

```

int main(){

    string input, comp, finish, reversed;

    int i, val;

    while(1){

        getline(cin,input);

        if(input.compare("DONE") == 0){

            break;

        }

        for(i=0; i<input.length(); i++){

            char x;

            x = input[i];

            if( (x != '.') && (x != ',') && (x != ' ') && (x != '!') && (x != '?')){

                comp += tolower(x);

            }

        }

        for(i=comp.length()-1; i>=0; i--){

            reversed += comp[i];

        }

        val = comp.compare(reversed);

        if(val == 0){

            cout<<"You won't be eaten!"<<endl;

        }

        else{

            cout<<"Uh oh.."<<endl;

        }

        comp = "";

        reversed= "";

    }

    return 0;

}

```

```
#include<iostream>
```

```
#include<queue>
```

```
#include<vector>
```

```
#include<functional>
```

```
using namespace std;
```

```
int main(){
```

```
    int T, N, M, aux, sumN, sumM, swap, auxM, auxN;
```

```
    priority_queue<int> Mq;
```

```
    priority_queue<int, vector<int>, greater<int>> Nq;
```

```
    cin>>T;
```

```
    while(T!=0){
```

```
        swap = 0;
```

```
        sumN = 0;
```

```
        sumM = 0;
```

```
        cin>>N>>M;
```

```
        for(int i = 0; i<N; i++){
```

```
            cin>>aux;
```

```
            Nq.push(aux);
```

```
            sumN = sumN + aux;
```

```
        }
```

```
        for(int i = 0; i<M; i++){
```

```
            cin>>aux;
```

```
            Mq.push(aux);
```

```
            sumM = sumM + aux;
```

```
        }
```

```
        while(sumN<=sumM){
```

```
            if(Nq.empty() || Mq.empty()){
```

```

        swap = -1;
        break;
    }
    auxM = Mq.top();
    auxN = Nq.top();

    Mq.pop();
    Nq.pop();
    sumN = sumN + auxM;
    sumN = sumN - auxN;
    sumM = sumM + auxN;
    sumM = sumM - auxM;
    swap++;

}
if(sumN == sumM){
    swap = -1;
}
cout<<swap<<endl;
T--;
}
return 0;
}

```