

Bài 1: Dãy số

Hoàn thiện các hàm trong file **Seq.py** thực hiện các yêu cầu sau

1. Hàm **isAlt(a)** thực hiện kiểm tra dãy số a có phải là dãy đan dẫu hay không? Nếu đúng trả về giá trị là **True**/ Nếu sai trả về giá trị là **False**
2. Hàm **isGrow(a)** thực hiện kiểm tra dãy số a có phải là dãy tăng hay không? Nếu đúng trả về giá trị là **True**/ Nếu sai trả về giá trị là **False**
3. Hàm **isMulti(a)** thực hiện kiểm tra dãy số a có phải là cấp số nhân hay không? Nếu đúng trả về giá trị là **True**/ Nếu sai trả về giá trị là **False**
4. Hàm **isAdd(a)** thực hiện kiểm tra dãy số a có phải là cấp số cộng hay không? Nếu đúng trả về giá trị là **True**/ Nếu sai trả về giá trị là **False**

Seq.py

```
1 def isAlt(a):
2     pass
3
4
5 def isGrow(a):
6     pass
7
8
9 def isMulti():
10    pass
11
12
13 def isAdd():
14    pass
```

Bài 2: Kiểm tra ma trận ma phương

Viết chương trình nhập vào ma trận từ bàn phím, kiểm tra xem ma trận có phải là ma phương hay không?

Ma trận ma phương là ma trận có tổng các hàng, các cột và các đường chéo bằng nhau.

- Hàm **isMagicSquare()** là hàm kiểm tra tính chất ma phương của một ma trận
- Hàm **inputMatrix()** thực hiện việc nhập và trả lại một ma trận (mảng 2 chiều) các số nguyên
- Mỗi hàng của ma trận được nhập trên 1 dòng, mỗi số cách nhau bởi 1 dấu **tab** ('\t')
Việc nhập kết thúc khi gặp dòng trống

MagicSquare.py

```
1 def isMagicSquare(m):  
2     return False  
3  
4 def inputMatrix():  
5     m = []  
6  
7     return m  
8  
9
```

Bài 3: Mật khẩu hợp lệ

Một mật khẩu của một tài khoản trên trang web môn học được coi là hợp lệ nếu có đủ các yếu tố sau:

1. Độ dài từ **8** đến **100** ký tự
2. Có chữ cái in hoa
3. Có chữ cái thường
4. Có số
5. Có ký tự đặc biệt là một trong các ký tự sau: **~!@#\$%^&***

Viết chương trình nhập vào một mật khẩu, kiểm tra xem mật khẩu đó có hợp lệ theo quy tắc ở trên hay không?

Hoàn thành hàm **checkPassword(s)** trong file **Password.py**, hàm này trả về True nếu mật khẩu là hợp lệ, trả lại false nếu mật khẩu không hợp lệ.

Password.py

```
1 def checkPassword(s):  
2     return False  
3  
4
```

Bài 4: Tính phương sai

Viết chương trình tính giá trị trung bình phương sai, độ lệch chuẩn của một danh sách các số thực

- Giá trị trung bình là trung bình cộng của các giá trị trong danh sách
- Phương sai tính bằng công thức bên dưới trong đó e là giá trị trung bình, x_i là giá trị thứ i của danh sách.
- Độ lệch chuẩn được tính bằng căn bậc 2 của phương sai.

$$\delta = \frac{1}{n} \sum_1^n (x_i - e)^2$$

Hoàn thiện các phương thức trong file [Statistical.py](#)

Statistical.py

```

1 def mean(s):
2     return 0
3
4 def variance(s):
5     return 0
6
7 def standarddeviation(s):
8     return 0
9
10
11

```

Bài 5: Góc giữa hai vector

Viết chương trình tính góc giữa 2 vector:

Hoàn thiện hàm **def cosineb2v(u,v)** trong file **Vec.py**, hàm này trả về giá trị **cosine** của góc giữa hai **vector** u và v theo công thức sau:

$$\cos(\vec{u}, \vec{v}) = \frac{(\vec{u} \cdot \vec{v})}{(|\vec{u}| \times |\vec{v}|)}$$

Trong đó $(\vec{u} \cdot \vec{v}) = u_1v_1 + u_2v_2$, trong đó, $u = (u_1, u_2)$, $v = (v_1, v_2)$. Nếu véc-tơ có nhiều hơn hai thành phần, bạn chỉ việc thêm tiếp $+ u_3v_3 + u_4v_4\dots$

$$|\vec{u}| = \sqrt{u_1^2 + u_2^2}$$

Vec.py

```
1 def cosineb2v(u, v):  
2     return 1  
3  
4
```

Bài 6: Sắp xếp chẵn lẻ

Hoàn thiện hàm **customSort(a)** trong file [SortAdv.py](#).

Hàm thực hiện sắp xếp các phần tử trong danh sách a theo thứ tự như sau:

Các phần tử chẵn bên trái, lẻ bên phải, các số chẵn tăng dần, các số lẻ sắp xếp giảm dần

Kết quả trả về là 1 danh sách được sắp xếp theo thứ tự trên

Ví dụ **a = [1,2,3,4,5,6,7,8,9,10]**

kết quả trả về là

[2,4,6,8,10,9,7,5,3,1]

SortAdv.py

```
1 #Hoàn thiện hàm customSort  
2  
3  
4 def customSort(a):  
5     '''  
6         Hàm thực hiện sắp xếp các phần tử trong a, theo thứ tự:  
7         - Chẵn bên trái, lẻ bên phải  
8         - Chẵn tăng dần, lẻ giảm dần  
9         ví dụ a = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
10        kết quả là [2, 4, 6, 8, 10, 9, 7, 5, 3, 1]  
11        '''  
12  
13  
14  
15     return a  
16
```

Bài 7: Bank account query

Viết chương trình tính số tiền thực của một tài khoản ngân hàng dựa trên nhật ký giao dịch được nhập vào từ giao diện điều khiển.

Định dạng nhật ký được hiển thị như sau:

D 100

W 200

(D là tiền gửi, W là tiền rút ra).

Giả sử đầu vào được cung cấp là:

D 300

D 300

W 200

D 100

Thì đầu ra sẽ là: 500

Đầu vào được nhập từ bàn phím trên từng dòng, việc nhập kết thúc khi gặp dòng trống.