

Problem A. 中秋节快乐

题目描述

虽然并没有到中秋节，但作为一道快乐的签到题，还是预祝大家中秋节快乐。

输入格式

无输入

输出格式

一行，一个字符串“Happy Mid-Autumn Festival !”（不包含引号）。

样例输入

无

样例输出

```
Happy Mid-Autumn Festival !
```

Hint

```
#include<stdio.h>
int main(){
    printf("Happy Mid-Autumn Festival !");
    return 0;
}
```

Problem B. AC

题目背景

作为快乐学习C语言的开始，这次的题目都很简单，预祝同学们轻松AC。

输入格式

无

输出格式

AC字符画，见样例输出。

样例输入

样例输出

Problem C. Relu函数

题目背景

题目描述

$$f(x) = \begin{cases} 0, & x \leq 0, \\ x, & x > 0. \end{cases}$$

输入格式

输出格式

样例输入1

样例输出1

样例输入2

-1

样例输出2

0

Hint

使用 `scanf` 函数输入整数，使用 `printf` 函数输出整数，使用 `if` 语句进行条件判断。

AUTHOR: StyWang

Problem D. 生日

题目背景

有一只蒟蒻忘记了自己的生日，因此只要你输入三个数字，他就只会开始复读“My birthday is xxxx.xx.xx”。

输入格式

一行，三个用空格隔开的整数 y, m, d ，分别对应年、月、日。

输出格式

一行，一个字符串“My birthday is xxxx.xx.xx”，注：当月、日不足两位时，需要在首位补 0。

样例输入

2000 5 16

样例输出

My birthday is 2000.05.16

数据范围

$1000 \leq y \leq 9999, 1 \leq m \leq 12, 1 \leq d \leq 28$ 。

Hint

你当然可以使用 `if else` 语句解决此问题，但或许有更简洁的办法。

Prblem E. 真·计算机

题目描述

WYJ小时候只会用计算器。有一天他偶然打开计算机，希望水掉小学算术题，结果双击了某游戏图标，从此走上不归路.....现在是时候实现WYJ儿时的梦想了！

WYJ每次会给你两个非负整数 a 和非负整数 b ，表示作运算的两个数字。WYJ希望你写一个程序，一口气把这两个数字作加、减、乘、整除以及取模的结果（总共五个结果）依次求出来，这样他什么算术题都不怕了。

输入格式

一行，两个由空格隔开的非负整数 a, b 。

输出格式

b 不为0时，共五行，每行一个整数，分别表示两个数字进行加、减、乘、**整除**和取模的结果。整除和取模时， a 为被除数， b 为除数。其中，整除是指两数相除结果取整数部分（如 $5/3 = 1$ ）；取模是指取两数相除的余数。

当除数 b 为0时，请在对应位置输出一行：“[Warning] division by zero [-Wdiv-by-zero]”，不含双引号，且整除和取模运算**总共输出一次**。

样例输入1

```
5 2
```

样例输出1

```
7
3
10
2
1
```

样例输入2

```
5 0
```

样例输出2

```
5
5
0
[warning] division by zero [-wdiv-by-zero]
```

数据范围

$$0 \leq a, b \leq 10^4$$

Hint

C语言里 $/$ 为整除操作； $\%$ 为取模，即相除取余数的操作。

Problem F. 输出倒三角

题目描述

给定一个除空格外的可见字符，用该字符绘制一个底边长度为5的倒三角。

输入格式

一行，一个非空格的可见字符。

输出格式

一个由输入字符组成的底边长为5倒三角。

样例输入

```
a
```

样例输出

```
aaaaa
aaa
a
```

Hint

可以使用以下方式输出n个*号：

```
while(n--)  
    printf("*");
```

当然不使用循环也可以解决这道题。

AUTHOR: YUKILSY

Problem G. 三角形的周长

题目背景

有一只蒟蒻在平面直角坐标系里发现了三个点，他想要知道这三个点组成的三角形的周长。

输入格式

三行，每行两个浮点数，分别表示三角形的三个顶点的坐标。

输出格式

一行，一个浮点数，表示三角形的周长（保留两位小数）。

样例输入

```
0 0
0 3
4 0
```

样例输出

```
12.00
```

数据范围

对于任意一个顶点的坐标 (x, y) , 有 $-10^5 \leq x, y \leq 10^5$ 。

Hint

- 保证输入的三个点不共线。
- 浮点数可使用double类型。
- 可以使用scanf("%lf%lf",&x,&y);读入两个浮点数, 使用printf("%.2f",d);的方式输出一个浮点数并保留两位小数。
- #include<math.h>中sqrt函数可以进行开根号计算, 如sqrt(x)表示 \sqrt{x} 。

Problem H. 打印进度条

题目描述

StyWang 正在认真阅读《程序设计基础》课程的课件, 他希望你根据他当前阅读的进度, 打印出以下形状的进度条:

```
|#####-----
-----|
```

这份课件共有100页, StyWang 会输入他正在阅读的页码 $n(n \in [1, 100])$, 请你按照以下规则输出进度条:

1. 进度条只有1行, 共102列, 其前、后各有一个竖线: |
2. 两个竖线之间含有 n 个警号: #
3. 在警号与右侧竖线之间用短横线 (减号) 将进度条补全为102列: -

有时 StyWang 会在梦中不小心输入闭区间 $[1, 100]$ 之外的整数, 这时请你输出 wake up! 叫醒他。

输入格式

输入一个 int 范围内的整数 n 。

输出格式

输出如题所示的进度条, 或叫醒信息。

样例输入1

样例输出1

```
|#####-----  
-----|
```

样例输入2

-1

样例输出2

wake up!

Hint

你需要输出多少个短横线（减号）？

Author: StyWang

Problem I. 有向立方和

题目描述

给出一组整数 $\{a_1, a_2, \dots, a_n\}$ ，输出其有向立方和，其中这组数前 n 项有向立方和 S_n 定义为：

- 当 $n = 1$ 时：

$$S_1 = \begin{cases} -a_1^3, & \text{若 } a_1 \text{ 为奇数} \\ a_1^3, & \text{若 } a_1 \text{ 为偶数} \end{cases}$$

- 当 $n > 1$ 时：

$$S_n = \begin{cases} S_{n-1} - a_n^3, & \text{若 } a_n \text{ 为奇数} \\ S_{n-1} + a_n^3, & \text{若 } a_n \text{ 为偶数} \end{cases}$$

输入格式

输入包含任意 n 行，每行一个整数 a_i 。

输出格式

一行，一个整数，表示输入中所有整数 $\{a_1, a_2, \dots, a_n\}$ 的有向立方和 S_n 。

样例输入1

3
4
5

样例输出1

-88

样例输入2

-3
-4
-5

样例输出2

88

样例解释

- 样例1: $-3^3 + 4^3 - 5^3 = -88$
- 样例2: $-(-3)^3 + (-4)^3 - (-5)^3 = 27 - 64 + 125 = 88$

Hint

- 对于60%的测试用例, 满足 $1 < n \leq 100$, 对于任意 $1 \leq i \leq n$ 有 $0 < a_i \leq 20$ 。
对于100%的测试用例, 满足 $1 < n \leq 1000$, 对于任意 $1 \leq i \leq n$ 有 $0 < a_i \leq 1000$ 。
- 若预先不知道输入数字的个数, 可以采取以下的方式进行读入:

```
int n;  
while (scanf("%d", &n) != EOF) {  
    // do something  
}
```

- 本题的答案可能会超出 `int` 的范围, 或许你需要了解一种叫long long的类型。

Problem J. 零花钱

题目描述

因为小函做错了事, 妈妈决定惩罚下小函——更改给小函的零花钱方案。第一天, 给1元钱; 之后两天, 每天给2元钱; 之后三天, 每天给3元钱;; 这种零花钱方案会一直延续下去。但是当零花钱达到15元时就不会再增加! (即之后每天都是15元零花钱)

请计算在前k天里(包括第k天), 小函一共获得了多少零花钱。

注:所有数据均在 `int` 范围内。

输入格式

一个非负整数 k ，表示发零花钱的天数。

输出格式

一个整数，表示小函收到的零花钱。

样例输入1

6

样例输出1

14

样例输入2

121

样例输出2

1255

样例解释

见题面描述

Hint

可以利用数学方法做题~

AUTHOR: yyh