**考试说明**

1. **考试时间：合计180分钟，选择题部分不超过60分钟，允许提前交卷。**
2. **考试过程中，不能连接未经指定网站或服务器。**
3. **闭卷考试部分，不能查阅任何类型的参考资料。**
4. **开卷考试部分，可以查阅纸质文档，不能查阅编程环境自带帮助文件以外的任何类型的电子文档。**
5. **考试过程中，不得使用任何形式的电子存储设备，不可使用手机。**
6. **违反上述2-5条者，视为考试作弊。**

**选择题答题方式（20分，闭卷，自动阅卷，严禁使用编程环境进行尝试）**

1. **打开浏览器，在地址栏中输入**http://47.103.208.54/mytest/index.htm**，点击相应链接进入登录页面。**
2. **按要求输入两遍自己的考号。**
3. **点击“登录”按钮即可进入答题页面。如考试尚未开始，系统会进入等待页面并倒计时。考试开始时间到，系统会自动进入答题页面。**
4. **在页面左侧选择题号，页面右侧即会显示相应的题目。考生只需点击选择相应的选项。**
5. **答题过程中如关闭浏览器或出现系统故障导致计算机重新启动，系统不会丢失之前已经完成的题目的答案。考生可以打开浏览器重新登录并继续考试。**
6. **答题完成后，点击“交卷”按钮即可完成交卷。交卷后不能再次登录系统继续考试。**
7. **考试结束时间到，系统会自动收卷。**

**编程题注意事项与提交方式（80分，开卷，自动阅卷）**

1. **根据选课语种下载sample.py、samply.c、samply.cpp、samply.java（以下统称sample文件）中的一个文件，保存到本地硬盘。**
2. **根据题目要求补全sample文件中的空函数。sample文件中的已有函数的函数名、参数数量和顺序不可以修改。**
3. **不要在代码中调用从键盘接受输入的函数，如：input()、scanf()等，否则会导致阅卷失败。如阅卷失败，编程题部分不得分。**
4. **每个函数中不得使用显示输出函数输出结果，如：print()、printf()等。函数须使用return返回结果。如用显示输出函数输出结果将导致该函数不得分。**
5. **代码中不要出现任何中文。**
6. **提交的时候，确保程序可以正常运行，不要遗留任何语法错误。语法错误、程序异常等将导致阅卷失败。如阅卷失败，编程题部分不得分。**
7. **所有脚本程序内容必须仅包含在一个源程序文件中。**
8. **提交前务必关闭编程环境。**
9. **考试过程中，请经常进行存盘操作，以防突发异常情况而导致程序未保存。**
10. **在浏览器的地址栏中输入**http://47.103.208.54/mytest/index.htm**，点击相应链接进入提交页面。**
11. **按要求输入两遍自己的考号。**
12. **点击“选择文件”按钮，选择自己的脚本程序文件。点击“提交”按钮提交。**
13. **如提交成功，系统会显示相关信息。如果提交不成功，请重复步骤23-25。**
14. **提交成功后，可点击“查看内容”按钮检查提交的内容。**
15. **编程题部分由阅卷系统自动批阅，以运行正确的测试用例数计分。不采取人工阅卷的方式。**

**题目说明**

1. 给定一个整数序列的字符串，求整数的构成数字中含给定数字且重复次数最多的那个整数构成的字符串。

|  |  |
| --- | --- |
| 相关说明 | |
| 输入条件 | * 参数lst是字符串，它由若干以逗号间隔的整数构成（>=0个），其中最后一个整数为给定的0-9之间的某个数字。 * 参数合法，无需额外的检查。 |
| 输出要求 | * 如果没有元素包含指定数字，则返回特殊字符串“-1”。 * 如果只有一个元素出现次数最多，则返回由该整数构成的字符串。 * 若序列中包含重复值，则仅统计首次出现的整数。 * 如果出现次数最多的整数有多个，则返回这些整数构成的字符串序列，该序列中各整数字符串以逗号间隔，保持这些整数的原有次序。 |
| 其它要求 | 将代码写入函数getMostFrequent |

测试用例：

|  |  |
| --- | --- |
| 输入 | 返回 |
| “60, 36, 1, 49, 135, 35, 2” | “-1” |
| “60, 36, 235, 122, 3, 17, 235, 2” | “122” |
| “69, 369, 703,12, 4332, 69, 4332, 91, 6, 3” | “4332” |
| “15, 229, 692, 93, 23, 221, 2” | “229, 221” |
| “95, 95, 39, 125, 125, 23, 105, 150, 5” | “95, 125, 105,150” |
| “9” | “-1” |

1. 给定一个无序的正整数序列，并且序列中整数个数大于4个。已知该序列如果按照升序排序后是中间缺少一个整数的二阶等差数列，请找出缺少的那个整数。所谓二阶等差数列就是相邻两个整数的差构成的序列为等差数列，例如序列：2，5，10，17，26；其相邻元素的差序列为：3，5，7，9构成等差数列。

|  |  |
| --- | --- |
| 相关说明 | |
| 输入条件 | * 无序的正整数序列 * 参数合法，无需额外的检查。 |
| 输出要求 | * 返回缺少的整数 |
| 其它要求 | 将代码写入函数getTrgNum |

测试用例：

|  |  |
| --- | --- |
| 输入 | 返回 |
| 1, 3, 6, 15, 21 | 10 |
| 1, 4,7,11,16 | 2 |
| 1,2,4,7,11, 22 | 16 |

1. 给定一个只包含字母和空格的字符串，字符串是一些由空格分隔的单词，其中单词之间的空格个数大于等于1个，且单词的最大长度不超过12个字母。请返回这些单词中单词权重最大的单词的权重值。其中单词权重定义为单词各字母的ASCII值之和的均值。例如：“am”的ASCII值之和为65+109=174，则“am”的权重为：174/2=87.

|  |  |
| --- | --- |
| 相关说明 | |
| 输入条件 | * 一个有空格分隔的单词字符串 * 参数合法，无需额外的检查。 |
| 输出要求 | * 返回单词序列中的最大权重值 |
| 其它要求 | 将代码写入函数getMostWeight |

测试用例：

|  |  |
| --- | --- |
| 输入 | 返回 |
| “This is a test” | 112.0 |
| “This is a python program” | 112.33333333333333 |
| “I love program” | 109.5 |

1. 鸡兔同笼是否有解？参数heads、feet分别代表鸡兔同笼问题中头和脚的数量，判断是否构成一个有效的鸡兔同笼题目。

|  |  |
| --- | --- |
| 相关说明 | |
| 输入条件 | * 参数heads、feet是两个整数，分别表示鸡兔同笼问题中头和脚的数量。 * 输入参数确定是正整数，不用考虑非正整数的情况。 |
| 输出要求 | * 如果有解输出True * 否则输出False |
| 其它要求 | 将代码写入函数isSameCage。 |

测试用例：

|  |  |  |
| --- | --- | --- |
| 输入 | 返回 | 说明 |
| 35,94 | True | 23只鸡，12只兔 |
| 2,4 | True | 2只鸡 |
| 4,2 | False | 无解 |
| 10,30 | True | 5只鸡、5只兔 |

### 5、单链表结点重排

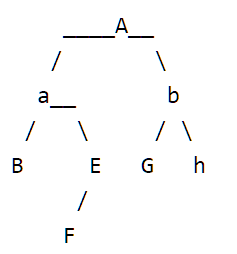
1. 设带头结点的整数单链表包含n个元素结点，设其中间位置为m，即原单链表可以表示为：a0 -> a1 -> ... -> am -> ...-> an-2 -> an-1。
2. 设计算法，重新调整单链表中的结点次序，将原链表调整为： a0 -> am+1 -> a1 -> am+2 -> .... 这样的顺序。
3. 例如：链表原值结点为1->2->3->4->5->6，则调整后链表为1->4->2->5->3->6。
4. 又例如：链表原值结点为1->2->3->4->5，则调整后链表依次为1->4->2->5->3。
5. 请完成源程序文件中的**re\_arrange()**算法，注意：可以添加其他算法供**re\_arrange**调用，但不可修改其他程序代码。

测试用例：

|  |  |
| --- | --- |
| 输入 | 返回 |
| 11,12,13,14,15,16 | 11 14 12 15 13 16 |
| 1,2,3,4,5,6 | 1 4 2 5 3 6 |
| 34,35,37,29,12 | 34 29 35 12 37 |

### 6、二叉树给定结点与离它最近的叶子之间的路径长度

1. 设计算法，求二叉树中值为x的结点到以它为根的子树上离它最近的叶子之间的路径长度。
2. 二叉树采用二叉链表作为存储结构。测试代码中假设二叉树的结点个数不超过20，结点值为互不相同的单个字符。源程序文件中已通过带#号的先序序列字符串创建了二叉树的二叉链表，请完成其中的closestleaf()算法。
3. 注意：可以添加其他算法供closestleaf调用，但不可修改其他程序代码。



例如上图所示二叉树中，与a最近的叶子是B，a到B的路径长度为1，与A最近的叶子是B、G或h，路径长度为2。

测试用例：

|  |  |
| --- | --- |
| 输入 | 返回 |
| AaB##EF###bG##h##  a  A | 1  2 |

1. 现给一排栋房子的外墙涂色，共有种颜色可以使用。为了美观，要求每栋房子和它相邻的房子具有不同的颜色，并且任意连续4栋房子至少包含3种颜色。求一共有多少种涂色方案。

测试用例：

|  |  |
| --- | --- |
| 输入 | 返回 |
| 4,3 | 18 |
| 5,6 | 3480 |
| 8,10 | 45158400 |

1. 篮球比赛中有1分球、2分球和3分球三种进球得分。给定最终得分，有多少种可能的得分组合()，即个1分球，个2分球和个3分球。

测试用例：

|  |  |
| --- | --- |
| 输入 | 返回 |
| 15 | 27 |
| 5 | 5 |
| 25 | 65 |
| 0 | 0 |