

BÀI TẬP

Lập trình cơ bản

Mục tiêu: Sinh viên thực hành những bài tập cơ bản bằng ngôn ngữ Python, làm việc với biến, thực hiện các lệnh nhập xuất, vòng lặp, hàm, dữ liệu kiểu xâu ký tự - String, kiểu Danh sách – List. Sinh viên thực hành thành thạo các thao tác cơ bản trên IDE eclipse với plugin Pydev để lập trình các chương trình bằng ngôn ngữ Python. Sinh viên tham khảo các bài tập mẫu để làm các bài tập được yêu cầu.

Bài tập mẫu

Bài tập mẫu: Tính tổng các giá trị chẵn, tổng các giá trị lẻ, tổng các giá trị trong một dãy số.

```
def sumEven(a):
    s = 0
    for e in a:
        if(e % 2 == 0):
            s = s + e

    return s

def sumOdd(a):
    s = 0
    for e in a:
        if(e % 2 == 1):
            s = s + e

    return s

def sumList(a):
    return sum(a)

n = int(input('Nhập số phần tử trong dãy số n = '))

aList = [0 for j in range(0, n)]

for i in range(0, n):
    aList[i] = int(input('list[' + str(i) + '] = '))
print('Tổng các giá trị chẵn trong dãy = ', sumEven(aList))
print('Tổng các giá trị lẻ trong dãy = ', sumOdd(aList))
print('Tổng các giá trị trong dãy = ', sumList(aList))
```

Chương trình thực hiện nhập vào số nguyên n, khai báo một danh sách kích thước n phần tử với giá trị ban đầu bằng 0 (`aList = [0 for j in range(0, n)]`)

Chú ý cách dùng hàm `sum()` như trong ví dụ hàm `sumList(a)`:

```
def sumList(a):  
    return sum(a)
```

Dưới đây là một ví dụ khác, cách viết dưới đây thường được dùng trong lập trình Python, thay vì cách viết như ví dụ trước đó thường dùng trong Java.

```
def sumEven(a):  
    return sum(a[i] for i in range(0, len(a)) if a[i]%2 == 0)  
  
def sumOdd(a):  
    return sum(i for i in a if i%2 == 1)  
  
def sumList(a):  
    return sum(a)  
  
n = int(input('Nhập số phần tử trong dãy số n = '))  
  
aList = []  
  
for i in range(0, n):  
    aList.append(int(input('list[' + str(i) + '] = ')))  
print('Tổng các giá trị chẵn trong dãy = ', sumEven(aList))  
print('Tổng các giá trị lẻ trong dãy = ', sumOdd(aList))  
print('Tổng các giá trị trong dãy = ', sumList(aList))
```

Dưới đây trình bày về một số phương thức cơ bản khi làm việc với chuỗi ký tự

Một số phương thức thông dụng khi làm việc với dữ liệu chuỗi.

- Lấy độ dài chuỗi `len()`
`a = "Hello, World!"`
`print(len(a))`
- Chuyển ký tự về ký tự hoa / thường `lower()` / `upper()`
`a = "Hello, World!"`
`print(a.lower())`
`print(a.upper())`

- c) Thay thế chuỗi `replace()`
- ```
a = "Hello, World!"
print(a.replace("H", "J"))
```
- d) Tách chuỗi `split()`
- ```
a = "Hello, World!"
print(a.split(",")) # returns ['Hello', ' World!']
```
- e) Kiểm tra chuỗi có chứa / không chứa chuỗi con `in` / `not in`
- ```
txt = "The rain in Spain stays mainly in the plain"
x = "ain" in txt
print(x)

txt = "The rain in Spain stays mainly in the plain"
x = "ain" not in txt
print(x)
```
- f) Cộng chuỗi / nối chuỗi
- ```
a = "Hello"
b = "World"
c = a + b
print(c)
```
- g) Định dạng chuỗi `format()`
- ```
age = 36
txt = "My name is John, and I am {}"
print(txt.format(age))

quantity = 3
itemno = 567
price = 49.95
myorder = "I want {} pieces of item {} for {} dollars."
print(myorder.format(quantity, itemno, price))

quantity = 3
itemno = 567
price = 49.95
myorder = "I want to pay {2} dollars for {0} pieces of
item {1}."
print(myorder.format(quantity, itemno, price))
```

## Bài tập mẫu: Đếm số từ trong văn bản

```
def wordCount(s):
 token = s.split(' ')
 w = []
 c = []
 for t in token:
 if t not in w:
 w.append(t)
 c.append(count(t,token))
 return w, c

def count(w, token):
 count = 0
 for t in token:
 if w == t:
 count +=1
 return count

text = "Lúa nếp Là Lúa nếp Làng Lúa Lên Lớp Lớp Lòng nàng
Lâng Lâng"

words, counts = wordCount(text)

for i in range(0,len(words)):
 print(words[i],':',counts[i])
```

Dưới đây là phiên bản ngắn hơn và thường được dùng hơn trong Python

```
def wordCount(s):
 tokens = s.split(' ')
 words = list(dict.fromkeys(tokens))
 counts = [tokens.count(words[i]) for i in range(0,
len(words))]
 return words, counts

text = "Lúa nếp là Lúa nếp Làng Lúa Lên Lốp Lốp Lòng nàng
Lâng Lâng"

words, counts = wordCount(text)

for i in range(0, len(words)):
 print(words[i], ': ', counts[i])
```

## Bài mẫu: Nhập và in ma trận chuyển vị

```
def inputMatrix(matrix):
 for i in range(0, len(matrix)):
 for j in range(0, len(matrix[i])):
 matrix[i][j] =
int(input('matrix['+str(i)+'']['+str(j)+'']='))

def printMatrix(matrix):
 for i in range(0, len(matrix)):
 for j in range(0, len(matrix[i])):
 print(matrix[i][j],end=" ")
 print()

def tranMatrix(matrix):
 n = len(matrix)
 m = len(matrix[0])
 t = [[0 for j in range(0,n)] for i in range(0,m)]

 for i in range(0, len(matrix)):
 for j in range(0, len(matrix[i])):
 t[j][i] = matrix[i][j]

 return t

m = int(input('Nhập vào số hàng = '))
n = int(input('nhập vào số cột = '))
matrix = [[0 for j in range(0,n)] for i in range(0,m)]

inputMatrix(matrix)
printMatrix(matrix)
t = tranMatrix(matrix)
print('Ma trận chuyển vị')
printMatrix(t)
```

## Bài tập thực hành

### Bài 1:

Viết chương trình nhập vào số nguyên dương  $n$  và nhập vào 1 dãy  $n$  số nguyên, viết các hàm kiểm tra tính chất của dãy:

1. Kiểm tra dãy có phải dãy đan dấu hay không?
2. Kiểm tra dãy có phải dãy tăng hay không?
3. Kiểm tra dãy có lập thành cấp số nhân hay không?
4. Kiểm tra dãy có lập thành cấp số cộng hay không?

### Bài 2:

Viết chương trình tìm dạng tối giản của phân số, một phân số được biểu diễn bằng hai số nguyên tương ứng với tử số và mẫu số, viết các hàm tìm ước chung lớn nhất, bội chung nhỏ nhất của tử số và mẫu số, sau đó viết hàm tìm dạng tối giản của phân số.

### Bài 3:

Nhập vào số nguyên dương  $m$ , in ra tam giác PASCAL kích thước  $m$ . Tam giác PASCAL là tam giác biểu diễn các hệ số của nhị thức  $(a + b)^m$ . Ví dụ:

$1 > m = 0$

$1 \ 1 > m = 1$

$1 \ 2 \ 1 > m = 2$

$1 \ 3 \ 3 \ 1 > m = 3$

### Bài 4:

Viết hàm tìm và trả lại ký tự xuất hiện nhiều nhất trong chuỗi  $S$

### Bài 5:

Nhập vào một danh sách gồm  $N$  họ tên, viết hàm tách tên và họ đệm từ một chuỗi họ tên, sắp xếp danh sách  $N$  họ tên theo tên, nếu tên trùng nhau thì sắp theo họ đệm.

### Bài 6:

Một phân tử được gọi là điểm yên ngựa của ma trận nếu nó là phân tử bé nhất của hàng chứa nó đồng thời là phân tử lớn nhất của cột chứa nó. Nhập ma trận  $A$  và tìm các điểm yên ngựa của ma trận thông qua các hàm:



1. Hàm tìm giá trị nhỏ nhất của hàng i trong ma trận m
2. Hàm tìm giá trị lớn nhất trong cột j trong ma trận m
3. Hàm kiểm tra phần tử [i,j] của ma trận m có phải là điểm yên ngựa hay không?
4. Hàm in ra các điểm yên ngựa trong ma trận m

### Bài 7:

Viết chương với hai hàm thực hiện phép cộng và phép nhân 2 ma trận.

### Bài 8:

Nhập vào số nguyên dương N và N số nguyên, in ra các ma trận có kích thước khác nhau từ N số nguyên đã nhập. Ví dụ với N = 8 có thể tạo thành các ma trận có kích thước khác nhau là: 1 x 8, 2 x 4, 4 x 2, 8 x 1

### Bài 9:

Ma trận vuông A được gọi là ma phương nếu tổng mỗi hàng, tổng mỗi cột, tổng mỗi đường chéo đều bằng nhau. Nhập ma trận vuông A và kiểm tra tính chất ma phương.

### Bài 10:

Trong toán vui, một ma trận kì ảo bậc n (còn gọi là ma phương hay hình vuông ma thuật) là một cách sắp xếp  $n^2$  số, thường là các số nguyên phân biệt, trong một bảng vuông sao cho tổng n số trên mỗi hàng, cột, và đường chéo đều bằng nhau. Ma trận kì ảo chuẩn chứa các số nguyên từ 1 đến  $n^2$ .

Viết chương trình sinh ma phương lẻ (n là số lẻ) theo thuật toán sau:

1. Xuất phát từ một ô ở chính giữa hàng, hoặc cột với giá trị ban đầu là 1
2. Tiếp tục điền giá trị các ô chưa được điền bằng cách đi theo hướng Phía trên bên phải, nếu đi hết biên của ma trận thì xoay vòng
  - a. Nếu đi đến ô trên cùng bên phải thì sẽ di chuyển xuống dưới 1 ô
  - b. Nếu đi tới một ô đã có giá trị thì di chuyển xuống dưới 1 ô

