Microsoft Questions

1)

You are given an array A of N integers and an integer S. Your task is to compute how many ways one can choose a contiguous fragment of A that has an arithmetic mean equal to S. The arithmetic mean (average) of a fragment is the sum of the elements of the fragment divided by its length. For example, the arithmetic mean of [1, 4, 4, 5] is 14/4 = 3.5.

Write a function:

```
int solution(vector<int> &A, int S);
```

which returns the number of contiguous fragments of A whose arithmetic means are equal to S.

If the result is greater than 1,000,000,000, your function should return 1,000,000,000.

Examples:

1. Given A = [2, 1, 3] and S = 2, your function should return 3, since the arithmetic means of fragments [2], [1, 3] and [2, 1, 3] are equal to 2.

2. Given A = [0, 4, 3, −1] and S = 2, your function should return 2, since fragments [0, 4] and [4, 3, −1] have an arithmetic mean equal to 2.

3. Given A = [2, 1, 4] and S = 3, your function should return 0, since there exist no contiguous fragments whose arithmetic mean is equal to 3.

Write an **efficient** algorithm for the following assumptions:

Write an **efficient** algorithm for the following assumptions:

- N is an integer within the range [1..100,000];
- S is an integer within the range [−1,000,000,000..1,000,000,000];
- each element of array A is an integer within the range [−1,000,000,000..1,000,000,000].

2)

You are given an array A of N integers and an integer S. Your task is to compute how many ways one can choose a contiguous fragment of A that has an arithmetic mean equal to S. The arithmetic mean (average) of a fragment is the sum of the elements of the fragment divided by its length. For example, the arithmetic mean of [1, 4, 4, 5] is 14/4 = 3.5.

Write a function:

```
int solution(vector<int> &A, int S);
```

which returns the number of contiguous fragments of A whose arithmetic means are equal to S.

If the result is greater than 1,000,000,000, your function should return 1,000,000,000.

Examples:

1. Given A = [2, 1, 3] and S = 2, your function should return 3, since the arithmetic means of fragments [2], [1, 3] and [2, 1, 3] are equal to 2.

2. Given A = [0, 4, 3, −1] and S = 2, your function should return 2, since fragments [0, 4] and [4, 3, −1] have an arithmetic mean equal to 2.

3. Given A = [2, 1, 4] and S = 3, your function should return 0, since there exist no contiguous fragments whose arithmetic mean is equal to 3.

Write an **efficient** algorithm for the following assumptions:

3)

There are N balls positioned in a row. Each of them is either red or white. In one move we can swap two adjacent balls. We want to arrange all the red balls into a consistent segment. What is the minimum number of swaps needed?

Write a function:

```
int solution(string &S);
```

that, given string S of length N built from characters `"R"` and `"W"`, representing red and white balls respectively, returns the minimum number of swaps needed to arrange all the red balls into a consistent segment. If the result exceeds 10^9, return -1.

Examples:

1. Given S = "WRRWWR", your function should return 2. We can move the last ball two positions to the left:

- "WRRWRW"
- "WRRRWW"

2. Given S = "WWRWWWRWR", your function should return 4. We can move first and last red ball towards the middle one:

- "WWWRWWRWR"
- "WWWWRWRWR"
- "WWWWWRRWR"

2. Given S = "WWRWWWRWR", your function should return 4. We can move first and last red ball towards the middle one:

- "WWW**R**WWRWR"
- "WWWW**R**WRWR"
- "WWWWW**R**RWR"
- "WWWWWRR**RW**"

3. Given S = "WWW", your function should return 0. There are no red balls to arrange into a segment.

4. Given S is "RW" repeated 100,000 times, your function should return -1. The minimum needed number of swaps is greater than 10^9.

Write an **efficient** algorithm for the following assumptions:

- N is an integer within the range [1..200,000];
- string S consists only of the characters "R" and/or "W".