

```

1  #include <iostream>
2  #include <vector>
3  #include <algorithm>
4  using namespace std;
5
6  struct Jersey
7  {
8      int num;
9      int sml;
10     Jersey(int num, string si)
11     {
12         this->num = num;
13         if(si == "S")
14         {
15             sml = 1;
16         }
17         else if(si == "M")
18         {
19             sml = 2;
20         }
21         else if(si == "L")
22         {
23             sml = 3;
24         }
25     }
26 };
27 int recurse (vector<Jersey> &order, vector<Jersey> &player, int numJ, int i = 0, int match = 0)
28 {
29     if(i == player.size())
30     {
31         return match;
32     }
33     for(int j = 0; j < numJ; j++)
34     {
35         if(order.empty())
36             break;
37         if(player[i].num == order[j].num)
38         {
39             if(player[i].sml <= order[j].sml)
40             {
41                 match++;
42                 order.erase(order.begin()+j, order.begin()+j+1);
43             }
44         }
45     }
46     i++;
47     recurse(order, player, numJ, i, match);
48 }
49
50 int main()
51 {
52     int numJ;
53     int numP;
54     cin >> numJ >> numP;
55     string a;
56     int b;
57     vector<Jersey> order;
58     vector<Jersey> player;
59     int match = 0;
60     for(int i = 1; i <= numJ; i++)
61     {
62         cin >> a;
63         order.emplace_back(Jersey(i, a));
64     }
65     for(int i = 0; i < numP; i++)
66     {

```

```
67         cin >> a >> b;
68         player.emplace_back(Jersey{b,a});
69     }
70     match = recurse(order,player,numJ);
71
72     cout << match;
73     return 0;
74 }
```