

1.关于评测机

- 首先建议大家仔细阅读C1简介，了解OJ平台的使用方法以及WA,TLE等错误提示。

<http://10.251.254.83:3000/contest-ng/index.html#/715>

- 目前OJ平台配置较低，提交评测后需要平均等待30min才能返回结果，期间显示为waiting状态，请大家在写完程序后自行构造一些测试数据，尽量确保代码在本地运行无误后再提交。

2.关于本地测数据

- 可以点击运行界面（就是那个黑框）左上角，再点击“属性”，在“编辑选项”中勾选“快速编辑模式”。此时在运行界面中单击右键，即可将剪切板内容粘贴到运行界面中。也就意味着大家可以先将样例复制，然后直接粘贴过来，而不用自己手动输入。





3.scanf函数

- scanf函数可以实现按格式“从标准输入（一般为控制台）输入”的功能。其中**按格式**指的是，可以输入不同类型、不同数量的数据，并存入变量。

```
scanf("%d", a);    // 错误，无取址符  
scanf("%d, a");    // 错误，双引号不要括住逗号后边的内容  
scanf("%d\n", &a); // 错误，字符串中的'\n'可能导致输入无法正常结束  
scanf("%d", &a);  // 正确
```

- 其中的取地址符 & 是必须的，但在使用printf函数输出时不能用。该运算符的相关知识之后会介绍。

4.多组输入

- 多组数据输入时，可以使用以下代码：

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

- 该代码表示，不停执行 `scanf("%d", &n)`，直到到达文件末尾(EOF, End Of File)。

在Windows下执行此代码，输入结束时，可以按下Ctrl + Z并输入一个回车，为程序提供“到达文件末尾”信号，使程序跳出该while循环从而正常结束。

- 同学们可以尝试执行以下代码并观察输出：

```
int n;
while (scanf("%d", &n) != EOF)
{
    printf("%d\n", n);
}
```

- 可以发现，大括号内部的代码会对**每个输入的数据**均执行一遍。

练习

- 编写以下程序：输入多组数据，每组数据包含两个整数，并针对每组数据的两个整数，输出它们的和。
- 程序如下：

```
#include <stdio.h>
int main()
{
    int a, b;
    while (scanf("%d%d", &a, &b) != EOF)
    {
        printf("%d\n", a + b);
    }
}
```

5.printf函数

- 该函数的格式为：`printf(双引号包围的字符串, 变量列表);`
- 使用时需要注意：
 - 不要和scanf混淆，这里的变量前不加取地址符&
 - 该函数将双引号内部的内容输出到标准输出（一般是控制台）上，其中的`%d`，`%f`，`%02d`等会被变量列表中提供的**变量的值**替换，并按格式输出，其余内容**原样输出**。
- 输出多行数据时，要使用“换行符”，即`\n`，而非`\n\\`。例如：

```
// 优美的写法：将变量a,b,c,d的值分行输出
printf("%d\n%d\n%d\n%d", a, b, c, d);
```

- 不要使用`\n\\`+代码中的换行来完成输出换行操作，以下代码是不推荐的：

```
// 不推荐的写法，与上述代码等价
printf("%d\n\
%d\n\
%d\n\
%d", a, b, c, d);
```

参考文章: <http://c.biancheng.net/cpp/html/293.html>

6.if-else

- 建议的书写格式:

```
if(<statement>){
    //do something
}else{
    //do something
}
```

- 如果这样写:

```
if(<statement>)
    a = 0;
    b = 1;
```

- 则 `b = 1` 会无条件执行，因为if后若不添加大括号，则仅匹配下方的第一条语句。else同理。
- 关于 `=` 与 `==`：`=` 是赋值运算，而 `==` 是条件判断，不要混淆。
- 一组if-else语句基本格式如下:

```
if(条件1)
{
    //语句1
}
else if(条件2)
{
    //语句2
}
else if(条件3)
{
    //语句3
}
//此处可加入更多else if语句
else
{
    //语句4
}
```

(其中"else if"语句块和"else"语句块都可以省略)

这段代码表示:

若【条件1】满足，则执行【语句1】，**不执行语句2，语句3，和语句4**

否则（即不满足条件1），若【条件2】满足，则执行【语句2】，**不执行语句3，和语句4**

否则（即不满足条件1和2），若【条件3】满足，则执行【语句3】，**不执行语句4**

否则（即三个条件都不满足），执行语句4。

当某个大括号内的语句（语句之间以分号为界）只有一句时**可以**省略大括号以增加美观性，若有多句则不可省略。

请大家观察并运行以下代码以更好理解if-else语句(当然要放入 `int main()` 中才可运行)。

```
//第1个代码
int a = 15;
if(a >= 10)
    printf("a is greater than 10");
else if(a >= 15)
    printf("a is greater than 15");
```

```
//第2个代码
int a = 15;
if(a >= 10)
    printf("a is greater than 10");
if(a >= 15)
    printf("a is greater than 15");
```

```
//第3个代码
int a = 10;
if(a >= 10)
    printf("a is greater than 10");
else if(a >= 15)
    printf("a is greater than 15");
printf("a is good");//其实此行代码【不应该】缩进，便于理解代码
```

```
//第4个代码
int a = 10;
if(a >= 10)
    printf("a is greater than 10");
else if(a >= 15)
{
    printf("a is greater than 15");
    printf("a is good");
}
```

7.关于题目中所给数据范围

题目中可能会给类似于 $1000 \leq y \leq 9999$ 的数据范围。这表示在系统运行你的程序时，系统所输入的数据在这个范围区间内。也就是说，不用在写程序时另行判断这个条件。（可以理解为出题人与做题人之间的一种约定：出题人保证数据满足一定的条件）。

但这个条件可能会影响你写程序。例如输出 $a + b$ 的值，当 $1 \leq a, b \leq 10^9$ 时，使用int类型的a,b即可。但如果 $1 \leq a, b \leq 10^{18}$ 时，务必要使用long long类型的a,b。

8.其他

- 对于题面上可以**复制粘贴**的字符串，应当避免手打。
- 建议在编译时加上 `-Wall` (dev-c++在工具-编译选项-编译时加入以下命令里)，如图



这个命令会针对诸如 `scanf("%d",a)`, `if(a=0)` ,变量没初始化, 局部变量覆盖全局变量等低级错误进行报错 (warning) , 可以极大地减少同学们犯低级错误的概率。

- `scanf` 读入数据时不要加空格, `\n` 之类的字符, 只写 `%d,%c,%lf` 这种就可以了。
- 涉及到 EOF 的问题, 严格来说属于超纲了。关于 EOF ,在本地终端运行测试时可以使用 `ctrl+z` 来模拟 EOF
- 养成良好的代码风格, 包括但不限于: **变量命名符合变量语义**、**变量、常量、运算符前后用空格分开**、**正确地缩进代码**等, 这对程序的阅读与调试都尤为重要。
- 请多多练习, 充分的练习可以使你避免许多不必要的问题。

9.最后

- 请大家在上机前做好复习, 并准备好课本与PPT, 以供上机时参考。
- 代码没有通过编译或出现错误, 请**不要第一时间提问**, 要养成自己排查错误的能力。
- 当感到题目难度较高时, 可以参考每道题目的**通过率与正确率**, 先做简单题。
- 在线上提问时, 建议不要直接粘贴代码, 良好的提问应当给出问题的描述与自己大致的猜想。
- 本次练习赛涉及的超纲知识点确实有点多, 大家如果成绩不理想的话不要气馁, 不要说什么“看别人都一直在切题, 我啥也不会”之类的话。下来以后要认真阅读题解和总结, 总结自己不会的知识点。即使是零基础的同学, 把程设学得很好也完全不是问题。
- 这里给大家推荐一些知识获取途径。
 - 课程, 课件, 老师, 助教, OJ平台, 这是学校给大家提供的资源, 一定要利用好。
 - https://gitee.com/Blessures/Programming-Fundamentals-Course_2021C
 - 上面是答疑平台的链接, 请有效使用。
- CSDN论坛或搜索引擎: 基本所有东西都可以在上面找到, 关键要会描述清楚问题。
- 洛谷: 一个做题网站, 如果觉得课程提供的题目不够的话, 可以做做这个题单里的题<https://www.luogu.com.cn/training/list>
- 更多的内容可以参考这篇文章: <https://www.cnblogs.com/BUAA-Wander/p/14161831.html> by 某大佬学长。

