

```
///  
//NAME: FICARD Silviu  
//GRADE: 11  
//SCHOOL: NCSC "Gr. Moasil" Brasov  
//DIVISION: Sr-3
```

```
#include <iostream>  
#include <fstream>
```

```
using namespace std;
```

```
ifstream f("patolli.in");
```

```
int table[100];
```

```
struct pl{  
    int mk[3], tableVal;  
    pl(int tbv){  
        this->mk[0] = -1;  
        this->mk[1] = -1;  
        this->mk[2] = -1;  
        this->tableVal = tbv;  
    }  
    pl(int m1, int m2, int m3, int tbv){  
        this->mk[0] = m1;  
        this->mk[1] = m2;  
        this->mk[2] = m3;  
        this->tableVal = tbv;  
    }  
};
```

```
void resetTable(){
    table[0] = -1;
    for (int i = 53; i < 100; ++i) {
        table[i] = -1;
    }
    for (int i = 1; i <= 52; ++i) {
        table[i] = 0;
    }
}

bool rule9Check(int posStart, int posEnd){
    if(posStart <= 6 && posEnd > 7)
        return true;
    if(posStart <= 11 && posEnd > 12)
        return true;
    if(posStart <= 16 && posEnd > 17)
        return true;
    if(posStart <= 21 && posEnd > 22)
        return true;
    if(posStart <= 26 && posEnd > 27)
        return true;
    if(posStart <= 34 && posEnd > 35)
        return true;
    if(posStart <= 39 && posEnd > 40)
        return true;
    if(posStart <= 44 && posEnd > 45)
        return true;
    return posStart <= 49 && posEnd > 50;
```

```
}
```

```
void move(pl &player, int roll){  
    int minMarker = 100, mm = -1;  
    if(player.mk[0] < minMarker && player.mk[0] > 0)  
        minMarker = player.mk[0], mm = 0;  
    if(player.mk[1] < minMarker && player.mk[1] > 0)  
        minMarker = player.mk[1], mm = 1;  
    if(player.mk[2] < minMarker && player.mk[2] > 0)  
        minMarker = player.mk[2], mm = 2;  
    if(table[minMarker + roll] != 0){  
        return;  
    }  
    if(minMarker + roll == 52){  
        player.mk[mm] = -1;  
        table[minMarker] = 0;  
        return;  
    }  
    table[minMarker] = 0;  
    int oldPos = minMarker;  
    minMarker += roll;  
    if(minMarker == 2 || minMarker == 3 || minMarker == 5 || minMarker == 7 || minMarker == 11 ||  
minMarker == 13 || minMarker == 17 || minMarker == 19 || minMarker == 23 || minMarker == 29 ||  
minMarker == 31 || minMarker == 37 || minMarker == 41 || minMarker == 43 || minMarker == 47){  
        int positions = 6;  
        for(int i = 1; i <= 6; ++i){  
            if(table[minMarker + i] != 0){  
                positions = i - 1;  
                break;  
            }  
        }  
    }  
}
```

```

    }
}
if(minMarker + positions < 53)
    minMarker += positions;
}else if(minMarker == 9 || minMarker == 16 || minMarker == 25 || minMarker == 36 || minMarker
== 49){
    int positions = 6;
    for (int i = 1; i <= 6 && minMarker - i >= 0; ++i) {
        if(table[minMarker - i] != 0){
            positions = i - 1;
            break;
        }
    }
    if(minMarker - positions > 0)
        minMarker -= positions;
}else if(rule9Check(oldPos, minMarker) && !(minMarker == 2 || minMarker == 3 || minMarker == 5
|| minMarker == 7 || minMarker == 11 || minMarker == 13 || minMarker == 17 || minMarker == 19 ||
minMarker == 23 || minMarker == 29 || minMarker == 31 || minMarker == 37 || minMarker == 41 ||
minMarker == 43 || minMarker == 47) && !(minMarker == 9 || minMarker == 16 || minMarker == 25 ||
minMarker == 36 || minMarker == 49)){
    int position = -1;
    for(int i = oldPos + 1; i <= minMarker; ++i){
        if(table[i] == 0 && i % roll == 0)
            position = i;
    }
    if(position != -1)
        minMarker = position;
    else minMarker = oldPos;
}
table[minMarker] = player.tableVal;

```

```
    player.mk[mm] = minMarker;
}
```

```
int main() {
    for(int acsl = 0; acsl < 5; ++acsl){
        resetTable();
        pl oponent(2), player(1);
        int rolls;
        f>>oponent.mk[0]>>oponent.mk[1]>>oponent.mk[2]>>player.mk[0]>>player.mk[1]>>player.mk[2];
        if(player.mk[0] != 52)
            table[player.mk[0]] = 1;
        else player.mk[0] = -1;
        if(player.mk[1] != 52)
            table[player.mk[1]] = 1;
        else player.mk[1] = -1;
        if(player.mk[2] != 52)
            table[player.mk[2]] = 1;
        else player.mk[2] = -1;
        if(oponent.mk[0] != 52)
            table[oponent.mk[0]] = 2;
        else oponent.mk[0] = -1;
        if(oponent.mk[1] != 52)
            table[oponent.mk[1]] = 2;
        else oponent.mk[1] = -1;
        if(oponent.mk[2] != 52)
            table[oponent.mk[2]] = 2;
        else oponent.mk[2] = -1;
        f>>rolls;
        int roll;
```

```
bool plr = false;
for (int i = 0; i < rolls; ++i) {
    f>>roll;
    if(plr) move(player, roll);
    else move(oponent, roll);
    plr = !plr;
}
int s1 = 0, s2 = 0;
for (int i = 1; i < 53; ++i) {
    if(table[i] == 1){
        s1 += i;
    }else if(table[i] == 2)
        s2 += i;
}
cout<<s2<<' '<<s1<<"\n";
}
return 0;
}
```