5675. Closest Subsequence Sum

My Submissions (/contest/weekly-contest-227/problems/closest-subsequence-sum/submissions/) Back to Contest (/contest/weekly-contest-227/)

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Hard

You are given an integer array `nums` and an integer `goal`.

You want to choose a subsequence of `nums` such that the sum of its elements is the closest possible to `goal`. That is, if the sum of the subsequence's elements is `sum`, then you want to **minimize the absolute difference** `abs(sum - goal)`.

Return *the* ***minimum*** *possible value of* `abs(sum - goal)`.

Note that a subsequence of an array is an array formed by removing some elements **(possibly all or none)** of the original array.

Example 1:

```
Input: nums = [5,-7,3,5], goal = 6
Output: 0
Explanation: Choose the whole array as a subsequence, with a sum of 6.
This is equal to the goal, so the absolute difference is 0.
```

Example 2:

```
Input: nums = [7,-9,15,-2], goal = -5
Output: 1
Explanation: Choose the subsequence [7,-9,-2], with a sum of -4.
The absolute difference is abs(-4 - (-5)) = abs(1) = 1, which is the minimum.
```

Example 3:

```
Input: nums = [1,2,3], goal = -7
Output: 7
```

Constraints:

- `1 <= nums.length <= 40`
- -10^7 `<= nums[i] <=` 10^7
- -10^9 `<= goal <=` 10^9

JavaScript

```
/**
 * @param {number[]} nums
 * @param {number} goal
 * @return {number}
 */
var minAbsDifference = function(nums, goal) {

};
```