

```

//Vaibhav Patil
//April 9th, 2020
//Int 5
//Enloe HS
//Contest #4

# include <iostream>
# include <fstream>
# include <string>
# include <vector>
# include <cmath>
using namespace std;

ifstream myfile("int.txt");
int board[52]; //1 = player, 2 = opponent
vector <int> dieRoll;

void print(){
    cout<<"1      5      10      15      20      25      30      35
40      45      50"<<endl;
    for (int i = 0; i<52; i++){
        cout<<board[i]<<' ';
    }
    cout<<endl;
}

bool isPrime(int num){
    for (int i = sqrt(num); i>=2; i--){
        if(num%i == 0)
            return false;
    }
    return true;
}

int lowest(){
    for (int i = 0; i<52; i++){
        if (board[i] == 1)
            return i;
    }
    return -1;
}

void primemove(int low, int die){
    int finalnum = low+die;

```

```

    bool e = false;
    for (int i = low+die; i<low+die+6; i++ ){
        if ((i+1>=52 || (board[i+1] != 1 && board[i+1] != 2)) && e == false)
            finalnum++;
        else
            e = true;
    }
    if (finalnum<=51){
        board[low] = 0;
        board[finalnum] = 1;
    }
}

void squaremove(int low, int die){
    int finalnum = low+die;
    bool e = false;
    for (int i = low+die; i>low+die-6; i-- ){
        if ((i-1 == low || (board[i-1] != 1 && board[i-1] != 2)) && e == false)
            finalnum--;
        else
            e = true;
    }

    board[low] = 0;
    board[finalnum] = 1;
}

bool check(int low, int die, int a, int b){
    low = low+1;
    if ( a>= low && a <= low+die && b>=low && b<=low+die)
        return true;
    else
        return false;
}
//Goes to 47
void multmove(int low, int die){
    if (check(low, die, 6, 8) || check(low, die, 11, 13) || check(low, die, 16,
18) || check(low, die, 21, 23) || check(low, die, 26, 28) || check(low, die, 34,
36) || check(low, die, 39, 41) || check(low, die, 44, 46) || check(low, die, 49,
51)){
        int finalnum = low;//11 6
        for (int i = low+1; i<=low+die; i++){
            if ((i+1)%die == 0 && board[i] == 0){
                finalnum = i;
            }
        }
    }
}

```

```

    }
    board[low] = 0;
    board[finalnum] = 1;
}
else{
    board[low] = 0;
    board[low+die] = 1;
}

}

void play(){
    for (int i = 0; i<dieroll.size(); i++){
        int low = lowest();
        if (low != -1){
            if (board[low+dieroll[i]] == 1 || board[low+dieroll[i]] == 2 ||
low+dieroll[i]+1>52){}
            else if (isPrime(low + dieroll[i] + 1)){
                primemove(low, dieroll[i]);
            }
            else if ((int)sqrt(low+dieroll[i]+1) == sqrt(low+dieroll[i]+1) &&
low+dieroll[i]+1>4){
                squaremove(low, dieroll[i]);
            }
            else if (isPrime(low + dieroll[i] + 1) == false &&
((int)sqrt(low+dieroll[i]+1) != sqrt(low+dieroll[i]+1))){
                multmove(low, dieroll[i]);
            }
            else{
                board[low] = 0;
                board[low+dieroll[i]] = 1;
            }

        }

        //cout<<"Dieroll:"<<dieroll[i]<<endl;
        //print();
        board[51] = 0;
    }
    if (lowest() == -1)
        cout<<"GAME OVER"<<endl;
    else{
        for (int i = 0; i<52; i++){
            if (board[i] == 1)
                cout<<i+1<<' ';
        }
    }
}

```

```

    }
    cout<<endl;
}

}

void input(){
    dieroll.clear();
    int playerpos, oppos, r, die;
    string temp;
    for (int i = 0; i<52; i++){
        board[i] = 0;
    }

    for (int i = 0; i<3; i++){
        myfile>>oppos;
        board[oppos-1] = 2;
    }
    for (int i = 0; i<3; i++){
        myfile>>playerpos;
        board[playerpos-1] = 1;
    }
    myfile>>r;
    for (int i = 1; i<=r; i++){
        myfile>>die;
        dieroll.push_back(die);
    }

//    print();
}

int main(){
    if (!myfile)
        cout<<"file not found"<<endl;
    for (int i = 0; i<5; i++){
        input();
        play();
    }
    //print();
    return 0;
}

```