

编程作业: 编程作业—文件操作与模板

您还未提交解答。您必须获得 72/90 分才能通过。

截止时间 在以下日期前通过此作业 五月 21, 11:59 晚上 PDT

说明
我提交的作业
讨论

准备

在开始下面的作业前，请先[点击这里](#)下载代码模版。

编程题 # 1

来源: POJ (Coursera声明：在POJ上完成的习题将不会计入Coursera的最后成绩。)

注意： 总时间限制: **1000ms** 内存限制: **65536kB**

描述

实现一个三维数组模版CArray3D，可以用来生成元素为任意类型变量的三维数组，使得下面程序输出结果是：

0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,

注意，只能写一个类模版，不能写多个。

```
1  #include <iostream>
2  using namespace std;
3  // 在此处补充你的代码
4  int main()
5  {
6      CArray3D<int> a(3,4,5);
7      int No = 0;
8      for( int i = 0; i < 3; ++ i )
9          for( int j = 0; j < 4; ++j )
10             for( int k = 0; k < 5; ++k )
11                 a[i][j][k] = No ++;
12      for( int i = 0; i < 3; ++ i )
13          for( int j = 0; j < 4; ++j )
14             for( int k = 0; k < 5; ++k )
15                 cout << a[i][j][k] << ", ";
16      return 0;
17  }
```

输入

无

输出

0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,

样例输入

```
1  无
```

样例输出

```
1 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28
   ,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52
   ,53,54,55,56,57,58,59,
```

提示

提示：类里面可以定义类，类模版里面也可以定义类模版。例如：

```
1 class A
2 {
3     class B {
4
5     };
6 };
7
8 template
9 class S
10 {
11     T x;
12     class K {
13         T a;
14     };
15 };
```

编程题 # 2： 实数的输出格式

来源: POJ (Coursera声明：在POJ上完成的习题将不会计入Coursera的最后成绩。)

注意： 总时间限制: **1000ms** 内存限制: **1000kB**

描述

利用流操纵算子实现： 输入一个实数，先以非科学计数法输出，小数点后面保留5位有效数字；再以科学计数法输出，小数点后面保留7位有效数字。

注意：在不同系统、编译器上的输出格式略有不同，但保证在程序中采用默认格式设置一定能在OJ平台上得到正确结果。

输入

以非科学计数法表示的一个正实数，保证可以用double类型存储。

输出

第一行：以非科学计数法输出该实数，小数点后面保留5位有效数字；

第二行：以科学计数法输出该实数，小数点后面保留7位有效数字。

样例输入

```
1 12.34
```

样例输出

```
1 12.34000
2 1.2340000e+01
```

编程题 # 3： 整数的输出格式

来源: POJ (Coursera声明：在POJ上完成的习题将不会计入Coursera的最后成绩。)

注意： 总时间限制: **1000ms** 内存限制: **1000kB**

描述

利用流操纵算子实现： 输入一个整数，先将该整数以十六进制输出，然后再将该整数以10个字符的宽度输出，宽度不足时在左边补0。

注意：在不同系统、编译器上的输出格式略有不同，但保证在程序中采用默认格式设置一定能在OJ平台上得到正确结果。

输入

一个正整数，保证可以用int类型存储。

输出

第一行：以十六进制输出该整数；

第二行：以10个字符的宽度输出该整数。

样例输入

```
1 23
```

样例输出

```
1 17
2 0000000023
```

编程题 # 4： 字符串操作

来源: POJ (Coursera声明：在POJ上完成的习题将不会计入Coursera的最后成绩。)

注意： 总时间限制: **1000ms** 内存限制: **65536kB**

描述

给定n个字符串（从1开始编号），每个字符串中的字符位置从0开始编号，长度为1-500，现有如下若干操作：

copy N X L：取出第N个字符串第X个字符开始的长度为L的字符串。

add S1 S2：判断S1，S2是否为0-99999之间的整数，若是则将其转化为整数做加法，若不是，则作字符串加法，返回的值为一字符串。

find S N：在第N个字符串中从左开始找寻S字符串，返回其第一次出现的位置，若没有找到，返回字符串的长度。

rfind S N：在第N个字符串中从右开始找寻S字符串，返回其第一次出现的位置，若没有找到，返回字符串的长度。

insert S N X：在第N个字符串的第X个字符位置中插入S字符串。

reset S N：将第N个字符串变为S。

print N：打印输出第N个字符串。

printall：打印输出所有字符串。

over：结束操作。

其中N，X，L可由find与rfind操作表达式构成，S，S1，S2可由copy与add操作表达式构成。

输入

第一行为一个整数n（n在1-20之间）

接下来n行为n个字符串，字符串不包含空格及操作命令等。

接下来若干行为一系列操作，直到over结束。

输出

根据操作提示输出对应字符串。

样例输入

```
1 3
2 329strjvc
3 0padfk48
4 Ifjoqwoqejr
5 insert copy 1 find 2 1 2 2 2
6 print 2
7 reset add copy 1 find 3 1 3 copy 2 find 2 2 2 3
8 print 3
9 insert a 3 2
10 printall
11 over
```

样例输出

```
1 0p29adfk48
2 358
3 329strjvc
4 0p29adfk48
5 35a8
```

提示

推荐使用string类中的相关操作函数。

How to submit

When you're ready to submit, you can upload files for each part of the assignment on the "My submission" tab.

