实验六 分支与循环

**实验目的**

（一）掌握分支语句的应用。

（二）掌握循环语句的使用。

（三）掌握使用分支语句和循环语句解决简单问题。

（四）掌握next语句和break语句的应用。

请完成实验一、二、四、五、六、七题，实验三与实验八请在课后完成。

**实验内容**

**实验一**：if语句与if-else语句的应用。打开脚本文件test0701.R，完成下面操作。

⑴ 若n是偶数，则用print函数显示“n是偶数！”，否则不显示。

⑵ 若m是偶数，则用print函数显示“m是偶数！”，否则显示“m是奇数！”。

**实验二**：多路分支的应用。打开脚本文件test0702.R，完成下面操作。

假设score表示分数，根据score判断等级，90分及以上为优秀，80分至89分为良好，70分至79分为中等，60分至69分为及格，60分以下为不及格。

**实验三**：分支语句的应用。打开脚本文件test0703.R，用分支语句完成下面操作。

⑴ 以a, b, c为边长，判断是否可以构成一个三角形。

⑵ x, y, z是三个任意的整数，用分支语句找出这三个数中的最大值，并用cat函数显示x, y, z三个数及其最大值。

**实验四**：分支函数的应用。打开脚本文件test0704.R，完成下面操作。

⑴ 用ifelse函数产生一个新向量over40，向量的元素由“是”和“否”构成。产生规则为：对应于age中的元素，若大于40，则为“是”，否则为“否”。

⑵ 用ifelse函数找出x, y, z中的最大值，并用cat函数显示x, y, z三个数及其最大值。

**实验五**：循环语句的应用。打开脚本文件test0705.R，用循环语句完成下面操作。

⑴ 用for循环计算并输出向量x的所有元素之和，并用sum函数验证计算结果的正确性。

⑵ 用while循环计算并输出向量x的所有元素之和。

⑶ 用repeat-break循环计算并输出向量x的所有元素之和。

**实验六**：循环语句与switch函数的应用。打开脚本文件test0706.R，完成下面操作。

依据向量z的内容，用循环语句修改数据框deg列，修改规则为：若z中的元素为1，则deg列对应位置修改为“本科”；若为2，则修改为“硕士”；若为3，则修改为“博士”。

**实验七**：next语句和break语句的应用。打开脚本文件test0707.R，完成下面操作。

⑴ 计算向量x中大于0的元素之和，且遇到大于30的元素时，立即停止计算，要求使用next语句和break语句控制程序流程。

⑵ 用print函数显示计算结果（注：参考结果为696.87）。

**实验八**：循环语句的应用。打开脚本文件test0708.R，完成下面操作。

⑴ 用循环计算计算向量x和向量y的相关系数，其定义如下：



其中，为向量x中的元素，为向量y中的元素，和为向量x和y的均值，可用mean函数计算，公式中求和运算（）要求用循环语句实现。

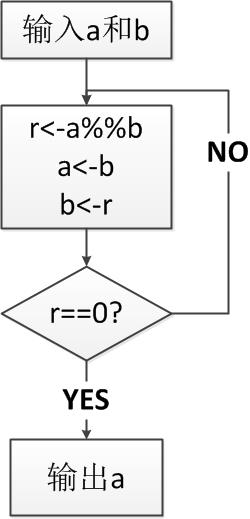
⑵ 用print函数显示结果，并与cor(x, y)的返回结果对比。

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

以下实验为课后思考题

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

**实验九**：算法实验一。打开脚本文件test0709.R，实现欧几里德算法求两个正整数的最大公约数，其中欧几里德算法的流程如下图所示。



**实验十**：算法实验二。打开脚本文件test0710.R，使用冒泡算法对向量x进行排序。

⑴ 算法简要描述

x = c(5, 6, 3, 4, 1, 2)，由小到大排序，排序过程与体育课由矮到高排队相似。

▪ 第1趟：第6个与第5个比较，小的移到第5的位置，再与第4个比较，…，最小的那个（数字1）就冒泡到第1的位置。

▪ 第2趟：剩下的5个数（第2个到第6个），重复上述过程，这样第2小的数就排到了正确位置。

▪ 第3趟：剩下的4个数（第3个到第6个），重复上述过程，这样第3小的数排到了正确位置。

▪ 第4趟：剩下的3个数（第4个到第6个），重复上述过程，这样第4小的数排到了正确位置。

▪ 第5趟：剩下的2个数（第5个和第6个），重复上述过程，这样第5小的数排到了正确位置。

▪ 最后只剩下1个数了，就不用再比较了。

⑵ 总结

▪ n个数字只需要n-1趟循环。

▪ 每趟都从最后那个开始，与前面的比较，若后面的小，则交换。因此，每趟需要做循环，且第k趟循环的最后那次比较，是第k+1个元素与第k个元素比较。

⑶ 伪代码

n <- x的长度

k <- 从1到(n-1)

j <- 从n到(k+1)

if(x[j]<x[j-1]){交换x[j]与x[j-1]}

下一个j

下一个k

输出x