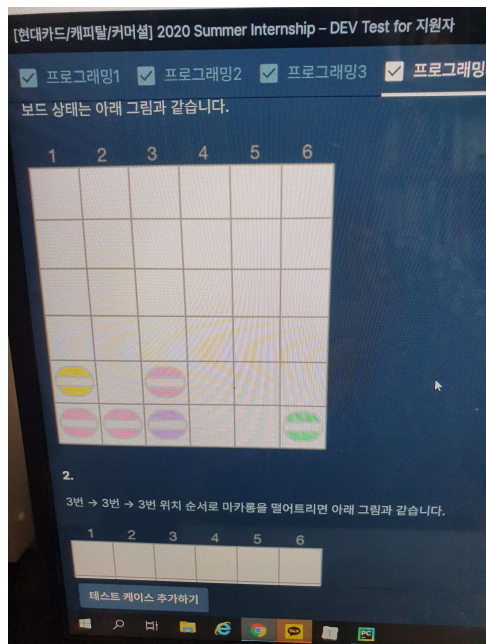
registered_list	new_id	result
["card", "ace13", "ace16", "banker", "ace17", "ace14"]	"ace15"	"ace15"
["cow", "cow1", "cow2", "cow3", "cow4", "cow9", "cow8", "cow7", "cow6", "cow5"]	"cow"	"cow10"
["bird99", "bird98", "bird101", "gotoxy"]	"bird98"	"bird100"
["apple1", "orange", "banana3"]	"apple"	"apple"

## 4번

마카롱 게임은 6 x 6 크기 게임 보드에 위에서 아래 방향으로 마카롱을 하나씩 떨어뜨려 같은 색 마카롱이 상, 하, 좌, 우 방향으로 3개 이상 연결되면 터지는 게임입니다. 마카롱은 항상 빈칸 없이 가장 아래부터 차곡차곡 쌓입니다. 또, 마카롱이 터져서 없어지면 위 칸의 마카롱이 차례대로 아래 칸으로 떨어집니다. 만약, 떨어진 마카롱이 다시 3개 이상 연결된다면 연결된 마카롱도 연쇄적으로 터지며, 더 터지는 마카롱이 없을 때까지 게임 보드 위의 마카롱이 아래 방향으로 떨어집니다. 단, 현재 게임보드에 3개 이상 연결된 마카롱이 여러 개라면 한꺼번에 터진다고 가정합니다.

아래는 게임이 진행되는 예시입니다. 처음에 게임 보드 각 칸은 전부 빈 상태입니다.

1.  
1번 → 2번 → 1번 → 3번 → 6번 → 3번 위치 순서로 각 색상의 마카롱을 떨어트린 후 보드 상태는 아래 그림과 같습니다.



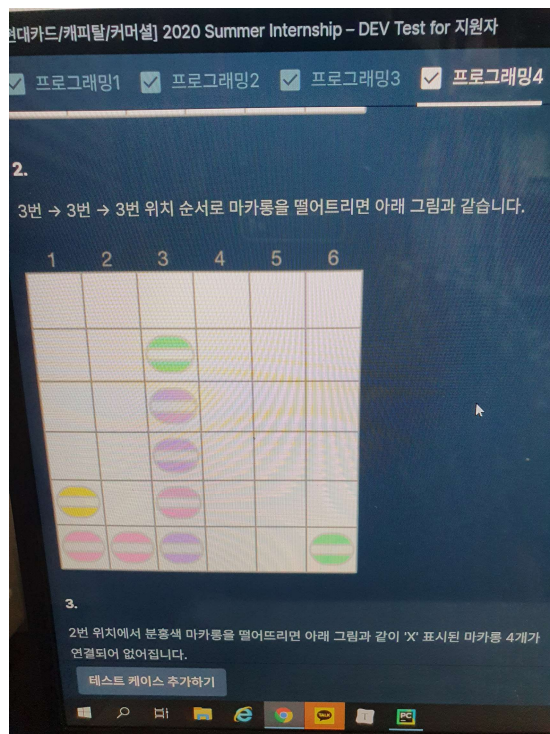
## 4번

마카롱 게임은 6 x 6 크기 게임 보드에 위에서 아래 방향으로 마카롱을 하나씩 떨어뜨려 같은 색 마카롱이 상, 하, 좌, 우 방향으로 3개 이상 연결되면 터지는 게임입니다. 마카롱은 항상 빈칸 없이 가장 아래부터 차곡차곡 쌓입니다. 또, 마카롱이 터져서 없어지면 위 칸의 마카롱이 차례대로 아래 칸으로 떨어집니다. 만약, 떨어진 마카롱이 다시 3개 이상 연결된다면 연결된 마카롱도 연쇄적으로 터지며, 더 터지는 마카롱이 없을 때까지 게임 보드 위의 마카롱이 아래 방향으로 떨어집니다. 단, 현재 게임보드에 3개 이상 연결된 마카롱이 여러 개라면 한꺼번에 터진다고 가정합니다.

아래는 게임이 진행되는 예시입니다. 처음에 게임 보드 각 칸은 전부 빈 상태입니다.

2.

3번 → 3번 → 3번 위치 순서로 마카롱을 떨어트리면 아래 그림과 같습니다.



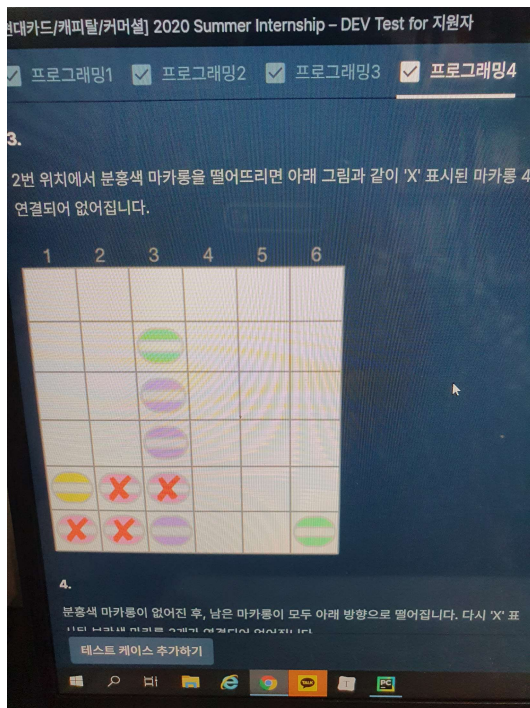


## 4번

마카롱 게임은 6 x 6 크기 게임 보드에 위에서 아래 방향으로 마카롱을 하나씩 떨어뜨려 같은 색 마카롱이 상, 하, 좌, 우 방향으로 3개 이상 연결되면 터지는 게임입니다. 마카롱은 항상 빈칸 없이 가장 아래부터 차곡차곡 쌓입니다. 또, 마카롱이 터져서 없어지면 위 칸의 마카롱이 차례대로 아래 칸으로 떨어집니다. 만약, 떨어진 마카롱이 다시 3개 이상 연결된다면 연결된 마카롱도 연쇄적으로 터지며, 더 터지는 마카롱이 없을 때까지 게임 보드 위의 마카롱이 아래 방향으로 떨어집니다. 단, 현재 게임보드에 3개 이상 연결된 마카롱이 여러 개라면 한꺼번에 터진다고 가정합니다.

아래는 게임이 진행되는 예시입니다. 처음에 게임 보드 각 칸은 전부 빈 상태입니다.

- 3.
- 2번 위치에서 분홍색 마카롱을 떨어뜨리면 아래 그림과 같이 'X' 표시된 마카롱 4개가 연결되어 없어집니다.

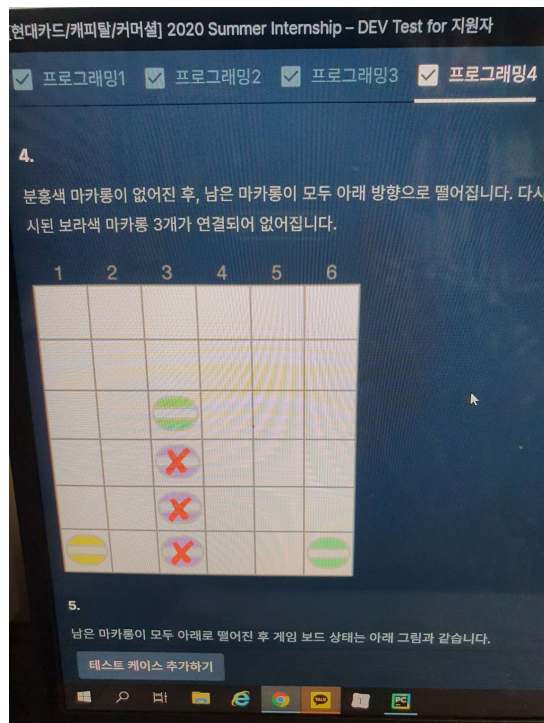


## 4번

마카롱 게임은 6 x 6 크기 게임 보드에 위에서 아래 방향으로 마카롱을 하나씩 떨어뜨려 같은 색 마카롱이 상, 하, 좌, 우 방향으로 3개 이상 연결되면 터지는 게임입니다. 마카롱은 항상 빈칸 없이 가장 아래부터 차곡차곡 쌓입니다. 또, 마카롱이 터져서 없어지면 위 칸의 마카롱이 차례대로 아래 칸으로 떨어집니다. 만약, 떨어진 마카롱이 다시 3개 이상 연결된다면 연결된 마카롱도 연쇄적으로 터지며, 더 터지는 마카롱이 없을 때까지 게임 보드 위의 마카롱이 아래 방향으로 떨어집니다. 단, 현재 게임보드에 3개 이상 연결된 마카롱이 여러 개라면 한꺼번에 터진다고 가정합니다.

아래는 게임이 진행되는 예시입니다. 처음에 게임 보드 각 칸은 전부 빈 상태입니다.

4. 분홍색 마카롱이 없어진 후, 남은 마카롱이 모두 아래 방향으로 떨어집니다. 다시 'X' 표시된 보라색 마카롱 3개가 연결되어 없어집니다.

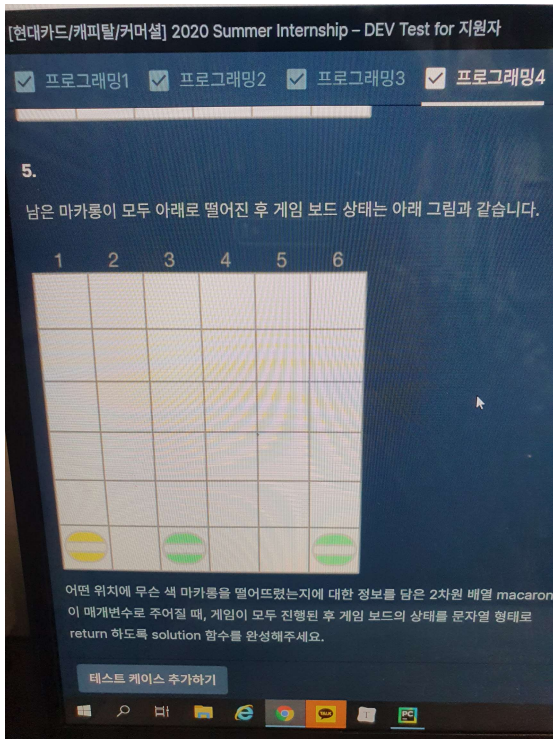


4번

마카롱 게임은 6 x 6 크기 게임 보드에 위에서 아래 방향으로 마카롱을 하나씩 떨어뜨려 같은 색 마카롱이 상, 하, 좌, 우 방향으로 3개 이상 연결되면 터지는 게임입니다. 마카롱은 항상 빈칸 없이 가장 아래부터 차곡차곡 쌓입니다. 또, 마카롱이 터져서 없어지면 위 칸의 마카롱이 차례대로 아래 칸으로 떨어집니다. 만약, 떨어진 마카롱이 다시 3개 이상 연결된다면 연결된 마카롱도 연쇄적으로 터지며, 더 터지는 마카롱이 없을 때까지 게임 보드 위의 마카롱이 아래 방향으로 떨어집니다. 단, 현재 게임보드에 3개 이상 연결된 마카롱이 여러 개라면 한꺼번에 터진다고 가정합니다.

아래는 게임이 진행되는 예시입니다. 처음에 게임 보드 각 칸은 전부 빈 상태입니다.

5. 남은 마카롱이 모두 아래로 떨어진 후 게임 보드 상태는 아래 그림과 같습니다.



## 4번

어떤 위치에 무슨 색 마카롱을 떨어뜨렸는지에 대한 정보를 담은 2차원 배열 macaron이 매개변수로 주어질 때, 게임이 모두 진행된 후 게임 보드의 상태를 문자열 형태로 return 하도록 solution 함수를 완성해주세요.

### 제한사항

macaron의 세로(행) 길이는 1 이상 100 이하이며, 가로(열) 길이는 2입니다.

macaron의 각 행에는 마카롱을 떨어뜨린 정보가 [떨어뜨린 위치, 마카롱의 색] 형태로 담겨있습니다.

마카롱을 떨어뜨린 위치는 1 이상 6 이하인 자연수입니다.

마카롱의 색은 1 이상 9 이하인 자연수 형태로 표현하며, 같은 색상의 마카롱은 같은 숫자로 표현합니다.

마카롱이 보드 밖으로 나가도록 하거나, 마카롱이 최대 높이까지 쌓인 줄에 다시 마카롱을 떨어뜨리는 경우는 없습니다.

정답은 길이가 6인 문자열 6개를 return 해주세요.

return 하는 배열의 첫 번째 원소는 게임 보드의 가장 윗 열을 나타내며, 이후 가장 아래 열까지 순서대로 놓으면 됩니다.

게임 보드에서 빈칸은 0, 마카롱이 채워진 칸은 해당 마카롱의 색을 나타내는 자연수를 채우면 됩니다.

[[1,1],[2,1],[1,2],[3,3],[6,4],[3,1],[3,3],[3,3],[3,4],[2,1]] [000000,000000,000000,000000,000000,204004]

[[1,1],[1,2],[1,4],[2,1],[2,2],[2,3],[3,4],[3,1],[3,2],[3,3],[3,4],[4,4],[4,3],[5,4],[6,1]] [000000,000000,000000,000000,000000,404001]