

青少年软件编程（C 语言）等级考试试卷（二级）

分数：100 题数：5

一、编程题(共 5 题，共 100 分)

1. 233 求和

题目描述

有一个序列，前几项如下：

23,233,2333,23333,...23 , 233 , 2333, 23333, \dots23,233,2333,23333,...

之后的数字以此类推。

给定正整数 n ，请输出这个序列的前 n 项和。

输入格式

单个正整数：表示 n 。

输出格式

单个正整数：表示序列的前 n 项和。

输入样例

3

输出样例

2589

说明提示

$1 \leq n \leq 17$

限制

时间限制：1000ms

内存限制：128MiB

试题编号：20250607-2-01

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：点击浏览

参考代码：

```
#include<iostream>
using namespace std;
int main(){
    long long n, a = 2, sum = 0;
    cin >> n;
    for(int i = 1; i <= n; i++){
        a = a * 10 + 3; // 递推生成当前项
        sum += a;      // 累加
    }
    cout << sum;
    return 0;
}
```

考生得分：20

是否评分：已评分

评价描述：

2. 区间的交

题目描述

一个区间可以由两个整数描述，表示为 $[a,b]$ ，意思是所有大于等于 a 且小于等于 b 的数字。

给定两个区间，第一个是 $[a,b]$ ，第二个是 $[p,q]$ ，请判断这两个区间的关系。

- 如果这两个区间没有公共的交集，输出 Disjoint
- 否则，如果一个区间完全包含另一个区间，输出 Nested
- 否则，如果两个区间有一部分内容重叠，输出 Overlap

输入格式

- 第一行：两个整数 aa 与 bb
- 第二行：两个整数 pp 与 qq

输出格式

根据区间的相交关系输出 Disjoint、Nested 或 Overlap

输入样例#1

```
1 10
2 20
```

输出样例#1

```
Overlap
```

输入样例#2

```
-1 1
10 20
```

输出样例#2

```
Disjoint
```

输入样例#3

```
-100 100
-10 10
```

输出样例#3

Nested

说明提示

$-1,000,000,000 \leq a \leq b \leq 1,000,000,000$
 $-1,000,000,000 \leq a \leq b \leq 1,000,000,000$

$-1,000,000,000 \leq p \leq q \leq 1,000,000,000$
 $-1,000,000,000 \leq p \leq q \leq 1,000,000,000$

限制

时间限制：1000ms

内存限制：128MiB

试题编号：20250607-2-02

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：点击浏览

参考代码：

```
#include<iostream>
#include<algorithm>
using namespace std;
int main(){
    long long a, b, p, q; // 数据范围大，用 long long
    cin >> a >> b >> p >> q;
    long long max_left = max(a, p);
    long long min_right = min(b, q);
    if (max_left > min_right) {
        cout << "Disjoint" << endl;
    } else {
        bool nested1 = (a <= p && q <= b); // 第一个区间包含第二个
```

```

    bool nested2 = (p <= a && b <= q); // 第二个区间包含第一个
    if (nested1 || nested2)
        cout << "Nested" << endl;
    else
        cout << "Overlap" << endl;
}
return 0;
}

```

考生得分：20

是否评分：已评分

评价描述：

3. 亲和数

题目描述

古希腊数学家毕达哥拉斯发现，某些正整数的真因子之和等于另一个数，而另一个数的真因子之和恰好就等于前者。

例如，220220220 的所有真因子之和为：

$$1+2+4+5+10+11+20+22+44+55+110=284$$

$$1+2+4+5+10+11+20+22+44+55+110=284$$

而 284284284 的所有真因子为：

$$1+2+4+71+142=220$$

$$1+2+4+71+142=220$$

人们对这样的数感到很惊奇，并称之为亲和数。一般地讲，如果两个数中任何一个数都是另一个数的真约数之和，则这两个数就是亲和数。

输入格式

输入数据第一行包含一个数 M ，接下有 M 行，每行一个实例，包含两个整数 A, B 。

输出格式

对于每个测试实例，如果 AAA 和 BBB 是亲和数的话输出 **YES**，否则输出 **NO**。

输入样例

```
2
220 284
100 200
```

输出样例

```
YES
NO
```

说明提示

$0 \leq M \leq 300 \leq M \leq 300$
 $0 \leq A, B \leq 1000000 \leq A, B \leq 1000000$

限制

时间限制：1000ms

内存限制：128MiB

试题编号：20250607-2-03

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：点击浏览

参考代码：

```
#include <iostream>
```

```

using namespace std;
// 计算真因子和（不包括自身）
int getDivisorSum(int x) {
    if (x <= 1) return 0; // 0 和 1 没有真因子
    int sum = 1; // 1 是所有大于 1 的数的真因子
    for (int i = 2; i * i <= x; i++) {
        if (x % i == 0) {
            sum += i;
            if (i != x / i) sum += x / i; // 避免重复添加平方根
        }
    }
    return sum;
}
int main() {
    int M;
    cin >> M;
    while (M--) {
        int A, B;
        cin >> A >> B;
        int sumA = getDivisorSum(A);
        int sumB = getDivisorSum(B);
        if (sumA == B && sumB == A) {
            cout << "YES" << endl;
        } else {
            cout << "NO" << endl;
        }
    }
    return 0;
}

```

考生得分：20

是否评分：已评分

评价描述：

4. 取球游戏

题目描述

一个口袋中放有若干个球，已知其中 aaa 个是红的， bbb 个是白的， ccc 个是黑的，现在从中任取 xxx 个，问有多少种颜色搭配。

输入格式

输入共两行。

第一行：三个正整数，表示 a, b, c 。

第二行：单个正整数，表示 x 。

输出格式

输出所有方案，每个方案一行，用三个正整数分别表示红、白、黑颜色球的数量，用空格隔开。按照红球个数为第一关键词，白球个数为第二关键词，黑球个数为第三关键词从小到大依次输出。

输入样例

```
3 4 2
7
```

输出样例

```
1 4 2
2 3 2
2 4 1
3 2 2
3 3 1
3 4 0
```

说明提示

$1 \leq a, b, c, x \leq 100$

限制

时间限制：1000ms

内存限制：128MiB

试题编号：20250607-2-04

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：点击浏览

参考代码：

```
#include <iostream>
using namespace std;
int main() {
    int a, b, c, x;
    cin >> a >> b >> c >> x;
    for (int i = 0; i <= a; i++) {
        for (int j = 0; j <= b; j++) {
            int k = x - i - j;
            if (k >= 0 && k <= c) {
                cout << i << " " << j << " " << k << endl;
            }
        }
    }
    return 0;
}
```

考生得分：0

是否评分：已评分

评价描述：

5. 图像相似度

题目描述

一副图像可以用 $n \times m$ 个像素点表示，每个像素点是 000

或 111，给出两幅图像，求它们的相似度。相似度定义为两幅图像在相同位置

上的像素点相等的百分比。

输入格式

第一行：两个整数 nn 和 mm ，表示一幅图像的行数和列数

接下来有 nn 行：每行 mm 个整数，表示第一幅黑白图像上各像素点的颜色。

再接下来有 nn 行：每行 mm 个整数，表示第二幅黑白图像上各像素点的颜色。

输出格式

一个浮点数，表示相似度（以百分比的形式给出），精确到小数点后两位，按照 C 语言的浮点数保留规则进行保留。

输入样例

```
3 3
1 0 1
0 0 1
1 1 0
1 1 0
0 0 1
0 0 1
```

输出样例

```
44.44
```

说明提示

$1 \leq n, m \leq 100$

限制

时间限制：1000ms

内存限制：128MiB

试题编号：20250607-2-05

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：[点击浏览](#)

参考代码：

```
#include <iostream>
#include <iomanip>
using namespace std;
int main() {
    int n, m;
    cin >> n >> m;
    int total = n * m;
    int cnt = 0;
    int matrix1[105][105], matrix2[105][105];

    // 读入第一个矩阵
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            cin >> matrix1[i][j];
        }
    }
    // 读入第二个矩阵并比较
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            cin >> matrix2[i][j];
            if (matrix1[i][j] == matrix2[i][j]) {
                cnt++;
            }
        }
    }
    // 计算并输出百分比
    double percentage = static_cast<double>(cnt) * 100 / total;
    cout << fixed << setprecision(2) << percentage << endl;
    return 0;
}
```

考生得分：0

是否评分：已评分

评价描述：