

C++方向编程题答案

答案说明：

大家如果对本次题目或者答案有问题，可以联系下方的出题老师答疑。

出题老师：

选择题：鲍松山 qq: 365690203

代码题：鲍松山 qq: 365690203

第三周

day16

题目ID: 36880-完全数计算

链接: <https://www.nowcoder.com/practice/7299c12e6abb437c87ad3e712383ff84?tpId=37&&tqId=21279&rp=1&ru=/activity/oj&ru=/ta/huawei/question-ranking>

【题目解析】：

本题题意明确

【解题思路】：

本题可以通过遍历每一个约数，求和，判断完全数。约数计算可以遍历 \sqrt{n} 的范围。

【示例代码】

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

int Count_Of_Perfect_Number(int n)
{
    int count = 0, sum = 0;
    for(int i=2; i<=n; ++i)
    {
        for(int j=1; j<i; ++j) //效率不高
        {
            if(i % j == 0) //对约数进行求和
                sum += j;
        }
        if(sum == i)
            count++;
        sum = 0;
    }
    return count;
}

int main()
```

```

{
    int n, count;
    while(cin >> n)
    {
        count = Count_Of_Perfect_Number(n);
        cout<<count<<endl;
    }
    return 0;
}

```

题目ID:36912-扑克牌大小

链接: <https://www.nowcoder.com/practice/0a92c75f5d6b4db28cfa3e65e5c9b3f?tpId=49&&tqId=29277&rp=1&ru=/activity/oj&qru=/ta/2016test/question-ranking>

【题目解析】：

本题的题目意思是输入的只是这些类型中的一种，个子，对子，顺子（连续5张），三个，炸弹（四个）和对王。其实就是最多5张牌（顺子），最少1一张牌之间的比较。不存在其他情况。

【解题思路】：

由输入保证两手牌都是合法的，顺子已经从小到大排列，按照题意牌面类型的确定和大小的比较直接可以转换为牌个数的比较。

【示例代码】

```

#include<iostream>
#include<string>
#include<algorithm>
using namespace std;

string FindMax(const string &line)
{
    if(line.find("joker JOKER") != string::npos)
        return "joker JOKER";

    int dash = line.find('-');
    //分开两手牌
    string car1 = line.substr(0, dash);
    string car2 = line.substr(dash+1);

    //获取两手牌的张数
    int car1_cnt = count(car1.begin(), car1.end(), ' ') + 1;
    int car2_cnt = count(car2.begin(), car2.end(), ' ') + 1;

    //获取两手牌的各自第一张牌
    string car1_first = car1.substr(0, car1.find(' '));
    string car2_first = car2.substr(0, car2.find(' '));

    if(car1_cnt == car2_cnt) //两手牌的类型相同
    {

```

```
        string str = "345678910JQKA2jokerJOKER";
        if(str.find(car1_first) > str.find(car2_first))
            return car1;
        return car2;
    }

    if(car1_cnt == 4) //说明是炸弹
        return car1;
    else if(car2_cnt == 4)
        return car2;
    return "ERROR";
}

int main()
{
    string line, res;
    while(getline(cin, line))
    {
        res = FindMax(line);
        cout<<res<<endl;
    }
    return 0;
}
```