

深圳大学实验报告

课程名称： 现代程序设计

实验项目名称： 复合数据类型

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 16 日

实验报告提交时间： 2023 年 10 月 19 日

教务部制

一、 实验要求

- a) 复合数据类型
- 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. 站队

n 名学生在操场，面向老师站成一排，从左到右编号是 $1 \sim n$ 。

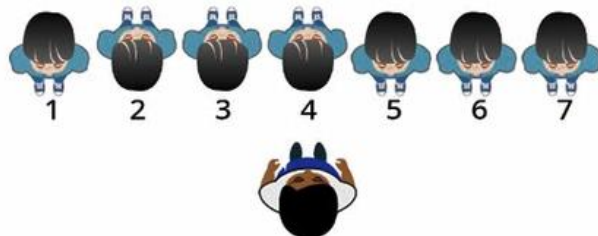
老师会喊 m 条口令，每条口令都包含一个整数 x 。口令下达后，凡是编号是 x 的倍数的学生向后转。

请你算一算，当 m 条口令都执行完毕后，面向老师的学生有几名。

例如：

$n=7, m=2$, 两条口令的 x 值分别是 2 和 3,

执行完所有口令后，编号为 1,5,6,7 的同学面向老师，编号为 2,3,4 的同学背向老师，如下图所示。



输入

第一行一个正整数 n , 为学生的数量 ($1 \leq n \leq 100$);

第二行一个正整数 m , 为口令的数量 ($1 \leq m \leq 10$);

第三行，以字符串形式输入 m 个不相同的正整数 (大小在 $1 \sim 10$ 范围内), 相邻两个正整数用一个空格分隔，依次表示每条口令的 x 值。

输出

一个整数，为 m 条口令都执行完毕后，面向老师的学生数量。

输入样例 1

7

2

2 3

输出样例 1

4

2. 太阳黑子

太阳黑子是太阳表面因温度相对较低而显得“黑”的局部区域，一般成群出现在太阳表面，并且对地球造成很明显的影。为此，人类根据多项科学研究追溯到几十万年前的黑子活动，并重建了数据记录。

现科学家拿到了一组数据，有 N 个整数，表示连续 N 年里每一年的太阳黑子最大数量，科学家希望找出所有数量连续增加的记录，并计算出连续增加记录中的最长连续年数。例如： $N = 8$ ，连续年里太阳黑子最大数量依次是 2、3、1、2、4、8、4、9，数量连续增加的记录有 (2、3)、(1、2)、(1、2、4)、(1、2、4、8)、(2、4)、(2、4、8)、(4、8) 和 (4、9)，连续增加的年数分别是 2、2、3、4、2、3、2、2，其中最连续年数为 4，故最后输出 4。

输入描述：

第一行输入一个正整数 N ($0 < N < 1000000$)，表示数据记录的年数。

第二行输入 N 个用空格隔开的正整数 ($0 < \text{正整数} \leq 1e9$)，表示连续 N 年的太阳黑子最大数量。

输出描述：

输出一个非负整数，表示太阳黑子最大数量连续增加记录中的最长连续年数

样例输入：

```
8
2 3 1 2 4 8 4 9
```

样例输出：

```
4
```

3. 面积最大

用一个 M 行 N 列的矩形网格表示某群岛疆域的地形图，每个小格子里都有一个数字 1 或者 0，其中，0 表示海水 (蓝色)，1 表示陆地 (绿色)，并且矩形网格之外的部分都为海水。每个小格子面积为 1，已知每个岛屿都是由 1 个或多个竖直或水平相邻的陆地小格子组成 (斜向不算相邻)，且周围都是海水。那么请你计算图中面积最大的岛屿与面积最小的岛屿之间的面积差值。

例如： $M=6$ ， $N=8$ 时，群岛疆域示意图如下：

0	0	0	0	1	0	1	1
1	1	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	0	1	1	1	0	1	1
0	0	0	1	1	0	1	0
0	0	0	1	0	0	1	0

其中，有 5 个岛屿，面积分别是 1，3，4，7，4，最大的岛屿面积是 7，最小的岛屿面积是 1，面积差是 $7-1=6$ 。

输入描述：

第一行输入两个正整数 M 和 N ($3 \leq M \leq 1000$ ， $3 \leq N < 1000$)，表示网格行数与列数，两数之间用一个空格隔开。

接下来输入 M 行，每行 N 个整数，表示地形图 (1 表示陆地，0 表示海水)，两个整数之间用一个空格隔开。

输出描述:

输出一个整数,表示面积最大的岛屿与面积最小的岛屿之间的面积差。

样例输入:

```
6 8
0 0 0 0 1 0 1 1
1 1 1 0 0 0 0 1
0 1 0 0 1 0 0 0
0 0 1 1 1 0 1 1
0 0 0 1 1 0 1 0
0 0 0 1 0 0 1 0
```

样例输出:

6

4. **斗鱼养殖场 (选做题)**

小慧用钢丝网建造了一批同样大小的正方体网箱,将它们拼成了 $M \times N$ 的矩形网格(每个格子都是一个网箱)放在池塘里,专门用来养殖一种凶猛的斗鱼。斗鱼天生好斗,只要旁边有其它斗鱼靠近,哪怕隔着钢丝网,它们都能斗个你死我活,所以每个网箱里最多只能饲养一条斗鱼;而且,两条斗鱼必须在不相邻的网箱(竖直或水平方向)。

另外,小慧在某些网箱里还安装了养殖设备,这些网箱里也不能养斗鱼。

小慧想知道,这组网箱一共有多少种可行的饲养方案(至少养一条斗鱼)由于方案数量比较大,所以只需要求出方案数量对 100000007 的取余结果。

例如: $M = 2, N = 2$, 矩形网格的格局如图所示(蓝色为水,灰色为设备):



饲养 1 条鱼的方案有 3 种(3 个蓝色网箱都可以养鱼);饲养 2 条鱼的方案有 1 种(用左上角和右下角的蓝色网箱养鱼):

这个 2×2 的矩形网格有 4 种饲养方案,4 对 100000007 取余的结果是 4,故输出 4。

输入描述:

第一行输入两个正整数 M 和 N ($2 \leq M \leq 100, 2 \leq N \leq 10$),分别表示矩形网格的行数和列数,两个正整数之间以一个空格隔开。

第二行开始输入 M 行,每行 N 个整数(只能是 1 或 0),1 表示水,表示设备,整数之间以一个空格隔开。

输出描述:

输出一个整数,表示至少养 1 条斗鱼的饲养方案数量对 100000007 取模的结果

样例输入:

```
2 2
1 1
0 1
```

样例输出:

4

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、站队

1. 首先输入学生的数量以及口令的数量。
2. 输入各口令的值。
3. 初始化学生队伍，使所有学生处于“面向老师”的状态。
4. 遍历学生队伍，对于对应口令 x 的倍数的学生，使他们“转身”。
5. 计算最终剩余的“面向老师”的学生数量。
6. 输出计算结果。

二、太阳黑子

1. 输入记录数据的年数。
2. 输入各年最大数据量，并用列表保存。
3. 初始化最长连续年份和当前连续年份计数器。
4. 遍历各年数据，当数据连续增长时，计数器加一。
5. 当数据不再连续增长时，更新此时的最大连续年份值并重置当前连续年份计数器。
6. 计算最终的最长连续年份的值并输出结果。

三、面积最大

1. 输入获取群岛地形的长度和宽度 (M 和 N)，然后获取用户输入的每行岛屿分布情况，并将其存储在一个二维列表中 ($grid$)。
2. 创建一个与群岛地形大小相同的布尔值二维列表 ($visited$)，用于跟踪已经访问过的岛屿。然后，它初始化最大面积 (max_area) 为 0，最小面积 (min_area) 为正无穷大。
3. 定义一个名为 $find$ 的递归函数，用于计算从给定位置开始的岛屿面积。在每次递归调用中，函数会检查当前位置是否越界或者已经访问过，如果是，则返回 0。如果当前位置是一个岛屿 ($grid[i][j] == 1$)，则将其标记为已访问并递归调用 $find$ 函数计算与当前位置相连的岛屿面积。最后，函数返回当前岛屿面积。

4. 然后，循环遍历群岛地形的每个位置。如果当前位置是一个未访问过的岛屿（`grid[i][j] == 1 and not visited[i][j]`），则调用 `find` 函数计算当前岛屿的面积，并更新最大面积和最小面积。
5. 计算最大面积和最小面积之差，并将其打印输出。

完整代码：

（必须有详细的注释）

一、站队 queue.py

```
n = int(input("请输入学生的数量 n(1<=n<=100): ")) #学生的数量
m = int(input("请输入口令的数量 m(1<=m<=100): ")) #口令的数量
commands = list(map(int, input("请输入各口令的值，以空格分隔数据:").split())) #口令的 x 值列表
```

#初始化数据，使所有学生都面向老师

```
students = [True] * n
```

```
for command in commands:
```

 #对于口令 x 的倍数

```
        for i in range(command - 1, n + 1, command):
```

```
            students[i - 1] ^= True   #转身
```

#计算最终面向老师的学生数量

```
count = students.count(True)
```

```
print("最终面向老师的学生数量为:%d"%count)
```

二、太阳黑子 sunstorm.py

```
N = int(input('请输入记录的年数 N(0<N<1000000): ')) #数据记录的年数
```

```

data = list(map(int, input('请输入各年最大数量, 各数据之间以空格分隔 (0<
正整数<=1*10e9): ').split())) #获取每年数据

max_len = 0 #用于计算最长连续年数
current_len = 1 #用于记录当年连续年数

#遍历各年数据进行计算
for i in range(1,N):
    if data[i] > data[i-1]: #当增长时, 计数加一
        current_len += 1
    else:
        max_len = max(max_len, current_len) #更新最大连续年数以及重置当前
连续年数的值
        current_len = 1

#计算最终的最大连续年数值
max_len = max(max_len, current_len)
print("最大数量连续增加记录中的最长连续年数为:%d"%max_len)

```

三、面积最大 area.py

```

# 获取群岛地形的长度和宽度
M,N = map(int, input("请输入群岛地形的长度和宽度, 以空格分
隔:").split())

# 创建一个空的群岛地形列表
grid = []

# 用户输入的每行岛屿分布情况存储在 grid 列表中
for k in range(M):
    row = list(map(int, input("请输入第 {} 行岛屿的分

```

```
布:".format(k+1)).split()))

        grid.append(row)

# 创建一个与群岛地形大小相同的布尔值二维列表，用于跟踪已经访问过的岛屿
visited = [[False] * N for _ in range(M)]

# 初始化最大面积和最小面积
max_area = 0
min_area = float('inf')

# 定义函数用于计算岛屿面积
def find(i, j):

    # 检查当前位置是否越界或者已经访问过
    if i < 0 or i >=M or j < 0 or j >= N:
        return 0

    if visited[i][j] or grid[i][j] == 0:
        return 0

    # 标记当前位置为已访问
    visited[i][j] = True

    # 初始化当前岛屿面积为 1
    area = 1

    # 调用函数计算与当前位置相连的岛屿面积
    area += find(i - 1, j) # 上
    area += find(i + 1, j) # 下
    area += find(i, j + 1) # 左
    area += find(i, j - 1) # 右
```



```
# 返回当前岛屿面积

return area

# 遍历群岛地形的每个位置
for i in range(M):
    for j in range(N):
        # 如果当前位置是一个未访问过的岛屿
        if grid[i][j] == 1 and not visited[i][j]:
            # 调用函数计算当前岛屿的面积
            area = find(i, j)

            # 更新最大面积和最小面积
            max_area = max(max_area, area)
            min_area = min(min_area, area)

# 计算最大面积和最小面积之差并打印结果
area_diff = max_area - min_area
print(area_diff)
```

五、 实验结果

（运行结果，截图）

一、站队

```
>>>
===== RESTART: D:\MineP\Program\Python.py\Modern_Program\Task6\queue.py
请输入学生的数量n(1<=n<=100): 7
请入口令的数量m(1<=m<=100): 2
请输入各口令的值, 以空格分隔数据:2 3
最终面向老师的学生数量为:3
>>>
===== RESTART: D:\MineP\Program\Python.py\Modern_Program\Task6\queue.py
请输入学生的数量n(1<=n<=100): 9
请入口令的数量m(1<=m<=100): 2
请输入各口令的值, 以空格分隔数据:2 3
最终面向老师的学生数量为:3
>>>
```

二、太阳黑子

```
>>>
===== RESTART: D:\MineP\Program\Python.py\Modern_Program\Task6\sunstorm.py
请输入记录的年数N(0<N<1000000): 8
请输入各年最大数量, 各数据之间以空格分隔(0<正整数<=1*10e9): 2 3 1 2 4 8 4 9
最大数量连续增加记录中的最长连续年数为:4
>>>|
```

三、面积最大

```
>>>
===== RESTART: D:\MineP\Program\Python.py\Modern_Program\Task6\area.py
请输入群岛地形的长度和宽度, 以空格分隔:6 8
请输入第1行岛屿的分布:0 0 0 0 1 0 1 1
请输入第2行岛屿的分布:1 1 1 0 0 0 0 1
请输入第3行岛屿的分布:0 1 0 0 1 0 0 0
请输入第4行岛屿的分布:0 0 1 1 1 0 1 1
请输入第5行岛屿的分布:0 0 0 1 1 0 1 0
请输入第6行岛屿的分布:0 0 0 1 0 0 1 0
6
```

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.熟悉了嵌套的操作，提高了解题能力。
- 2.掌握了对数据的各种处理方法以及对数据值的更新操作。
- 3.搜集资料学习了递归的操作。
- 4.学习了布尔值列表的操作。

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。