



Shri Vile Parle Kelavani Mandal's  
**INSTITUTE OF TECHNOLOGY**  
**DHULE (M.S.)**  
**DEPARTMENT OF COMPUTER ENGINEERING**

**Subject:** Competitive Programming Lab (BTCOL606)

**Name :** Mohammad Anas Aarif Baig Mirza

**Roll No. :** 31

**Class :** T.Y Comp

**Batch :** T2

**Division:** T

**Expt. No. :**04

**Date :**

**Title : Problem 4:**

Remark

Signature

**Code:**

```
// MOHAMMAD_ANAS_31_TY_COMP
```

```
#include <iostream>
```

```
#include <vector>
```

```
#include <algorithm>
```

```
using namespace std;
```

```
const int maxcontestants = 101;
```

```
const int maxproblems = 9;
```

```
int problemssolved[maxcontestants] = {0};
```

```
int penalty[maxcontestants] = {0};
```

```
bool hassubmitted[maxcontestants] = {0};
```

```
int incorrectattempts[maxcontestants][maxproblems] = {0};
```

```
bool compareContestant(int a, int b)
```

```
{
```

```
    if (problemssolved[a] != problemssolved[b] )
```

```
        return problemssolved[a] > problemssolved[b];
```

```

        if (penalty[a] != penalty[b])
            return penalty[a] < penalty[b];

    return a < b;
}

int main()
{
    int n;
    cout << "Enter the input Text Size: ";
    cin >> n;

    vector<int> contestantid(n);
    vector<int> problemNo(n);
    vector<int> timeR(n);
    vector<char> judge(n);

    for (int i = 0; i < n; i++)
    {
        cout << "Enter Contestant ID, Problem No, Time, and Judge (e.g., 1 2 10 C):
";
        cin >> contestantid[i] >> problemNo[i] >> timeR[i] >> judge[i];
    }

    cout << "\nInput Data:\n";
    for (int i = 0; i < n; i++)
    {
        cout << contestantid[i] << " " << problemNo[i] << " " << timeR[i] << " " <<
judge[i] << endl;
    }
    for(int i = 0; i < n; i++)
    {
        int c = contestantid[i];
        int p = problemNo[i];
        int t = timeR[i];
        char j = judge[i];
    }
}

```

```

hassubmitted[c] = true;

if(j == 'C')
{
    if(problemssolved[c] == 0 || incorrectattempts[c][p] != -1)
    {
        problemssolved[c]++;
        int penaltyTime = incorrectattempts[c][p] * 20;

        penalty[c] += t + penaltyTime;
        incorrectattempts[c][p] = -1;
    }
}
else if(j == 'I')
{
    if(incorrectattempts[c][p] != -1)
    {
        incorrectattempts[c][p]++;
    }
}
}

vector<int> contestants;
for(int i = 0; i < maxcontestants; i++)
{
    if(hassubmitted[i])
    {
        contestants.push_back(i);
    }
}
sort(contestants.begin(), contestants.end(), compareContestant);
cout << "\nFinal Scoreboard:\n";
for(int i = 0; i < contestants.size(); i++)
{
    int c = contestants[i];
    cout << c << " " << problemssolved[c] << " " << penalty[c] << endl;
}

```

```

    }

    return 0;
}

```

## Output:

The screenshot displays a C++ IDE with two windows. The left window shows the source code for a program named 'contestcard.cpp'. The code includes standard headers, defines constants for the number of contestants and problems, and implements a comparison function for contestants based on the number of problems solved and the penalty. The right window shows the program's execution output, which prompts the user to enter the input text size and then the details for five contestants. The output also shows the input data, the final scoreboard, and the process completion message.

```

// MOHAMMAD_ANAS_31_TY_COMP
#include <iostream>
#include <vector>
#include <algorithm>

using namespace std;

const int maxcontestants = 101;
const int maxproblems = 9;

int problemssolved[maxcontestants] = {0};
int penalty[maxcontestants] = {0};

bool hassubmitted[maxcontestants] = {0};
int incorrectattempts[maxcontestants][maxproblems] = {0};

bool compareContestant(int a, int b)
{
    if (problemssolved[a] != problemssolved[b])
        return problemssolved[a] > problemssolved[b];

    if (penalty[a] != penalty[b])
        return penalty[a] < penalty[b];
}

```

```

Enter the input Text Size: 5
Enter Contestant ID, Problem No, Time, and Judge (e.g., 1 2 10 C): 1 1 30 C
Enter Contestant ID, Problem No, Time, and Judge (e.g., 1 2 10 C): 2 2 40 I
Enter Contestant ID, Problem No, Time, and Judge (e.g., 1 2 10 C): 3 1 50 C
Enter Contestant ID, Problem No, Time, and Judge (e.g., 1 2 10 C): 2 2 60 C
Enter Contestant ID, Problem No, Time, and Judge (e.g., 1 2 10 C): 1 2 70 I

Input Data:
1 1 30 C
2 2 40 I
3 1 50 C
2 2 60 C
1 2 70 I

Final Scoreboard:
1 1 30
3 1 50
2 1 80

Process returned 0 (0x0)   execution time : 24.466 s
Press any key to continue.

```