

(https://news.samsung.com/kr) (https://news.samsung.com/kr/)

## 친구들

▼ 제출 현황

4/10

▼ 문제별 현황

공지사항

1:1 문의

2021-07-16 15:00 ~ 2021-07-17 15:00

00:24:09

$N$ 명의 사람들이 좌우로 서 있다. 사람들은 왼쪽에서 순서대로 1번부터 번호가 붙어 있다. 사람들 사이의 친구 관계는 사람들이 들고 있는 자연수를 이용하여 아래 규칙으로 알아낼 수 있다. 아래 규칙에 따라 만들어지지 않는 친구 관계는 없다.

- 번호  $i$ 인 사람은 자연수  $D_i$ 를 가지고 있다. ( $0 < D_i \leq N$ )
- 번호  $i$ 인 사람은 번호  $i + D_i$ 인 사람과 친구 관계이다. 친구 관계는 대칭적이다. 즉, 한쪽만 상대방이 친구라고 생각하는 경우는 없다. 만약 번호  $i + D_i$ 인 사람이 존재하지 않는다면, 이  $D_i$ 는 무시된다.
- 사람  $A$ 와  $B$ 가 친구 관계이고  $B$ 와  $C$ 가 친구 관계이면,  $A$ 와  $C$ 도 친구 관계이다.

친구 관계인 극대 그룹은 사람들의 모임인데, 그룹 내의 모든 사람들이 친구 관계이고, 그 조건을 유지하면서 더 이상 사람을 추가할 수 없는 경우를 말한다. 예를 들어 아래와 같이 5명이 각자 가지고 있는  $D_i$ 를 표시했다고 하자.

1	1	3	3	3
---	---	---	---	---

1번 사람과 2번 사람, 2번 사람과 3번 사람이 친구 관계라는 것은 직접 알 수 있다.

규칙에 따라 1번 사람과 3번 사람도 친구 관계이다.

3, 4, 5번 사람이 가지고 있는 수는 무시된다.

그룹 {1번 사람, 3번 사람}을 생각해 보자. 이 그룹 안의 모든 사람은 친구 관계이다.

하지만 2번 사람을 추가해도 모두 친구 관계가 되므로 {1번 사람, 3번 사람}은 친구 관계인 극대 그룹이 아니다.

그룹 {1번 사람, 2번 사람, 3번 사람}은 친구 관계인 극대 그룹이다.

그룹 {1번 사람, 4번 사람, 5번 사람}은 친구 관계가 아닌 쌍이 있으므로 친구 관계인 극대 그룹이 아니다.

친구 관계인 극대 그룹은 {1번 사람, 2번 사람, 3번 사람}, {4번 사람}, {5번 사람}의 3개가 있음을 알 수 있다.

각 사람이 들고 있는  $D_i$ 들을 입력 받아 친구 관계인 극대 그룹의 개수를 출력하는 프로그램을 작성하시오.

- 제한시간: 전체 테스트 케이스는 50개 이하이며, 전체 수행 시간은 1초 이내. (Java 2초 이내)

제한 시간을 초과하면 제출한 소스코드의 프로그램이 즉시 종료되며,

그때까지 출력한 내용이 파일에 저장되지 않아 점수가 제대로 반영되지 않을 수 있습니다.

그러나, 제한 시간을 초과하더라도 테스트 케이스를 1개 그룹 이상 통과하였다면 '부분 점수( $0 < \text{점수} < \text{만점}$ )'를 받을 수 있으며,

이를 위해서는, C / C++ 에서 "printf 함수" 사용할 경우, 프로그램 시작부분에서 "setbuf(stdout, NULL);"를 한번만 사용하십시오.

C++에서는 "setbuf(stdout, NULL);"와 "printf 함수" 대신 "cout"를 사용하고, Java에서는 "System.out.println"을 사용하시면,

제한 시간을 초과하더라도 '부분 점수'를 받을 수 있습니다.

※ 언어별 기본 제공 소스코드 내

용 참고

만약, 제한 시간을 초과하지 않았는데도 '부분 점수'를 받았다면, 일부 테스트 케이스를 통과하지 못한 경우입니다.

- 메모리 사용 제한 : heap, global, static 총계 256MB, stack 100MB

- 제출 제한 : 최대 10회

## 메모리 사용 제한

heap, global, static (총계) : 256MB

stack : 100MB

## 입력

입력 파일에는 여러 테스트 케이스가 포함될 수 있다.

파일의 첫째 줄에 테스트 케이스의 개수를 나타내는 자연수  $T$ 가 주어지고,

이후 차례로  $T$ 개의 테스트 케이스가 주어진다. ( $1 \leq T \leq 50$ )

각 테스트 케이스의 사람들의 수  $N$ 이 주어진다. ( $N \leq 100,000$ )

다음 줄에는 왼쪽부터 각 사람들이 들고 있는 자연수가 주어진다.

- 점수 : 각 제출에서 취득한 점수 중에서 최대 점수 (만점 80 점)

주어지는 테스트 케이스 데이터들의 그룹은 아래와 같으며,

각 그룹의 테스트 케이스를 모두 맞추었을 때 해당되는 부분 점수를 받을 수 있다.

· 그룹 1 (30 점) :  $N \leq 1,000$

· 그룹 2 (50 점) : 이 그룹의 테스트 케이스에서는 원래의 조건 외에는 다른 제약조건이 없다.

\* 모든 테스트 케이스를 풀지 않고 일부분의 그룹에 속하는 테스트 케이스만을 푸는 경우에도 입력 받은 모든 케이스에 대해 (답이 틀릴지라도) 출력 양식에는 맞는 출력을 생성해야 점수가 반영되는 것이 보장된다.

\* 제한 시간을 초과하면 제출한 소스코드의 프로그램이 즉시 종료되며, 그때까지 출력한 내용이 파일에 저장되지 않아 점수가 제대로 반영되지 않을 수 있습니다.

### 출력

각 테스트 케이스의 답을 순서대로 표준출력으로 출력하여야 하며,  
각 테스트 케이스마다 첫 줄에는 “Case # $C$ ”를 출력하여야 한다. 이때  $C$ 는 테스트 케이스의 번호이다.  
그 다음 줄에, 친구 관계인 그룹의 수를 출력하시오.

### 입출력에

입력

```
2
5
1 1 3 3 3
10
8 10 5 4 2 5 1 3 1 9
```

출력

```
Case #1
3
Case #2
4
```

C++14(10.2.0) ▼

코드 초기화

```
1 #include <iostream>
2 #include <vector>
3
4 using namespace std;
5
6 int Answer;
7
8 void dfs(vector<vector<int>>& adj, vector<bool>& check, int cur) {
9     check[cur] = true;
10    for(int i=0; i<adj[cur].size(); i++) {
11        if(!check[adj[cur][i]]) dfs(adj, check, adj[cur][i]);
12    }
13 }
14
15 int solve() {
16     int N, res = 0;
17     cin >> N;
18     vector<vector<int>> adj;
19     vector<bool> check(N+10, false);
20     adj.resize(N+10);
21     for(int i=1; i<=N; i++) {
22         int tmp;
23         cin >> tmp;
```

```
24         if(i+tmp > N) continue;
25         adj[i].push_back(i+tmp);
26         adj[i+tmp].push_back(i);
27     }
```

sample\_input.txt

<b>목록</b>	<b>임시저장</b>	<b>컴파일</b>	<b>제출</b>
-----------	-------------	------------	-----------