# 答题目标：

**正确性要求：**

作为QA部未来的新童鞋，在代码能力方面我们对各位要求的并非是解题的思路以及代码的编写技巧，关键是要让大家在写代码的过程中学会如何保证所产出代码的质量；本轮考核我们给各位新同学设计了四道必做题，融合了自学计划里面的C#基础一级和C#基础二级的知识点，**各位需要在指定的时间范围内通过四道题方为完成本轮考核任务**；

C#基础一级的知识点主要通过前三道题目来进行考核，这三道题目的难度相对比较基础，但是**各位要多考虑各种可能的边界情况或是极限情况**，增强所写的接口的效率、容错等方面的质量；避免判题用例中的一些特殊用例弄垮你们所编写的功能接口；（这些也是未来你能把工作做好提高自身职能影响力的重要技能）。

C#基础二级的知识点则通过第四道题目来进行考核；作为C#二级的考评，考核的内容也是比较基础的，我们预设的目标是考查各位：**通过C#这个语言工具来理解面向对象的建模思维**，所以题目会更偏向于对类库的理解、调用方面知识点的考查；希望能促进各位入职后开展相关的测试和开发工作时，除了有一个“**实现**”的思维，还能有一个“**设计**”的思维来看待和解决相关的复杂问题。

**效率要求：**

代码的执行是有时限控制的，目前的设置是每个用例的运行时间最长不能超过2秒，若超时将被中断执行，该题会被判为不通过；所以切记要考虑极限情况以及提高运行效率；

**自测要求：**

首先，**节操为上，请勿抄袭**！（\* **一旦发现抄袭，视为违约处理** \*）。另外，本轮考核共有4道必做题，每道1分，而测评成绩排序将按照：**得分总数**>**提交次数**>**最后提交时间** 三个数据进行排序，所以要获得好的名次除了正确性之外，提交次数也要注意控制好；在提交前强烈建议各位先进行自测，预先设计一些用例来对接口进行校验，用测试来驱动开发提高代码质量；

# 答题方法：

**提交方式**：把邮件发送到指定的邮箱【[**qaexam2014@163.com**】](mailto:qaexam2014@163.com】)

**标题格式**：**[QAExam]CS基础考核\_李小明**

**附件要求**：附件附上一个c#答案代码档（无c#文档的邮件会被直接忽略，大小超过32K也直接忽略），具体的代码接口参看下面的题目章节或者代码答案文件模板，各位在代码模板文件中补充代码逻辑，充分测试后作为附件提交即可；

**代码接口定义**：参看代码答案模板文件**QAExam\_CS\_2014.cs**，命名空间为**qaExam2014**，各位在函数体内补充代码即可；（建议使用.net framework4.0 / vs2010开发）

**判题结果反馈：**判题系统会每隔5分钟检查一次提交到邮箱的代码邮件，并会在判题之后返回没有通过用例的接口的判题情况，各位可以根据提示进行代码的修正和完善；当然，若通过了所有的用例，也会给各位反馈。判题系统有下列可能的反馈情况：

1. 代码编译出错
2. 没有找到指定接口（提交的代码中没有实现指定的功能接口）
3. 运行时崩溃（譬如：堆栈等资源申请过度等情况）
4. 运行时出现异常
5. 运行超时
6. 接口运行的输出结果和期望值不符
7. 全部通过

**注意要点：**

* 为了减少提交的次数，各位可以在保证正确性的前提下，一次提交多道题目的接口；
* 若某次提交的多道题目中，有部分通过用例检测的，正确的题目会被记录，分数被刷新；后面提交这道题与否或者后续这道题是否正确都不影响结果；
* 每道题目的函数**接口必须严格按照接口定义进行编写**，否则判题时会认为接口未实现而直接跳过；
* 判题器有严格的安全限制，代码中若有访问服务器资源与网络资源的操作将会导致代码运行时除错而被中断，该接口会被判为不通过；
* 基础题目都比较简单，所以判题器在代码编译时只会默认加载System.dll引用库，以及用于第四题的类库Level\_2\_Lib.dll，其他引用库不会加载，若using不在以上引用库的命名空间将会引发编译错误。

## 必做题一：【强大的攻击值】

游戏中的数值系数会随着等级或者装备的提升而不断增大，如果一开始没有规划好，将会导致数值计算超出程序语言所能支持的最大精度而溢出；譬如一个高阶玩家的攻击数值是“865987453”，在战斗中触发了强力暴击，攻击系数提升73倍，那算起来就超过了32bit的数值精度；现在为了全面解决这类问题，程序员已经全部把计算接口做了改善增加了对大数据的支持；现在你需要简单地写段大数据乘法的代码去测试程序员的改动是否正确。

===============================

【参数】：2个字符串参数（a，b）分别是两个用半角字符串表示的实数(只包括正负号、小数点和数字，且题目保证单个参数的字符长度<=50)；

比如：参数为**-865987453.36589**，**73.23658**）,代表将要乘的两个乘数是**-865987453.36589**，**73.23658**；

【返回】：全精度计算结果（半角字符串，碰到小数位末尾有0请截去，如198.00 应该返回 198）；

上述例子中的参数将对应输出字符串：**-63421959407.4272722562**;

## 必做题二：【夺命机关】

X游戏的策划小明想到了一个休闲的机关玩法并把玩法文档交给了你做审核；他的玩法要点如下：怪物和玩家所操控的角色都被限定在一段距离为S的直线路径上面且初始时分据两个端点，游戏开始后，无论怪物还是玩家角色都会自动沿着直线路径匀速移动（速度分别为V1和V2，且开始时是相向而行），当他们到达路径的端点后，会马上按原速度折返继续匀速移动；怪物和玩家相遇时，玩家可以跳起通过踩踏的方式攻击怪物，若未能及时跳起怪物将会发起攻击从而造成玩家角色血量的下降。看完小明的这个策划文档，对游戏数值异常敏感的你马上发现，要预估玩家在时间T之内是否能通关，很有必要算出在时间T之内，怪物和玩家角色相遇的次数，于是你准备写一个小程序用来算这个相遇次数。

===============================

【参数】：4个int参数（**s，v1，v2，t**）分别是：路径长度（米）、怪物速度（米/秒）、角色速度（米/秒）、运动的时间（秒）；如：（**2,1,1,1**）指的是：在2米的路径长度下，怪物和玩家角色都以1米/s的速度相向而行，要求1秒钟他们会相遇多少次？

【返回】：怪物和玩家角色的相遇次数（int），如上述输入参数，将会输出：**1**，代表怪物和玩家角色将会相遇**1**次。

## 必做题三：【卡牌的组合技】

C游戏是一个卡牌类游戏，玩家通过战斗或抽牌可以拿到一些技能牌，每张技能牌都有对应的伤害值（伤害值>=0），当你有了组合技属性之后，你可以在自己手头上选择任意张技能牌，以组合技的方式来攻击boss，组合技的总伤害将等于所组合的各张技能牌的伤害值的乘积(只有一张牌时，组合技伤害值等于这张牌本身的伤害值)，但是能发动组合技必须有个前提：所有被选择的技能牌的伤害系数之和必须等于m（m>0）以解开封印；你为了能赢得最终PK，需要在所有技能牌中挑出若干张技能牌触发组合技（每张牌只能用一次），以形成最大威力的组合技攻击效果。

============================

【参数】：

t: 你所拥有的n张技能牌 的伤害系数数组；（如：[1,2,3,4,5,6]，降低难度，牌的数量n<=20张）

m: 组合技需要凑的伤害数之和;(如：7)

【返回】：返回的是一个int整数（如：**12**），代表能解开封印（即：和等于m）且能发出最大攻击的卡牌伤害值组合数组（如：[3,4]）所触发的组合技总伤害值；若没有符合要求的卡牌组合，请输出int整数：**-1。**

## 必做题四：【挖宝机器人】

T游戏里面有个迷宫挖宝的玩法，每盘会载入不同的随机迷宫地图，让玩家竞（hu）争（keng），最快幸存到达地图中投放点的玩家将能成功拿到所投放的宝物；由于这个玩法的宝物投放力度较大，每天都吸引了很多玩家来玩；可近来由于项目组对底层代码进行大规模的优化重构，导致迷宫刷不出宝物的事故偶有发生。负责这个玩法质量的你觉得不能再踏入同一个陷阱了，为了检测已有正确功能不被相关代码的修改影响，你决定把这个迷宫挖宝的功能放到每天的日常回归测试列表中去，实现一个自动挖宝的机器人，让机器人每天晚上自动尝试挖宝，一旦挖宝失败，你便可提前发现问题，避免外放时影响玩家。

你把想法跟程序员介绍之后，程序员非常支持，给了挖宝核心接口基类库以及对应的说明文档给你，你于是便着手去实现挖宝机器人了。

============================

【任务要点】：

1. 阅读基类库的文档(CS\_Exam\_2014\_Level\_2\_Lib.chm)，了解挖宝机器人原型的可用操控方式及可用接口；
2. 在项目中引入附带的基类库(Level\_2\_Lib.dll),并在代码中继承此基类，并重写实现autoRun方法，完成自动挖宝的功能；
3. 为了降低难度，本题用于判题的迷宫地图都是平面地图，且尺寸限定在700\*700范围内，虽然路点数量有限，但是敬请各位注意代码的执行效率，2秒内都无法探寻到目的投放点，将被判为不通过；另外判题器分派的堆栈空间有限，若要使用递归解法请自行审慎优化，避免堆栈溢出或超时而被判为不通过。
4. 每当调用run函数，机器人都会按照设定好的运动方向及预设距离来尝试运动，然后会在停下来的时候广播委托事件，为了获得反馈信息你可以订阅此委托事件，注册到事件的方法将会在run调用后实现回调，回调的参数包含了实际的前行距离、方向及停止的原因（跑完/碰壁/找到投放点），具体的接口和定义可以通过VS自带的对象浏览器浏览DLL或阅读类库文档来了解；
5. 为了方便各位自测，基类默认提供了一个公共的样板用例接口（sampleTestCase），并预设在Main入口进行调用，调用此接口将能调试基本的寻宝功能，强烈建议各位在通过了样板用例后再行提交。