

# 深圳大学实验报告

课程名称： 现代程序设计

实验项目名称： 字符串与正则表达式

学院： 电子与信息工程学院

专业：

指导教师： 邹文斌

报告人：  学号：  班级：

实验时间： 2023 年 10 月 23 日

实验报告提交时间： 2023 年 10 月 24 日

教务部制

## 一、 实验要求

- a) 字符串常用操作方法
- b) 算法
- c) 需上交实验报告、py 源程序。

## 二、 实验环境

Python IDLE, Pycharm 等

## 三、 实验内容

### 1. 循环赛

n 支球队进行循环赛，每一支球队都要和其他 n-1 支球队进行一场比赛。  
赢得比赛可以得 3 分，输掉比赛要扣 1 分，平局两支球队都不得分，也不扣分。现在给你所有比赛场次的输赢结果，请你输出所有球队中的最高得分。

#### 输入

第一行一个正整数 n ( $1 \leq n \leq 20$ ), 表示参加比赛的球队数量。  
第二行 n 个单词，表示 n 支球队的名字，相邻两个单词之间用一个空格隔开。  
后面  $n*(n-1)/2$  行，表示所有比赛场次的结果，每一行表示一场比赛。每一行分成三部分，前两部分为两个字符串，为两支球队的名字，第三部分为一个数字，表示比赛结果，1 表示前面的球队胜，2 表示后面的球队胜，3 表示平局。三部分之间用一个空格隔开。

#### 输出

一个整数，表示最高得分。

#### 输入样例 1

```
4
Fire Wolf Sun Pigeon
Sun Fire 1
Pigeon Fire 1
Sun Pigeon 2
Pigeon Wolf 3
Fire Wolf 3
Sun Wolf 3
```

#### 输出样例 1

```
6
```

### 2. 时间换算

将秒数转换为与其等价的小时数、分钟数、秒数的表现形式。  
小时数、分钟数、秒数的数字之间以英文冒号(:)分隔，且数字小于 10 的，需要在数字前补 0。  
例如：500 秒等同于 8 分钟 20 秒，表现形式为：00:08:20;7259 秒等同于 2 小时 0 分钟 59 秒，表现形式为：02:00:59。

#### 输入描述：

输入一个正整数  $N$  ( $N < 86400$ ) 作为秒数。

**输出描述:**

输出与其对应的小时数、分钟数、秒数, 数字之间以英文冒号分隔。(数字小于 10 的, 需要在数字前补 0)

**样例输入:**

7259

**样例输出:**

02:00:59

### 3. 计算 24

“计算 24” 是一个流传已久的数字游戏, 小蓝最近对此痴迷不已。

游戏规则是: 从 1~10 之间的自然数任意拿出 4 个数 (4 个数各不相同, 顺序随机), 进行加、减、乘三种运算 (使用某种运算的次数、种类不限), 要求运算结果等于 24。乘法的优先级高于加、减, 并且算式中不可以用括号, 不可以改变 4 个数字出现的顺序。以下给出两个游戏的具体例子:

若给出的 4 个数是: 10、2、4、8, 则有两种解答方案 ( $10+2+4+8=24$ ,  $10*2-4+8=24$ ), 则输出 2

若给出的 4 个数是: 7、2、3、6, 则有零种解答方案, 则输出 0

**输入描述:**

输入四个 1 到 10 之间的正整数并以一个空格隔开 (四个正整数各不相同)

**输出描述:**

输出有多少种运算方案的结果为 24

**样例输入:**

10 2 4 8

**样例输出:**

2

### 4. 最佳策略 (A+挑战题)

有  $n$  个小朋友排成一排, 现在需要按身高从低到高的顺序进行排列。排序方式为: 如果位置相邻的两个小朋友不符合从低到高的顺序, 就交换这两个小朋友的位置。且每个小朋友都有一个不高兴的数值, 开始的时候, 所有小朋友的不高兴值为 0。如果某个小朋友第一次被交换, 则他的不高兴值加 1, 如果第二次被交换, 则他的不高兴值加 2, 如果第三次被交换, 则他的不高兴值加 3, 依此类推。

假如: 一个小朋友被交换了 3 次, 他的不高兴值为  $6(1+2+3)$ 。

如果让所有小朋友都按从低到高的顺序排好队, 那么所有小朋友的不高兴值的总和的最小值是多少 (也就是交换次数最少, 不高兴值得总和最小)。

**注意:**

1. 如果有两个小朋友身高一样, 谁在谁前无所谓 (不需要交换);
2. 每次交换的两个小朋友都需要增加不高兴值。

**输入描述:**

第一行输入一个正整数  $n$  ( $2 < n < 51$ ) 表示小朋友的数量

第二行输入  $n$  个正整数 (每个正整数  $< 160$ ), 分别表示  $n$  个小朋友的身高, 正整数之间以一个空格隔开

**输出描述:**

输出所有小朋友的不高兴值的总和的最小值。

样例输入:

3

130 115 98

样例输出:

9

#### 5. 病毒繁殖 (A+挑战题)

某种病毒具有很强的繁殖能力, 从病毒粒子出生后的第 5 分钟开始, 每分钟可以复制出一个新的病毒粒子。新出生的病毒粒子从第 5 分钟开始, 也可以每分钟复制一个新的病毒粒子。

举例来说, 第 1 分钟时有一个病毒粒子, 此病毒粒子从第 5 分钟开始复制新的病毒粒子, 因此第 5 分钟时的病毒数量为 2 个; 第 6 分钟时又复制出新的病毒粒子, 因此第 6 分钟的病毒数量为 3 个; 以此类推, 第 7 分钟时病毒粒子数为 4; 第 8 分钟时病毒粒子数为 5; 第 9 分钟时, 第 5 分钟复制出的病毒粒子开始复制新的病毒粒子, 因此第 9 分钟时的病毒总数为 7; 第 10 分钟时, 第 6 分钟复制出的病毒粒子开始复制新的病毒粒子, 因此第 10 分钟时的病毒粒子总数为 10。

编程实现:

计算病毒粒子总数, 已知第一分钟时出生了一个病毒粒子, 假设所有病毒粒子不会自动死亡, 请计算第  $N$  分钟时的病毒粒子总数。

例如: 前 10 分钟病毒粒子的总数分别为 1, 1, 1, 1, 2, 3, 4, 5, 7, 10。

输入描述:

输入正整数  $N$  ( $0 < N \leq 60$ ), 表示时间

输出描述:

输出第  $N$  分钟时, 病毒粒子的总数

样例输入:

6

样例输出:

3

## 四、实验过程

**思路:**

(涉及到算法实现的实验需阐述算法的逻辑关系)

### 一、循环赛

1. 获得球队总数量及各队名称, 将它们保存在一个字典中。
2. 初始化各个球队的得分。

3. 根据输入读取各场次比赛的结果，利用 if 语句判断给各队加减分数。
4. 读取字典中各个球队的键值，计算出得分最高的一项并输出。

## 二、时间换算

1. 获取输入的秒数。
2. 将秒数对 3600 取整除计算出小时，对 3600 取余再对 60 取整除计算得出分钟、对 3600 取余再对 60 取余计算出秒。
3. 将计算出的数据转化为字符串格式，并对于小于 10 的数在前面补 0。
4. 将数据按期望输出的格式排列并输出结果。

## 三、计算 24

1. 定义一个计算器函数，并初始化计数器为 0。
2. 计算器在加、减、乘法中遍历，分别构建 4 个整数计算的算式。
3. 计算算式的结果并与 24 进行比对，如果结果等于 24，则令计数器加一。
4. 获取用户输入并调用计算器，计算输出结果。

## 五、病毒繁殖

1. 设置一个递归函数，表示第 N 分钟时的病毒粒子总数。
2. 当 N 小于 4 时，病毒粒子总数为 1；当 N 大于等于 5 时，第 N 分钟的病毒粒子总数等于前一分钟的病毒粒子总数加上第 N-4 分钟的病毒粒子总数。
3. 获取输入，调用递归函数计算并输出结果。

### 完整代码:

（必须有详细的注释）

#### 一、循环赛 round\_robin.py

```
n = int(input('请输入球队数量:'))

teams = input('请输入各球队的名字，以空格分隔:').split()

#以一个字典保存球队名称

score = {}
```

```
#初始化各球队的分数

for team in teams:

    score[team] = 0


#获得各场次的比赛结果

print('请输入比赛的两个队伍名称及比赛结果\n1 表示第一个队伍赢，2 表示第二个队伍赢，3 表示平局')

for i in range(n * (n - 1) // 2):

    team1,team2,result = input('请输入各场比赛的队伍以及结果，以空格分隔队伍名称及结果:\n').split()

    result = int(result)#将输入转化为整型，方便后续比对操作


    #当前一队赢得比赛时

    if result == 1:

        score[team1] += 3

        score[team2] -= 1

    #当后一队赢得比赛时

    elif result == 2:

        score[team2] += 3

        score[team1] -= 1

    #当平局时

    else:

        score[team1] += 1

        score[team2] += 1


#计算出字典中保存的队伍里分数最高的一项

max_score = max(score.values())

print('最高得分为:',max_score)
```

## 二、时间换算 time.py

```
N = int(input('请输入一个正整数N 作为总秒数(N<86400):'))
```

```
#分别计算时/分/秒
```

```
hours = N // 3600
```

```
minutes = (N % 3600) // 60
```

```
seconds = (N % 3600) % 60
```

```
#将数据转化为字符串格式，并对于小于 10 的数在前面补 0
```

```
h = str(hours).zfill(2)
```

```
m = str(minutes).zfill(2)
```

```
s = str(seconds).zfill(2)
```

```
#将数据保存为期望输出的格式
```

```
time = f"{h}:{m}:{s}"
```

```
print('该秒数转换为时间格式为:',time)
```

### 三、计算 24 calculate24.py

```
def calculate(nums):
```

```
    #初始化计数器
```

```
    count = 0
```

```
    #在加法/减法/乘法中遍历
```

```
    for cal1 in ['+', '-', '*']:
```

```
        for cal2 in ['+', '-', '*']:
```

```
            for cal3 in ['+', '-', '*']:
```

```
                #构建算式
```

```
                expression
```

```
=f"{nums[0]} {cal1} {nums[1]} {cal2} {nums[2]} {cal3} {nums[3]}"
```

```
                #计算算式的结果
```

```
                result = eval(expression)
```

```

        #将结果与 24 比对，若结果为 24，计数器加一
        if result == 24:
            count += 1

    return count

#获得用户输入并将数据保存在列表中
nums = list(map(int,input('请输入四个 1 到 10 之间的各不相同的数字，以空格
分隔:').split()))

#调用计算函数得出结果
result = calculate(nums)
print(result)

五、病毒繁殖 virus.py

def virus(N):
    #病毒存在小于 4 分钟时
    if N <= 4:
        return 1

    #病毒存在第五分钟开始
    else:
        return virus(N-1) + virus(N-4)

N = int(input('请输入现在是第几分钟（0<N<=60），以计算当前时间的病毒总数：
'))

total = virus(N)

print('当前时间的病毒粒子总数为:', total)

```



## 五、 实验结果

(运行结果，截图)

### 一、循环赛

```
==== RESTART: D:\MineP\Program\Python.py\Modern_Pro
请输入球队数量:4
请输入各球队的名字，以空格分隔:Fire Wolf Sun Pigeon
请输入比赛的两个队伍名称及比赛结果
1表示第一个队伍赢，2表示第二个队伍赢，3表示平局
请输入各场比赛的队伍以及结果，以空格分隔队伍名称及结果:
Sun Fire 1
请输入各场比赛的队伍以及结果，以空格分隔队伍名称及结果:
Pigeon Fire 1
请输入各场比赛的队伍以及结果，以空格分隔队伍名称及结果:
Sun Pigeon 2
请输入各场比赛的队伍以及结果，以空格分隔队伍名称及结果:
Pigeon Wolf 3
请输入各场比赛的队伍以及结果，以空格分隔队伍名称及结果:
Fire Wolf 3
请输入各场比赛的队伍以及结果，以空格分隔队伍名称及结果:
Sun Wolf 3
最高得分为: 7
>>>
```

### 二、时间换算

```
>>>
===== RESTART: D:\MineP\Program\Python.
请输入一个正整数N作为总秒数(N<86400):7259
该秒数转换为时间格式为: 02:00:59
>>>
===== RESTART: D:\MineP\Program\Python.
请输入一个正整数N作为总秒数(N<86400):500
该秒数转换为时间格式为: 00:08:20
```

### 三、计算 24

```
==== RESTART: D:\MineP\Program\Python.py\Modern_Program
请输入四个1到10之间的各不相同的数字，以空格分隔:10 2 4 8
2
>>>
```

### 五、病毒繁殖

```
>>>
===== RESTART: D:\MineP\Program\Python.py\Modern_Program
请输入现在是第几分钟 (0<N<=60) , 以计算当前时间的病毒总数: 6
当前时间的病毒粒子总数为: 3
>>>
```

## 六、 实验心得

(本次实验遇到的问题, 解决过程, 有什么收获等)

- 1.熟悉了字典的各种操作。
- 2.熟悉了 for 循环中嵌套 if 语句等。
- 3.学习了 str().zfill()的方法, 以及 f"{}"的格式化输出。
- 4.学习了递归的思想。
- 5.练习了 def 的使用。

深圳大学学生实验报告用纸

成绩评定:

实验过程 (60 分)	实验结果 (30 分)	心得体会 (10 分)	总分 (100 分)

指导教师签字: 年 月 日

备注:

注: 1、报告内的项目或内容设置, 可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后 10 日内。