539. Minimum Time Difference

Given a list of 24-hour clock time points in **"HH:MM"** format, return *the minimum **minutes** difference between any two time-points in the list*.

Example 1:

```
Input: timePoints = ["23:59","00:00"]
Output: 1
```

Example 2:

```
Input: timePoints = ["00:00","23:59","00:00"]
Output: 0
```

Constraints:

- `2 <= timePoints.length <= 2 * 10^4`
- `timePoints[i]` is in the format "**HH:MM**".

539. Minimum Time Difference

```
/*
   Sorting
   Time complexity: O(n+nlogn+n)=O(nlogn)
   Space complexity: O(n)
*/
typedef std::vector<bool> vb;
typedef std::vector<int> vi;
typedef std::vector<std::string> vs;
class Solution {
   public:
      int get_hours(std::string& h){
         return (h[0]-'0')*10+h[1]-'0';
      }
      int get_minutes(std::string& h){
         return (h[3]-'0')*10+h[4]-'0';
      }
      vi convert_to_minutes(vs& timePoints){
         vi minutes;
         for(auto& h: timePoints){
            int hh=get_hours(h);
            int mm=get_minutes(h);
            minutes.push_back(hh*60+mm);
         }
         return minutes;
      }
      int findMinDifference(vs& timePoints){
         int n=timePoints.size();
         vi minutes=convert_to_minutes(timePoints);
         std::sort(minutes.begin(),minutes.end());
         int ans=24*60-minutes[n-1]+minutes[0];
         for(int i=0;i<n-1;++i){
            ans=std::min(ans,abs(minutes[i]-minutes[i+1]));
         }
         return ans;
      }
   };
```