

2678. Number of Senior Citizens

You are given a **0-indexed** array of strings `details`. Each element of `details` provides information about a given passenger compressed into a string of length `15`. The system is such that:

- The first ten characters consist of the phone number of passengers.
- The next character denotes the gender of the person.
- The following two characters are used to indicate the age of the person.
- The last two characters determine the seat allotted to that person.

Return *the number of passengers who are **strictly more than 60 years old***.

Example 1:

Input: `details = ["7868190130M7522","5303914400F9211","9273338290F4010"]`

Output: `2`

Explanation: The passengers at indices 0, 1, and 2 have ages 75, 92, and 40. Thus, there are 2 people who are over 60 years old.

Example 2:

Input: `details = ["1313579440F2036","2921522980M5644"]`

Output: `0`

Explanation: None of the passengers are older than 60.

Constraints:

- `1 <= details.length <= 100`
- `details[i].length == 15`
- `details[i]` consists of digits from '0' to '9'.
- `details[i][10]` is either 'M' or 'F' or 'O'.
- The phone numbers and seat numbers of the passengers are distinct.

2678. Number of Senior Citizens

/*

Time complexity: $O(n)$

Extra space complexity: $O(1)$

*/

class Solution {

public:

```
int countSeniors_v1(vector<string>& details) {
    int ans=0;
    for(auto& d: details){
        int tens=d[11]-'0';
        int units=d[12]-'0';
        if(units+tens*10>60) ans++;
    }
    return ans;
}
```

```
int countSeniors_v2(std::vector<std::string>& details) {
    int ans=0;
    for(auto& d: details){
        char tens=d[11];
        char units=d[12];
        if(tens>'6' || tens=='6' && units>'0') ans++;
    }
    return ans;
}
```

};