Assignment 1 C++ 基础

本次作业共4道题目，没有Challenge题。每题设置10个测试案例，其中的5个已经提供给你们用于测试。4道题共计10分。

1.温度单位转换

题目描述

输入一个华氏温度值，将其转换为摄氏温度值并输出，保留2位小数。计算公式为:

$$C = \frac{5}{9}(F - 32)$$

【注意】为了保证结果的一致性，请在计算的过程中采用`double`类型作为每一步浮点数运算的类型。输出估计值时统一采用2位小数，请在程序开始处引入`iomanip`头文件:

```
#include <iomanip>
```

并在输出时使用`fixed`和`setprecision`函数来选择格式，例如:

```
cout << fixed << setprecision(2) << output << endl;
```

输入输出格式及示例

输入和输出均为单个浮点数。

例如:

输入

```
99.9
```

输出

```
37.72
```

输入

```
2025
```

输出

```
1107.22
```

数据范围

`0 < n < 1000000`

2. 时间转换器

题目描述

输入一个正整数n代表总分钟数，将其转换为“天”、“小时“、和"分钟"的表达形式。如果任意一项为0时则不需要输出对应项。

输入输出格式

输入格式为一个正整数n,输出的基本格式为形如`xdyhzm`的字符串，`x,y,z`分别代表天数、小时数和分钟数。当某一项为0时不需要输出，参见下面的示例。

输入

```
61
```

输出

```
1h1m
```

输入

```
1450
```

输出

```
1d10m
```

输入

```
1440
```

输出

```
1d
```

数据范围

对于所有的输入`n`，`0 < n < 1000000`

3. 辗转相除法

辗转相除法又称欧几里得算法，是求两个数的最大公约数的经典算法。

这一算法是一个典型的递归算法，其计算过程为：

首先不妨设$x_1 > y_1$ 为两个输入数字，以较大的x_1为被除数，y_1为除数计算除法：

$x_1/y_1 = z_1 \ldots m_1$

其中z_1为商，m_1为余数。如果能够整除（即余数$m_1 = 0$） 则y_1 即为初始输入x_1和y_1的最大公因数。

否则令$x_2 = y_1, y_2 = m_1$, 继续计算

$x_2/y_2 = z_2 \ldots m_2$

再对m_2进行相同的判断，如果不能整除则令$x_{k+1} = y_k, y_{k+1} = m_k$。

如此不断计算直至m_k 为0，此时的除数y_k 即为辗转相除法求得的最大公因数。

本题要求实现辗转相除法并将计算过程呈现为从y_1 到最终答案y_k的序列，注意其中y_1 **应该是`m,n`中较小的数**，y_k 为`m`和`n`的最大公因数。

输入输出格式

输入为两个正整数`m,n`，用空格隔开。输出为一连串用空格隔开的正整数$y_1 \ldots y_k$。示例如下

输入

```
18 48
```

输出

```
18 12 6
```

输入

```
37 13
```

输出

```
13 11 2 1
```

数据范围

`0 < m,n < 1000000`

4. 素因数求和

任何正整数都能够被分解为一系列素数的乘积，这样的因数分解结果是唯一的，被称为素因数分解。 本题需要你判断两个正整数的的素因数**求和**结果是否相同。

例如`60 = 2*2*3*5`，`2+2+3+5 = 12`；`81 = 3*3*3*3`，`3+3+3+3 = 12`.那么我们称`60`和`81`的素因数之和是相等的。

输入输出格式

输入为两个正整数`m,n`，输出为“Yes”或者“No”.

输入

```
5 7
```

输出

```
No
```

输入

```
81 60
```

输出

```
Yes
```

数据范围

`1 < m,n <1000000`

提交格式

在编写完全部程序并测试完毕后，**将4个文件夹中除了main.cpp以外的文件删除之后一起打包压缩**，最后用自己的学号命名，即可提交。

你提交的文件结构应该和以下形式**完全一样**:

```
<your student number>.zip
|- 1_temperature
|  |- main.cpp
|
|- 2_time
|  |- main.cpp
|
|- 3_euclid
|  |- main.cpp
|
|- 4_sum
|  |- main.cpp
```