

编程作业: 编程作业一类和对象

您还未提交解答。您必须获得 80/100 分才能通过。

截止时间 作业截止日期为 四月 23, 11:59 晚上 PDT
您仍可以在课程结束前完成此作业。

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

下面程序输出的结果是：

0

5

请填空：

```

1  #include <iostream>
2  using namespace std;
3  class A {
4  public:
5      int val;
6      // 在此处补充你的代码
7  };
8  main() {
9      A a;
10     cout << a.val << endl;
11     a.GetObj() = 5;
12     cout << a.val << endl;
13 }

```

输入

无

输出

0

5

样例输入

1 无

样例输出

1 0
2 5

提示

所缺代码具有如下形式：

```

1      A(_____){ val = n; }
2      _____ GetObj() {
3          return _____;
4      }

```

编程题 # 2

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

下面程序的输出是：

10

请补足Sample类的成员函数。不能增加成员变量。

```
1  #include <iostream>
2  using namespace std;
3  class Sample{
4  public:
5      int v;
6      Sample(int n):v(n) { }
7  // 在此处补充你的代码
8  };
9  int main() {
10     Sample a(5);
11     Sample b = a;
12     cout << b.v;
13     return 0;
14 }
```

输入

无

输出

10

样例输入

1 无

样例输出

1 10

编程题 # 3

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

下面程序的输出结果是：

5,5

5,5

请填空：

```
1  #include <iostream>
2  using namespace std;
3  class Base {
4  public:
5      int k;
6      Base(int n):k(n) { }
7  };
8  class Big {
9  public:
10     int v; Base b;
11     // 在此处补充你的代码
12 };
13 int main() {
14     Big a1(5); Big a2 = a1;
15     cout << a1.v << "," << a1.b.k << endl;
16     cout << a2.v << "," << a2.b.k << endl;
17     return 0;
18 }
```

输入

无

输出

5,5

5,5

样例输入

1 无

样例输出

1 5,5
2 5,5

提示

所缺代码具有如下形式：

```
1      Big -----{ }
2      Big -----{ }
```

编程题 # 4: 魔兽世界之一: 备战

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

魔兽世界的西面是红魔军的司令部, 东面是蓝魔军的司令部。两个司令部之间是依次排列的若干城市。

红司令部, City 1, City 2,, City n, 蓝司令部

两军的司令部都会制造武士。武士一共有 dragon、ninja、iceman、lion、wolf 五种。每种武士都有编号、生命值、攻击力这三种属性。

双方的武士编号都是从1开始计算。红方制造出来的第n个武士, 编号就是n。同样, 蓝方制造出来的第n个武士, 编号也是n。

武士在刚降生的时候有一个生命值。

在每个整点, 双方的司令部中各有一个武士降生。

红方司令部按照iceman、lion、wolf、ninja、dragon的顺序循环制造武士。

蓝方司令部按照lion、dragon、ninja、iceman、wolf的顺序循环制造武士。

制造武士需要生命元。

制造一个初始生命值为m的武士, 司令部中的生命元就要减少m个。

如果司令部中的生命元不足以制造某个按顺序应该制造的武士, 那么司令部就试图制造下一个。如果所有武士都不能制造了, 则司令部停止制造武士。

给定一个时间, 和双方司令部的初始生命元数目, 要求你将从0点0分开始到双方司令部停止制造武士为止的所有事件按顺序输出。

一共有两种事件, 其对应的输出样例如下:

1) 武士降生

输出样例: 004 blue lion 5 born with strength 5,2 lion in red headquarter

表示在4点整, 编号为5的蓝魔lion武士降生, 它降生时生命值为5, 降生后蓝魔司令部里共有2个lion武士。(为简单起见, 不考虑单词的复数形式) 注意, 每制造出一个新的武士, 都要输出此时司令部里共有多少个该种武士。

2) 司令部停止制造武士

输出样例： 010 red headquarter stops making warriors

表示在10点整，红方司令部停止制造武士

输出事件时：

首先按时间顺序输出；

同一时间发生的事件，先输出红司令部的，再输出蓝司令部的。

输入

第一行是一个整数，代表测试数据组数。

每组测试数据共两行。

第一行：一个整数M。其含义为， 每个司令部一开始都有M个生命元($1 \leq M \leq 10000$)。

第二行：五个整数，依次是 dragon 、ninja、iceman、lion、wolf 的初始生命值。它们都大于0小于等于10000。

输出

对每组测试数据，要求输出从0时0分开始，到双方司令部都停止制造武士为止的所有事件。

对每组测试数据，首先输出"Case:n" n是测试数据的编号，从1开始 。

接下来按恰当的顺序和格式输出所有事件。每个事件都以事件发生的时间开头，时间以小时为单位，有三位。

样例输入

```
1 1
2 20
3 3 4 5 6 7
```

样例输出

```
1 Case:1
2 000 red iceman 1 born with strength 5,1 iceman in red
   headquarter
3 000 blue lion 1 born with strength 6,1 lion in blue
   headquarter
4 001 red lion 2 born with strength 6,1 lion in red
   headquarter
5 001 blue dragon 2 born with strength 3,1 dragon in blue
   headquarter
6 002 red wolf 3 born with strength 7,1 wolf in red
   headquarter
7 002 blue ninja 3 born with strength 4,1 ninja in blue
   headquarter
8 003 red headquarter stops making warriors
9 003 blue iceman 4 born with strength 5,1 iceman in blue
   headquarter
10 004 blue headquarter stops making warriors
```

How to submit

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

