



MY PAGE |



wookioki ▾



Ko ▾



Programming >



Learning >



Expertise ▾

Best Reviewer >

S/W Architect >



Online Professional
Study Guide

Programming

```
#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
#mirror ob select = 0
```

🏠 > Programming > Solving Club

Problem

User Problem

Solving Club



목록

★ Bookmark

문제 풀러 가기

H

Professional 실전실습 문제 [H1921] [Pro] 여행상품추천

성공

Problem

Result

※ 문제 및 풀이에 대해 **사외 온라인/오프라인에 게시/공유 하는 것은 금지**되어 있습니다.
SW Expert 아카데미의 문제는 **삼성전자 직원**에게만 오픈 되어 있습니다.

[제한 사항]

시간	10개 테스트케이스를 합쳐서 C++ 의 경우 1초 / Java 의 경우 2초
메모리	힙, 정적 메모리 합쳐서 256MB 이내, 스택 메모리 1MB 이내

※ 본 문제는 임직원의 S/W 문제해결 역량 향상을 위한 **Professional 실전형 연습문제**이며
함께 제공되는 **Testcase 또한 학습을 위한 연습용**입니다.



MY PAGE



wookioki



Ko



Programming



Learning



Expertise



Best Reviewer



S/W Architect



Online Professional
Study Guide

A. Main : 수정할 수 없는 코드이며, 채점 시 비 정상적인 답안 검출 등 평가를 위한 로직이 추가 될 수 있습니다.

B. User Code : 실제 응시자가 작성해야 하는 코드이며, 제출 시에는 표준 입출력 함수가 포함되어 있으면 안 됩니다.

③ Local PC 에서 프로그래밍 시 유의 사항

A. 2개의 파일을 생성하셔야 합니다. (main.cpp / solution.cpp 또는 Solution.java / UserSolution.java)

B. Main 부분의 코드를 main.cpp 또는 Solution.java 에 복사해서 사용하시기 바랍니다.

C. sample_input.txt 를 사용하기 위해서는 Main 부분의 코드 내에

표준 입력을 파일로 전환하는 코드 (주석처리 되어 있음) 의 주석을 풀어서 사용하시면 됩니다.

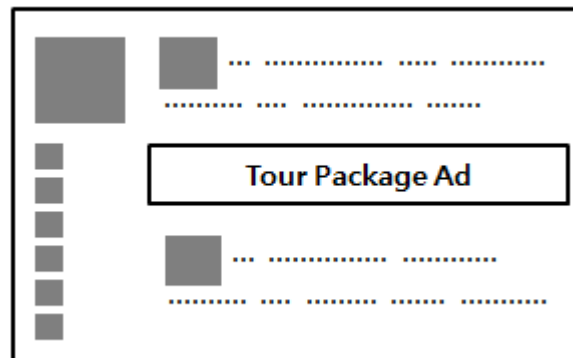
D. User Code 부분의 코드를 작성하신 후 서버에 제출하실 때,

디버깅을 위한 표준 입출력 함수를 모두 삭제 또는 주석 처리해 주셔야 합니다.

④ 문제 내에 제약조건을 모두 명시하지 않으므로 주어지는 코드를 분석하셔야 합니다.

⑤ 코드는 개발 언어에 따라 상이할 수 있으므로, 작성할 언어를 기준으로 분석하셔야 합니다.

[문제 설명]



[Fig. 1]

N 명의 사용자들이 있는 소셜 미디어 사이트가 있다.



MY PAGE |



wookioki ▾



Ko ▾



Programming >



Learning >



Expertise ▾

Best Reviewer >

S/W Architect >

Online Professional
Study Guide

사용자가 예약한 여행 상품은 여행 상품 목록에서 삭제되고 더 이상 판매되지 않는다.

각 여행 상품에는 상품 ID, 지역, 가격 정보가 있다.

이 사이트에서 사용자가 가장 선호할 만한 여행 상품을 선정하고 그 사용자에게 여행 상품을 추천하는 기능을 제공하고자 한다.

사용자에게 가장 선호할 만한 여행 상품은 다음과 같이 선정된다.

- ㉠ 사용자와 사용자의 친구들이 예약한 여행 상품 중 가장 많이 예약한 지역(들)을 찾는다.
- ㉡ ㉠ 과정에서 찾은 지역을 가지는 여행 상품들 중 가격이 가장 싼 여행 상품을 선정한다.
- ㉢ ㉡ 과정에서 찾은 가격이 가장 싼 여행 상품이 다수인 경우, 그 상품들 중에서 상품 ID가 가장 작은 여행 상품을 선정한다.
- ㉣ 만약 ㉡ 과정에서 해당하는 여행 상품이 없는 경우, 사용자와 사용자의 친구들이 두번째로 많이 예약한 지역에 대해서 ㉡와 ㉢ 과정을 수행한다.
만약 그런 경우에도 해당하는 여행 상품이 없는 경우, 사용자와 사용자의 친구들이 세번째로 많이 예약한 지역에서 대해서 진행된다.
이렇게 여행 상품이 선정될 때까지 사용자와 사용자의 친구들이 많이 예약한 지역의 순서로 가장 선호할 만한 여행 상품을 찾는다.

이와 같이 사용자에게 가장 선호할 만한 여행 상품을 선정하고 추천하는 기능을 수행하는 API를 구현하라.

다음은 구현해야 할 API이다.

※ 아래 함수 signature는 C/C++에 대한 것으로 Java에 대해서는 제공되는 Solution.java와 UserSolution.java를 참고하라.

```
void init(int N, int M)
```



MY PAGE |



wookioki ▾



Ko ▾



Programming >



Learning >



Expertise ▾

Best Reviewer >

S/W Architect >

Online Professional
Study Guide

M은 여행 상품의 지역의 수를 의미한다. 여행 상품의 지역은 1 이상 M 이하의 정수 값으로 나타낸다.

각 테스트 케이스 초기에는 어느 누구도 친구가 된 사용자는 없고 등록된 여행 상품도 없다.

Parameters

N : 사이트의 사용자 수 ($5 \leq N \leq 1,000$)

M : 여행 상품의 지역의 수 ($2 \leq M \leq 10$)

void befriend(int uid1, int uid2)

사용자 uid1과 사용자 uid2는 서로 친구가 된다.

즉, 사용자 uid1은 사용자 uid2의 친구가 되고 사용자 uid2는 사용자 uid1의 친구가 된다.

uid1과 uid2는 사용자 ID를 의미한다.

uid1과 uid2가 서로 같은 경우는 없다.

이미 친구가 된 사용자 uid1과 uid2에 대해 다시 호출되는 경우는 없다.

Parameters

uid1, uid2 : 서로 친구가 되고자 하는 사용자 ID ($1 \leq uid1, uid2 \leq N$)

void add(int pid, int area, int price)

상품 ID가 pid이고 지역이 area이고 가격이 price인 여행 상품이 여행 상품 목록에 등록된다.

여행 상품 목록에 등록된 상품은 사용자에게 판매된다.

상품 ID는 여행 상품에 대한 고유한 값이다.

예약된 상품과 여행 상품 목록에 있는 상품 중 같은 상품 ID를 가지는 여행 상품은 없다.

Parameters

pid : 상품 ID ($1 \leq pid \leq 1,000,000,000$)

area : 지역 정보 ($1 \leq area \leq M$)

price : 여행 상품의 가격 ($1 \leq price \leq 1,000$)

void reserve(int uid, int pid)

사용자 ID가 uid인 사용자가 상품 ID가 pid인 여행상품을 예약한다.

예약된 상품은 여행 상품 목록에서 삭제되고 더 이상 판매될 수 없다.



MY PAGE |



wookioki ▾



Ko ▾



Programming >



Learning >



Expertise ▾

Best Reviewer >

S/W Architect >

Online Professional
Study Guide

uid : 여행 상품을 예약하려는 사용자 ID ($1 \leq uid \leq N$)
pid : 상품 ID ($1 \leq pid \leq 1,000,000,000$)

int recommend(int uid)

사용자 uid에게 가장 선호할 만한 여행 상품을 찾아 그 상품 ID를 반환한다.

해당 함수가 호출될 때, 여행 상품 목록에 여행 상품이 1 개 이상 있다.

Parameters

uid : 사용자 ID ($1 \leq uid \leq N$)

Return

사용자 uid에게 가장 선호할 만한 여행 상품의 상품 ID

[Table 1]과 같이 API가 호출되는 경우는 생각해 보자.

#	API Call	Description	Return	Figure
1	init(5, 4)	테스트 케이스 초기화, $N=5$, $M=4$		
2	befriend(5, 1)	사용자 5와 사용자 1이 서로 친구가 된다.		
3	befriend(2, 4)			
4	befriend(1, 4)			
5	befriend(2, 1)			
6	add(71, 1, 10)	상품 ID=71, 지역=1, 가격=10인 상품이 추가된다.		
7	add(67, 4, 8)			
8	add(43, 3, 4)			
9	add(3, 2, 4)			
10	add(75, 4, 4)			[Fig. 2]
11	recommend(4)	사용자 4에 대한 여행 상품을 추천한다.	3	
12	reserve(4, 75)	사용자 4가 상품 75을 예약한다.		[Fig. 3]
13	recommend(1)		67	
14	add(81, 2, 10)			
15	reserve(2, 67)			
16	add(69, 1, 5)			



MY PAGE



wookioki

Ko



Programming



Learning



Expertise



Best Reviewer



S/W Architect



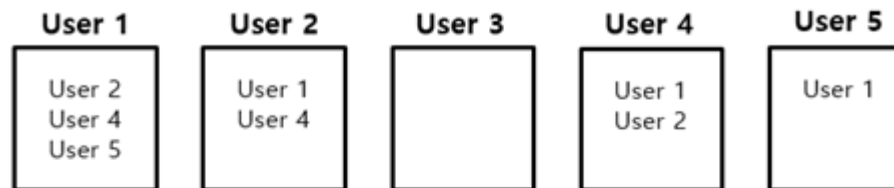
Online Professional
Study Guide

20	recommend(5)		71	
21	befriend(3, 5)			
22	befriend(3, 4)			
23	add(28, 4, 10)			[Fig. 6]
24	recommend(3)		28	

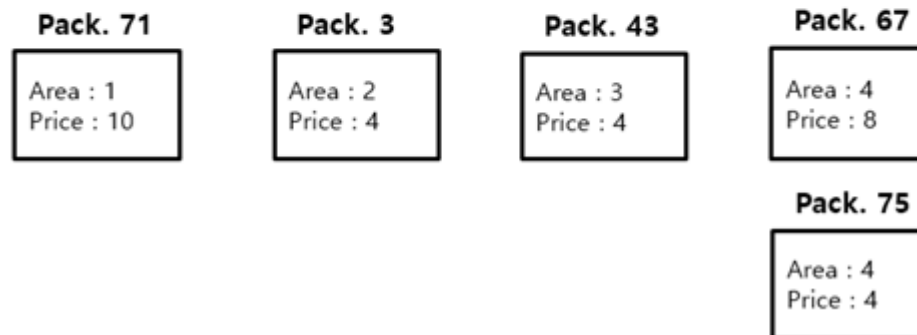
[Table 1]

10번째 add(75, 4, 4) 함수가 호출된 후에 사용자에게 대한 친구 리스트와 판매하고 있는 여행 상품은 [Fig. 2]와 같다.

Friend List



Travel Package List



[Fig. 2]



MY PAGE |



wookioki ▾



Ko ▾



Programming



Learning



Expertise



Best Reviewer



S/W Architect



Online Professional
Study Guide

따라서, 여행 상품 목록에 있는 여행 상품 중 가장 싼 여행 상품은 상품 3, 상품 43, 상품 75이다.

상품 ID가 가장 작은 여행 상품은 상품 3이다. 상품 3을 추천한다.

12번째에 `reserve(4, 75)`가 호출되면 상품 75는 예약되고 여행 상품 목록에서 상품 75는 삭제된다.

Friend List

User 1	User 2	User 3	User 4	User 5
User 2 User 4 User 5	User 1 User 4		User 1 User 2	User 1
Reservations			Pack. 75 Area : 4	

Travel Package List

Pack. 71	Pack. 3	Pack. 43	Pack. 67
Area : 1 Price : 10	Area : 2 Price : 4	Area : 3 Price : 4	Area : 4 Price : 8

[Fig. 3]

13번째에 `recommend(1)`이 호출될 때 사용자 1과 그의 친구들이 가장 많이 예약한 지역은 4이다.

이 경우에 지역이 4이고 가장 싼 여행 상품은 상품 67이므로 상품 67을 추천한다.

17 번째 `add(55, 3, 6)` 함수가 호출된 후 [Fig. 4]와 같이 된다.



MY PAGE |



wookioki ▾

Ko ▾



Programming >



Learning >



Expertise ▾

Best Reviewer >

S/W Architect >

Online Professional
Study Guide

Friend List

User 1	User 2	User 3	User 4	User 5
User 2 User 4 User 5	User 1 User 4		User 1 User 2	User 1
Reservations	Pack. 67 Area : 4		Pack. 75 Area : 4	

Travel Package List

Pack. 71	Pack. 3	Pack. 43
Area : 1 Price : 10	Area : 2 Price : 4	Area : 3 Price : 4
Pack. 69	Pack. 81	Pack. 55
Area : 1 Price : 5	Area : 2 Price : 10	Area : 3 Price : 6

[Fig. 4]

18번째에 recommend(2) 함수가 호출될 때 사용자 2와 그의 친구들이 가장 많이 예약한 지역은 4이다.

그러나, 지역이 4인 여행 상품은 여행 상품 목록에 없다.

그 다음으로 많이 예약한 지역은 1, 2, 3이다. 이 지역을 가지고 가장 싼 여행 상품은 상품 3과 상품 43이다.

상품 ID가 작은 상품인 상품 3이 추천된다.



MY PAGE |



wookioki ▾



Ko ▾



Programming



Learning



Expertise



Best Reviewer



S/W Architect



Online Professional
Study Guide

Friend List

User 1	User 2	User 3	User 4	User 5
User 2 User 4 User 5	User 1 User 4		User 1 User 2	User 1
Reservations	Pack. 67		Pack. 75	Pack. 69
	Area : 4		Area : 4	Area : 1

Travel Package List

Pack. 71	Pack. 3	Pack. 43
Area : 1 Price : 10	Area : 2 Price : 4	Area : 3 Price : 4
	Pack. 81	Pack. 55
	Area : 2 Price : 10	Area : 3 Price : 6

[Fig. 5]

20번째에 recommend(5)이 호출될 때, 사용자 5와 그의 친구가 가장 많이 예약한 지역은 1이다.

여행 상품 목록에 지역이 1인 여행 상품은 상품 71이므로 상품 71이 추천된다.

23번 add(28, 4, 10) 함수가 호출된 후 [Fig. 6]과 같이 된다.



MY PAGE |



wookioki ▾



Ko ▾



Programming >



Learning >



Expertise ▾

Best Reviewer >

S/W Architect >

Online Professional
Study Guide

User 2 User 4 User 5	User 1 User 4	User 4 User 5	User 1 User 2 User 3	User 1 User 3
Reservations	Pack. 67		Pack. 75	Pack. 69
	Area : 4		Area : 4	Area : 1

Travel Package List

Pack. 71	Pack. 3	Pack. 43	Pack. 28
Area : 1 Price : 10	Area : 2 Price : 4	Area : 3 Price : 4	Area : 4 Price : 10
	Pack. 81	Pack. 55	
	Area : 2 Price : 10	Area : 3 Price : 6	

[Fig. 6]

24번째에 recommend(3) 함수가 호출될 때 사용자 3과 그의 친구들이 가장 많이 예약한 지역은 1과 4이다.

지역이 1 또는 4를 가지고 가장 싼 여행 상품은 상품 71과 상품 28이다.

상품 ID가 가장 작은 상품은 상품 28이므로 상품 28이 추천된다.

[제약사항]

1. 사용자의 수 N 은 5 이상 1,000 이하이다. ($5 \leq N \leq 1,000$)
2. 사용자 ID는 1 부터 N 까지 있다.
3. 지역의 수 M 은 2 이상 10 이하이다. ($2 \leq M \leq 10$)



MY PAGE



wookioki



Ko



Programming



Learning



Expertise



Best Reviewer



S/W Architect



Online Professional
Study Guide

6. 서로 다른 여행 상품이 같은 상품 ID 값을 가질 수 없다.
7. 여행 상품의 가격은 1 이상 1,000 이하의 정수 값을 가진다.
8. 각 테스트 케이스에서 한 사용자의 친구의 수는 최대 20이다.
9. 각 테스트 케이스에 대해서 모든 함수의 총 호출 횟수는 100,000 회 이하이다.
10. 각 테스트 케이스에 대해서 add() 함수는 40,000 회 이하 호출된다.
11. 각 테스트 케이스에 대해서 recommend() 함수는 60,000 회 이하 호출된다.

[입출력]

입출력은 제공되는 Main 부분의 코드에서 처리하므로 User Code 부분의 코드에서는 별도로 입출력을 처리하지 않는다.

sample input에 대한 정답 출력 결과는 아래와 같은 형태로 보여진다.

```
#1 100
#2 100
#3 100
#4 100
#5 100
```

연관 MOOC

S/W 문제해결 : [H1921]
Recommend a Holiday
Package



MY PAGE



wookioki

Ko



Programming



Learning



Expertise



Best Reviewer



S/W Architect



Online Professional
Study Guide

연관 TALK

[H1921] 여행상품추천

29 Comments

최신순 | 인기순

고정닉네임



0 / 1000 char

등록

^^ 2019-10-18 15:55:06

베스트댓글

21

댓글

기왕이면 50개 테케가 다 공개되면 좋겠네요.. 이 문제는 10개 수습초 나와도 accept이라 실제 시험과 괴리가 있어서, 실력향상을 위해 고려부탁드립니다.

Unknown 2019-10-10 21:17:02

삭제된 코멘트입니다

아..진짜 2019-10-18 10:52:24

0

댓글

테케7까지 잘 가다가 8부터 0점인데 유사한 문제 있으신 분 없나요..어케 풀어야 할지 난감



저도ㅠㅠ 2019-10-20 23:22:08

0

댓글



MY PAGE |



wookioki ▾



Ko ▾



Programming >



Learning >



Expertise ▾

Best Reviewer >

S/W Architect >

Online Professional
Study Guide

8019라인 300 46 2 21 => pid 46인 package가 추가됩니다. (area 2, price 21)
11780라인 300 4393 2 21 => pid 4393인 package가 추가됩니다. (area 2, price 21)
15103라인 500 402 4393 => uid 402에게 추천 요청
: 여기에서 제 프로그램은 pid 46을 리턴합니다. (정답은 pid 4393입니다.)

추천시까지 uid 449, 402만 서로 친구이고, reserved된 package는 없습니다.
그러면, price는 같고, pid가 더 작으니까 pid 46이 정답인 것 같은데요.

저도 ππ 더 이상 디버깅을 못하고 있습니다. 맞추신 분들^^ 도움 주세요.



카프카 2019-10-21 00:28:40

👍 2

댓글

친구를 저장하는 변수의 사이즈는 자기 자신까지 합하면 21이 되어야 하는데 혹시 이를 20으로 잡은건 아닌지요?



희망적^^ 2019-10-21 13:32:07

👍 1

댓글

말씀하신 부분 확인해 보니까 잘못되어 있어서 배열 크기를 MAX_FRIEND+1로 수정했습니다.
(감사합니다^^ 유사한 시험이 나와도 놓칠 뻔 했네요.)
또 놓치는 게 있을까 싶어서 잠시 더 확인했는데,
8번 샘플 데이터 실행 중, reserve pid(46)이 호출되는군요.
제가 뭔가 잘못 처리하고 있나봐요. 오늘 밤에 디버깅 후 후기 올려 보겠습니다.
감사합니다. :)



Pass :) 2019-10-21 23:55:24

👍 0

댓글

통과했습니다.
여행 예약할 때, hash table에서 해당 pid의 hash index의 key값을 0으로 초기화했었는데, 이로 인해 side-effect이 있었습니다.
(흠.. 공간을 재사용한다는 생각이었는데, 그러면 안되는 거였어요.)

```
#define ht_removeIndex(h) (tb[h].key = 0)
=> 이 함수 호출하는 부분 삭제
```



MY PAGE |



wookioki ▾



Ko ▾



Programming



Learning



Expertise



Best Reviewer



S/W Architect



Online Professional
Study Guide

이도 인해 pid 46를 예약상태도 만들어야 하는데, 예약 flag를 셋팅을 못했고,
valid한 pid로 인식되어서 리턴하게 되었네요.

요약하면,
* friends 배열을 MAX_FRIEND+1로 하고,
* 예약할 때 hash index의 key값 초기화하던 것 제거하고
Pass했습니다.
카프카님 감사합니다. :)

└ 저도... 2019-10-25 12:27:35

👍 0

댓글

자기자신+친구 합하면 21이 되어야하네요.
요것때문에 헤맸네요. 감사합니다.

└ 아..진짜 2019-11-07 17:33:48

👍 0

댓글

아 진짜...
원인은 heap에 사용한 배열 초기화를 안 한 거였어요..허무..ㅜㅜ

^^ 2019-10-18 15:55:06

👍 21

댓글

기왕이면 50개 테케가 다 공개되면 좋겠네요.. 이 문제는 10개 수십초 나와도 accept이라 실제 시험과 괴리가 있어서, 실력향상을 위해 고려부탁드립니다.

└ 네 2019-10-18 16:40:01

👍 1

댓글

결국 예외 케이스에 대한 대처가 실력일텐데, 그런 부분을 연습할 기회가 좀 없는거 같아요

└ 비공개 2019-10-25 12:30:35

👍 4

댓글

그건 공개가 되면. 의미가 없어지죠



MY PAGE |



wookioki ▾

Ko ▾

19

댓글



Programming >



Learning >



Expertise ▾

Best Reviewer >

S/W Architect >

Online Professional
Study Guide

어떤 예외 케이스가 나오는지 알아야 연습을 하죠,
이런식으로 채점 기준이 블라인드면 실력향상에 아무런 도움이 되지 않습니다.



gorapaduck 2019-10-23 23:51:50

0

댓글

reserve에서 만약 (우선순위가 제일높은게 삭제되면) packagesArea배열은 어떻게 업데이트해야하나요?? 이부분이 잘 이해가 안되네요ㅜㅜ 영상 18분에서 Pop 후에 새로운 원소를 뽑아주시면 된다고 하셨는데 이 과정에 대해서 설명해주실 수 있나요?



제 생각엔 2019-10-24 10:42:31

0

댓글

add시에 priority queue에 밀어넣고, reserve시에는 PQ를 recommend기준으로 다시 업데이트하는 방법을 써야 바로 pop을 통해 원하는 답을 뽑을거 같고, 코드가 심플해질거 같습니다만 머리와 구현이 따로 놓고 있어서 어렵네요..ㅎㅎ



카프카 2019-10-25 22:48:56

0

댓글

1. 일단 add시마다 hash에도 연결하고 priority queue에도 밀어넣습니다.
2. hash에 연결하는 노드에 priority queue의 index를 미리 저장합니다.
3. reserve시 hash table에서 pid로 검색하면 2번에서 미리 저장해놓은 PQ index를 알고있으니 해당 index의 노드와 마지막 index의 노드를 SWAP 합니다.
(PQ Pop 시 0번을 마지막번으로 교체 후 heap down update를 수행하는 것과 같은원리 입니다.)
4. 해당 index를 기준으로 heap down update, heap up update를 수행해주면 업데이트가 완료됩니다.
(참고로 SWAP 시 index도 같이 SWAP 해줘야 합니다.)



제 생각엔 2019-11-07 14:45:20

0

댓글

감사합니다. 일단 모범답안에서 제시한 해법대로 해도 pass는 되네요.
- 일단 PQ에서 뽑은 후에 그 PID가 상품리스트에 없으면 다시 POP해서 사용하는 방식...



MY PAGE |



wookioki ▾

Ko ▾



Programming



Learning



Expertise



Best Reviewer



S/W Architect

Online Professional
Study Guide

댓글

시간복잡도 줄이기가 어려워요 흑흑



vnnvnn 2019-11-11 11:36:34



댓글

초기화에서 시간을 많이 소요했더라고요 ㅠ
알고리즘 적으로 더이상 줄일부분이 없으신 분들은 초기화 부분을 신경써보세욤....

어렵다..ㅠ 2019-11-26 17:00:16



댓글

혹시 TC 6번 부터 fail나면 어떻게 문제일까요??
5번까지는 괜찮은데.. 6번부터.. 문제네요.. ㅠㅠ



저도 2019-11-26 20:35:28



댓글

저도 6번부터 0점이네요..초기화도 해보고 배열 사이즈도 늘려보고...모르겠네요



julie 2020-05-20 20:26:21



댓글

저도 6번부터 fail 났었는데, 친구들이 예약한 지역목록을 찾을 때, 친구별 해당 지역 예약 여부 뿐 아니라 예약 건수도 반영하여 대상 지역을 선정해야 하더라고요.

예를 들어, 친구 1은 지역1을 3번 여행했고, 친구 2는 지역 2를 1번 여행했고, 나는 아무곳도 여행한 지역이 없을 경우,

- 친구 1과 친구 2가 각각 지역 1과 지역 2를 여행했으므로, 지역1과 지역2가 대상 여행지가 아니라,

- 지역 1은 3번, 지역 2는 1번 친구들이 여행했으므로, 가장 많이 여행한 지역은 지역 1이 되며 지역 1만 대상 여행지가 되는 방식으로

계산했더니 풀리더라고요.



강세 2021-05-26 21:23:07



댓글



MY PAGE |



wookioki ▾

Ko ▾



Programming >



Learning >



Expertise ▾

Best Reviewer >

S/W Architect >

Online Professional
Study Guide

예제나 문제 설명에는 마지막에 예약한거 하나만 혹은 애초에 한사람은 예약을 하나만 할 수 있는것 처럼 설명이 되 있는데
매번 느끼지만 이렇게 모호하게 해석되는 문제를 내는게 이해가 안되네요
나머지는 풀 수 있었지만 이 댓글 없었으면 절대 못풀었을것 같습니다. 더러운 문제좀 안만나고 싶네요
글을 쓰신지 몇년이나 지났지만 감사합니다.



GOFORBR 2020-03-03 19:15:55

👍 2

댓글

아래 문장에서 "그 다음으로 많이 예약한 지역은..." 이 아니라 "그 다음으로 상품이 많은 지역은..." 으로 변경해야 하는 것 아닐까요?

사실 해당 시점에 예약한 지역 카운트가 모두 0이어서 이 부분을 해석하는데 상당히 힘들었습니다. πππ

18번째에 recommend(2) 함수가 호출될 때 사용자 2와 그의 친구들이 가장 많이 예약한 지역은 4이다.

그러나, 지역이 4인 여행 상품은 여행 상품 목록에 없다.

그 다음으로 많이 예약한 지역은 1, 2, 3이다. 이 지역을 가지고 가장 싼 여행 상품은 상품 3과 상품 43이다.

상품 ID가 작은 상품인 상품 3이 추천된다.



베리가소원 2021-03-17 00:31:02

👍 0

댓글

맞아요, 저도 이 부분이 잘 이해가 안되네요 π.π 18번 이전에 예약된 건 pid 67, 75 두개고 둘다 지역이 4니까 지역 4만 두번 예약된 셈인데 '그 다음으로 많이 예약한 지역'이 어떻게 1,2,3이 될 수 있답니까.. 저랑 비슷하게 생각하신 분이 계셔서 그나마 위안이 되네요 π.π



흙 2021-04-02 15:11:29

👍 0

댓글

상품의 지역 개수 합은 recommend에서 고려 대상이 아닌것같습니다.

예약된상품의 지역 합 : 1등 지역4(2번), 공동2등 지역 1,2,3(0번)



MY PAGE |



wookioki ▾

Ko ▾



Programming >



Learning >



Expertise ▾

Best Reviewer >

S/W Architect >



Online Professional
Study Guide

삭제된 코멘트입니다

Unknown 2021-02-25 16:29:12

삭제된 코멘트입니다



redy 2021-07-29 17:54:01

👍 0

댓글

STL 사용이 안되는 것 같은데 확인 부탁드립니다.



팡이 2021-08-10 00:33:47

👍 0

댓글

저도 다풀고 돌려보니까 컴파일실패되네요 ㅜㅜ map, list사용했는데

SAMSUNG

About S/W Expert Academy

찾아오는 길

SiteMap

이용약관

개인정보처리방침

^
TOP

경기도 용인시 기흥구 서천동로 59 (서천동 822) 삼성전자 인재개발원 | swexpert.sec@samsung.com