```
#include <iostream>
#include <vector>
#include <algorithm>
using namespace std;

struct Jersey
{
    int num;
    int sml;
    Jersey(int num, string si)
    {
        this->num = num;
        if(si == "S")
        {
            sml = 1;
        }
        else if(si == "M")
        {
            sml = 2;
        }
        else if(si == "L")
        {
            sml = 3;
        }
    }
};
int recurse (vector<Jersey> &order, vector<Jersey> &player,int numJ, int i = 0,int match = 0)
{
    if(i == player.size())
    {
        return match;
    }
    for(int j = 0; j < numJ; j++)
    {
        if(order.empty())
            break;
        if(player[i].num == order[j].num)
        {
            if(player[i].sml <= order[j].sml)
            {
                match++;
                order.erase(order.begin()+j,order.begin()+j+1);
            }
        }
    }
    i++;
    recurse(order,player,numJ,i,match);
}

int main()
{
    int numJ;
    int numP;
    cin >> numJ >> numP;
    string a;
    int b;
    vector<Jersey> order;
    vector<Jersey> player;
    int match = 0;
    for(int i = 1; i <= numJ; i++)
    {
        cin >> a;
        order.emplace_back(Jersey{i,a});
    }
    for(int i = 0; i < numP; i++)
    {
```

```cpp
        cin >> a >> b;
        player.emplace_back(Jersey{b,a});
    }
    match = recurse(order,player,numJ);

    cout << match;
    return 0;
}
```