实验八 自定义函数

**实验目的**

（一）掌握R函数的定义方法。

（二）掌握R运算符的定义方法。

（三）掌握自定义函数与运算符的简单应用。

（四）掌握递归函数的定义方法。

**实验内容**

**实验题1**：自定义函数。打开脚本文件test0801.R，完成下面操作。

⑴ 创建自定义函数count\_even，函数的输入参数为整数型向量，输出为向量中偶数的数目。

⑵ 首先需要对参数进行检查，检查内容包括：

* 是否缺失参数，若缺失，则终止调用，并给出错误信息“没有输入被检测向量！”
* 检查数据对象，若不是向量，则终止调用，并给出错误信息“输入数据对象不是向量！”
* 检查数据类型，若不是数值型数据，则返回0，并给出警告信息“输入数据不是数值型数据！”

⑶ 要求用循环语句实现计算。

⑷ 运行脚本程序后面的测试代码，观察结果是否正确。

**实验题2**：自定义运算符。打开脚本文件test0802.R，完成下面操作。

⑴ 自定义运算符%xor%，计算0与1的异或操作，其运算规则如下：

1 %xor% 1 = 0, 0 %xor% 0 = 0

1 %xor% 0 = 1, 0 %xor% 1 = 1

⑵ 若输入数据不是0或1，则返回NA。

⑶ 运行脚本程序后面的测试代码，观察结果是否满足题目要求。

**实验题3**：自定义函数及其参数的默认值。打开脚本文件test0803.R，完成下面操作。

⑴ 编写一个自定义函数matrix\_compute，该函数的输入参数为矩阵mat和标识参数flag。根据flag值，对矩阵进行计算，计算规则为：

* 若flag = 1，则计算mat中每行元素的平方和；
* 若flag = 2，则计算mat中每列元素的平方和；
* 若flag = 3，则计算mat中所有元素的平方和。
* 若flag为其它值，则返回NA

⑵ 将标识参数flag的默认值设置为3。

⑶ 运行脚本程序后面的测试代码，观察结果是否正确。

**实验题4**：省略号参数的使用。打开脚本文件test0804.R，完成下面操作。

⑴ 自定义函数stat\_compute，该函数有两个输入参数：

* 第一个参数fun是字符串，其值可设置为'sum'、'median'、'mean'、'var'，分别表示求和、求中位数、求平均值、求方差；
* 第二个参数为省略号参数。

⑵ 该函数根据第一个参数的意义，计算省略号参数所代表的全体元素的和、中位数、平均值或方差。

⑶ 运行脚本程序后面的测试代码，验证结果是否正确。

**实验题5**：自定义函数的应用1。打开脚本文件test0805.R，完成下面操作。

⑴ 定义函数myfuture(P,r,y,t)，该函数用于计算银行存款的本利之和，计算公式如下



其中是本利之和，是本金，是年利率（比如说是2.7，它表示年利率是2.7%），表示计息周期（，表示每月计息一次），表示存储的年数。

⑵ 参数分别为、、、，其中的默认值为2.7；若缺失参数，则终止运行，并给出错误信息"未输入存款的年数"；若缺失参数，则赋值为12，并给出警告信息"参数被设置为12"。

⑶ 用Excel的FV函数验证结果的正确性。

**实验题6**：自定义函数的应用 2。打开脚本文件test0806.R，完成下面操作。

⑴ 定义函数myvar，输入参数为向量x，函数的返回值为x的方差，计算公式为



其中为向量x的长度，为向量x中的第个元素。

⑵ 要求用循环语句实现（不能使用var函数或cor函数），可使用mean计算。

⑶ 运行脚本程序后面的测试代码，验证结果是否正确。

----------------------------------------------------------------------------------------------------------------

以下请课后完成

--------------------------------------------------------------------------------------------------------------

**实验题7**：自定义函数的应用3。打开脚本文件test0807.R，完成下面操作。

⑴ 定义函数bin2dec(sbin)，其功能是将二进制字符串转换成十进制整数，其中输入参数sbin是0和1构成的字符串，返回值为对应的整数，例如，输入为"101011"，输出为43)。

⑵ 需要对参数进行检查，若包含非0与非1字符，则停止运行，给出错误信息"输入包含了非0-1字符串"。提示：用正则表达式和grepl函数。

⑶ 转换规则为提取字符串中每个字符，然后表示为



其中为字符串中的第个字符（从右向左数），例如，"101011"的十进制整数为



⑷ 算法过程分析

* 假设初始累加器，那么





* 每次循环先乘以2，然后加上新提取的字符（从左向右提取）。

提示：用substr或substring函数提取字符。

**实验题8**：自定义函数的应用4。打开脚本文件test0808.R，完成下面操作。

⑴ 定义函数dec2bin(x)，其中x是正整数，函数的返回值为x的二进制数字符串（0-1字符串），例如，dec2bin(3)的结果是"10"，dec2bin(10)的结果是"1010"。

⑵ 转换方法是“求余数，自下向上”。例如，13的转换过程为：

* 13除以2的余数是1，商是6，得到第一个余数是**1；**
* 6（上一步的商）除以2的余数是0，商是3，得到第二个余数是**0**；
* 3 （上一步的商）除以2的余数是1，商是1，得到第三个余数是**1**；
* 1 （上一步的商）除以2的余数是1，商是0，得到第四个余数是**1**；
* 上一步的商是0，则停止，自下向上取余数，得到输出"1101"

提示：用ceiling(log2(x+1))计算二进制位数，想想为什么？

**实验题9**：简单递归函数。打开脚本文件test0809.R，完成下面操作。

⑴ 定义函数mysum(n)，其中n是大于1整数，函数返回



的值。

⑵ 要求用递归完成（不用循环语句）。

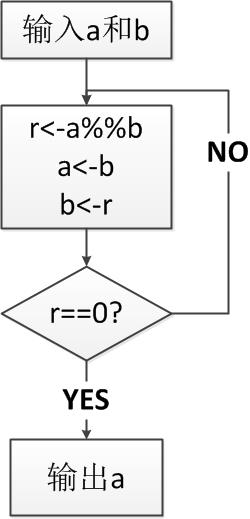
⑶ 用mysum(100)测试，其结果是否为5050。

**实验题10**：递归函数的应用1。打开脚本文件test0810.R，完成下面操作。

定义函数myrecur(x,n)，该函数返回x中第n个数字（从右向左算），其中参数x和n是正整数，例如myfun(123456, 3)的结果是4， myfun (123456, 6)的结果是1， myfun(123456,7)的结果是0。

**实验题11**：递归函数的应用2。打开脚本文件test0811.R，完成下面操作。

⑴ 定义函数mygcd(a,b)，该函数用欧几里德算法计算两整数a和b的最大公约数，流程图如下图所示。



⑵ 与上周的实验题9比较，请说明递归的优势。