**[ 1 ] 전공 : 소프트웨어학과**

**1.**

주어진 입력값에 포함된 방들 중에서 가장 저렴한 방의 인덱스를 출력하는 문제

(단, 최저 금액이 같은 방의 경우 평수가 작은 방의 인덱스를 출력한다.)

방의 금액은 다음과 같이 계산한다.

방의 금액 = price \* 평수 + 수수료

(price: 10만 이하의 양의 정수, 평수: 1만 이하의 양의 정수, 수수료: 1억 이하의 양의 정수)

**예시)**

**Input**

price = 100, 주어진 방의 정보 = [[10, 100000], [20, 300000], [40, 200000]]

1번 방: 100 \* 10 + 100000 = 101000

2번 방: 100 \* 20 + 300000 = 302000

3번 방: 100 \* 40 + 200000 = 204000

**Output**

1

**2.**

A 회사에는 여러 회의실이 존재한다. 이 때, 각 회의실마다 놓여있는 좌석에는 직원에게 부여된 지정 좌석이 있다. 또한, A 회사에서 종사하는 직원은 최소 1개 이상의 지정 좌석을 갖고 있다.

어느 날, 어떤 회의실에 새로운 좌석이 생기게 되었는데 이 새로운 좌석에도 특정 직원에 대한 좌석으로 지정하려고 한다.

단, 이 때 새로운 좌석이 생긴 회의실 공간에 이미 지정 좌석을 보유하고 있는 직원은 제외하며 새로운 좌석에 대한 우선순위는 다음과 같다.

1. A 회사 내에서 **가장 적은 지정 좌석 개수**를 보유하고 있는 직원을 우선시한다.
2. 보유하고 있는 지정 좌석의 개수가 같은 직원들에 대해서는 새로 좌석이 생긴 회의실과 **가장 가까운 회의실에 지정 좌석을 갖고 있는 직원**을 우선시한다.

이 때, 회의실 간의 거리는 회의실 번호의 차이이다.

1. 위의 항목도 일치할 경우, **직원 이름의 사전순**에 따라 우선순위를 부여한다.

위의 규칙에 따라 우선 순위가 높은 순서대로 직원들의 이름을 출력하는 문제

예시)

**Input** (string 배열, int)

[“[1234]Andy,John,Danny”, “[500]Andy,wow,of”, “[1000]of”], newChair: 1000

위와 같은 input을 받은 경우 다음과 같이 지정 좌석이 놓여 있다는 것을 의미한다.

1234 회의실에 Andy, John, Danny

500 회의실에 Andy, wow, of

1000 회의실에 of

**즉, 새로 좌석이 생기는 1000번 회의실에 대해 직원들의 우선순위를 구하는 과정은 다음과 같다.**

1) of는 이미 1000 회의실에 좌석이 있으므로 제외하고 시작한다.

2). Andy에게 부여된 지정 좌석은 총 2개로 가장 많기 때문에 가장 후순위의 우선순위를 부여받는다.

3). 1000 회의실과 가장 가까운 1234 회의실에 지정 좌석을 갖고 있는 John, Danny 중에서 이름 순에 의해 Danny, John 순으로 우선순위를 부여한다.

4). 마지막으로, 500회의실에 지정 좌석이 있는 wow에게 우선순위를 부여한다.

**Output (string 배열)**

[“Danny”, “John”, “wow”, “Andy”]

**3**

통장 잔고 입출금 문제

입력값은 “명령어 계좌명 숫자” 형태로 주어진다.

이 때, 주어지는 명령어는 다음과 같다.

1. 계좌 생성(CREATE)

계좌가 없는 경우, 새로운 계좌를 생성하고 숫자값에 따라 최대 한도를 설정하며 200 코드를 반환한다.

이미 계좌가 있는 경우, 403 코드를 반환한다.

1. 입금(DEPOSIT)

입력받은 계좌명이 기존에 없는 경우, 404 코드를 반환한다.

입력받은 계좌명이 있는 경우, 숫자값에 따른 입급액을 계좌에 넣고 202코드를 반환한다.

1. 출금(WITHDRAW)

입력받은 계좌명이 기존에 없는 경우, 404 코드를 반환한다.

입력받은 계좌명이 있는 경우, 숫자값에 따라 출금액을 인출하며 201코드를 반환한다.

(단, 계좌 생성시 설정된 최대 한도를 넘을 경우 출금하지 않고 405 코드를 반환한다.)

예시)

**Input (string 배열)**

[“DEPOSIT 32a 1000”, “CREATE 50a 5000”, “WITHDRAW 50a 4000”, “WITHDRAW 50a 1001”, “CREATE 100a 10000”, “DEPOST 100a 3000”]

**Output (int 배열)**

[404, 200, 201, 405, 200, 202]

**4.** sql 문제

Table에 주어진 distance 기록 중 최대 값을 출력

SELECT DISTANCE

FROM TABLE\_NAME

ORDER BY DISTANCE DESC

LIMIT 1

**[ 2 ] 전공 : 수학과**

1.

input: price=50000, info = { {10, 10000}, {20, 5000} };

ouput: 1 ({10, 10000}가 돈이 덜 들기 때문에 1)

price <= 100,000

info[0] (방 평수) <= 10,000

info[1] (예약금?) <= 100,000,000

1) price \* info[0] + info[1] 중에서 가장 돈이 덜 드는 방

2) 돈이 똑같이 덜 든다면 평수가 더 작은 방

-------------------------

2. \*변별력이 나뉘는 문제\*

input: rooms = { "[403]James", "[404]Azad,Louis,Andy", "[101]Azad,Guard" }, target = 403

output: {"Andy", "Louis", "Guard", "Azad"}

현재 방과 각 방에 있는 사람이 주어져있음.

이 때, 한 방에는 두개의 자리를 잡을 수 없음(예를 들어, 403이라는 방에 James가 자리를 두개 잡을 수 없음)

target 방이름에 자리가 하나 생겼다고 할 때, 이 때 이 방에 들어갈 수 있는 사람들을 출력해라

100 <= 방 번호 <= 9999

1) target 방에 이름이 있는 사람은 빈 자리에 앉을 수 없음

2) 지정자리수가 작은 사람이 우선순위가 높음

3) 지정자리수가 같다면 더 가까운 방에 있는 사람의 우선순위가 높음(이때, 여러개의 방을 갖고 있으면 가장 가까운 방과의 거리만 고려하면 됨)

(거리: |403 - 202| = 201)

4) 가까운 방도 같다면 이름 순(이름은 모두 다르며, 대소문자 구별. Azad와 azad는 다른 사람임)

개인 생각: 2번이 변별력 있는 문제. 1,3,4는 금방 풀지만 2번은 생각해야 할 것이 많음. 구조체를 이용해서 풀었음.

3.

input: { "DEPOSIT b4 50000", "CREATE a30 50000", "CREATE a3 10000", "WITHDRAW a30 50001" }; (명령어 계좌이름 돈)

ouput: { 404, 200, 403, 404 }

1) CREATE면

이 이름의 계좌가 이미 생성되어 있으면 403 출력

생성되어있지 않으면 생성하고 최대 마이너스 한도를 (돈 \* -1)로 설정하고 200 출력

2) DEPOSIT면

이 이름의 계좌가 이미 생성되어 있으면 입금하고 200 출력

생성되어있지 않으면 404 출력

3) WITHDRAW면

이 이름의 계좌가 이미 생성되어 있으면

- 출금했을 때 최대 마이너스 한도를 넘으면 404 출력

- 안 넘으면 출금해주고 200 출력

생성되어있지 않으면 404 출력

4. \* SQL문제

"이름, 거리(DISTANCE), 당시 시간?" 이런식으로 주어지고 거리가 가장 긴 애를 출력해라

SELECT MAX(DISTANCE) FROM GAME\_USERS -> 이게 답

개인 생각: SQL문제는 이전 기출도 그렇고 기본만 알면 간단하게 풀 수 있는 문제라고 생각.

**[ 3 ] 전공 : 컴퓨터공학**

1. A가 집을 사려고 함. 평수당 가격과 부동산 수수료를 포함하여 최저 값인 집을 구매하기로

결정.

* Price가 2000일 때, 20평의 방, 부동산 수수료가 10000이면 총 50000원 지불.
* Ex) price 2000, [[30, 10000], [15, 15000], [20, 30000]]일 때, 70000, 45000, 70000의 비용.
* 15평의 방을 사면 된다. Result = 2
* 만약 최저가의 방이 여러 개일 경우 평수가 작은 집을 구매한다.

2. 사무실을 호수로 구분. 한 사람은 한 사무실에서 1개의 자리만을 차지한다.

한 호수에 새로운 자리가 생겼을 때, 자리에 들어올 수 있는 사람을 우선순위 순으로 구하라.

조건 1) 해당 호수에 지정자리가 있을 경우 제외한다.

조건 2) 지정되어 있는 호수의 수가 동일할 경우 해당 호수에서 가장 가까운 방에

지정자리가 있을 경우 우선으로 한다.

Ex) 지정 호수가 201호일 때, 302호에 지정자리가 있는 사람의 거리는 101호 한다. (절대값)

조건 3) 그래도 동일한 사람이 있다면, 이름순으로 한다. 이름순은(A~Za~z 순으로 한다.)

그리고 Andy와 andy는 동일한 사람이 아니다.

Ex)[“[405]James”, “[404]Azad,Louis,Andy”, “[101]Azad,Guard”] target = 405

답) [“Andy”, “Louis”, “Guard”, “Azad”]

=> Azad의 경우 배정된 방이 2개 나머지는 1개이므로 Azad가 마지막순서.

=> Andy와 Louis는 405호와 거리가 1이므로 우선순위다. -> 이름순으로 Andy가 우선

=> Guard는 그 다음으로 거리가 가까움.

3. 은행 서버 시스템 만들기

Create, deposit, withdraw의 기능 구현.

* Create 계좌번호 한도를 입력 받았을 때, 계좌가 이미 존재할 경우 403 메시지
* 계좌가 존재하지 않을 경우 계좌 생성과 한도 설정하고 200 메시지
* Deposit 계좌번호 입금금액을 입력 받았을 때, 계좌가 존재하지 않을 경우 404 메시지
* 계좌가 존재할 경우 입금하고 200메시지 출력
* Withdraw 계좌번호 출금금액을 입력 받았을 때, 계좌가 존재하지 않을 경우 404 메시지
* 계좌가 있지만, 한도를 초과할 경우 출금을 하지 않고 403 메시지
* 그 외의 경우는 출금 후 200메시지

Ex) [“CREATE a333 200”, “CREATE b333 300”, “CREATE a333 100”]

출력 => [200, 200, 403]

4. 게임 최고기록 1개 출력

* Select score from table order by score desc limit 1; 로 해결함

**[ 4 ] 전공 : 컴퓨터공학**

문제1 : 여러 주택 후보중에서 평수에 상관없이 총 구매금액이 가장 낮은 주택의 번호를 return  
 입력값 : 평수당 가격, 각 주택의 평수와 수수료 2차원 배열   
 총 구매금액 = 평수 \* 평수당 가격 + 수수료  
 총 구매금액이 같다면 평수가 높은?낮은? 주택의 번호 return  
 -> 단순 구현 문제

문제2: 건물에서 한 사무실의 새로운 자리가 생겼을 때, 그 자리에 우선적으로 들어갈 수 있는 사람들의 이름을 순서대로 배열에 넣어서 return  
입력값 : 예시 : [ “[201]James,Aden,aden], [402]Tom,Cany”] , target(새로운 방호수)  
 -> 방호수와 그 방의 사람들의 이름

우선순위 : 새로운 자리의 방호수와 사람들의 방호수의 차이의 절대값이 가장 적은 사람  
 차이의 절대값이 같다면 이름 사전순  
 방호수가 같은 사람들은 제외  
 이름에서 대문자,소문자가 다르다면 다른 사람으로 취급

문제3 :은행 계좌 프로그램 구현  
계좌생성, 입금, 출금 세가지 거래 결과에 따른 결과 코드를 배열로 return  
 입력값 : [거래형태, x, y ] 배열로 들어온다 ( 거래에 따라 입력값이 다름)

계좌생성: 계좌가 있을 시와 없을 시 결과 코드가 다르고 없으면 계좌생성  
 입금 : 해당 계좌에 입금, 입금도 물론 결과코드 존재  
 출금 : 한도내에서 출금, 한도를 초과하면 에러결과코드

문제4: 집계함수를 이용한 SQL문