## BỘ CÔNG THƯƠNG TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỎ CHÍ MINH KHOA LÝ LUẬN CHÍNH TRỊ



# BÁO CÁO THỰC HÀNH 1 MÔN: NGÔN NGỮ PYTHON

Giáo viên hướng dẫn: Cao Văn Kiên

Nhóm thực hiện: Nhóm 3

*Lớp học phần:* DHDTVT16B – 420300358615

STT	HQ VÀ TÊN	MSSV
1	Đỗ Quang Vinh	20075691
2	Vũ Bạch Long	21139141
3	Bùi Lê Minh Thanh	20040201

Tp. Hồ Chí Minh, ngày 12 tháng 08 năm 2022

Bài 1: Viết chương trình sử dụng numpy thực hiện các phép cộng, trừ, nhân, chia 2 mảng 2 chiều. Thực hiện phép tính tương tự bằng tay, so sánh kết quả.

Code:

```
cau1.py > ...
    import hammatran as cal
    import numpy as np
    x = [12,45]
    z = [22,33]
    y = [24,5]
    t = [4,10]
    a = np.array([y,t])
    b = np.array([x,z])
    try:
       cal.add(a,b)
        cal.sub(a,b)
        cal.mult(a,b)
        cal.div(a,b)
    except ValueError:
        print("Ma trận không hợp lệ")
```

Các hàm tính toán: add là hàm cộng, sub là hàm trừ mult là hàm nhân, div là hàm chia

```
mmatran.py 🗸 🗘 div
 import numpy as np
 def sub(m1,m2):
     c = m1 - m2
     print("Trừ: ")
     print(c)
 def add(m1,m2):
     c = m1 + m2
     print("Công: ")
     print(c)
 def mult(ma,mb):
     sa = ma.shape
     sb = mb.shape
     if sa[1] == sb[0]:
         c = ma @ mb
     else:
          print("Kích thước không phù hợp")
     print("Kết quả sau khi nhận 2 ma trận:")
     print(c,"\n")
 def div(ma,mb):
     c = np.linalg.det(mb)
     if c != 0:
         md = np.matrix(mb)
         md = md.I
         print("Ma trận đảo: ")
         print(md)
         sa = ma.shape
         sd = md.shape
         if sa[1] == sd[0]:
             d = ma @ md
          else:
              print("Kích thước không phù hợp")
     else:
          print("Không có giá trị phù hợp")
      print("Kết quả sau khi chia 2 ma trận: ")
      print(d)
```

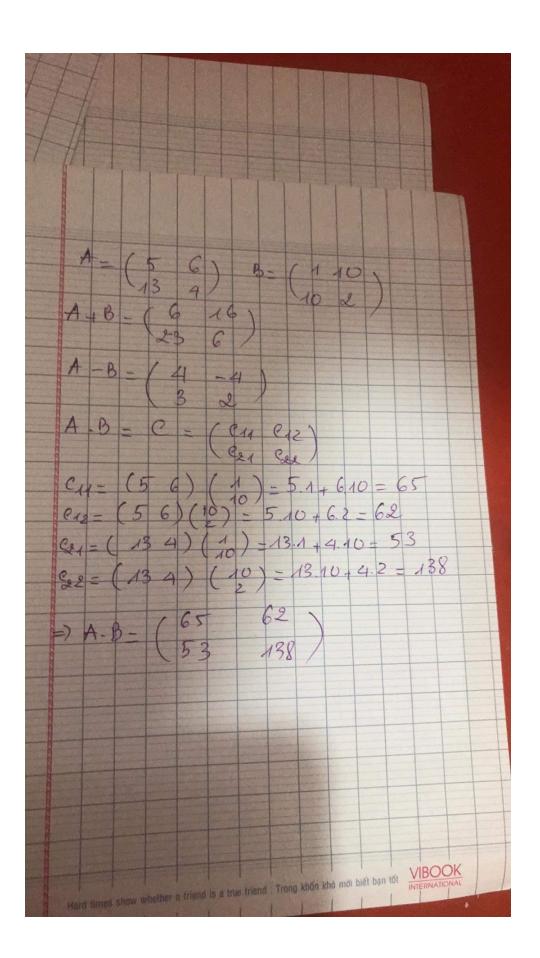
## Đầu vào:

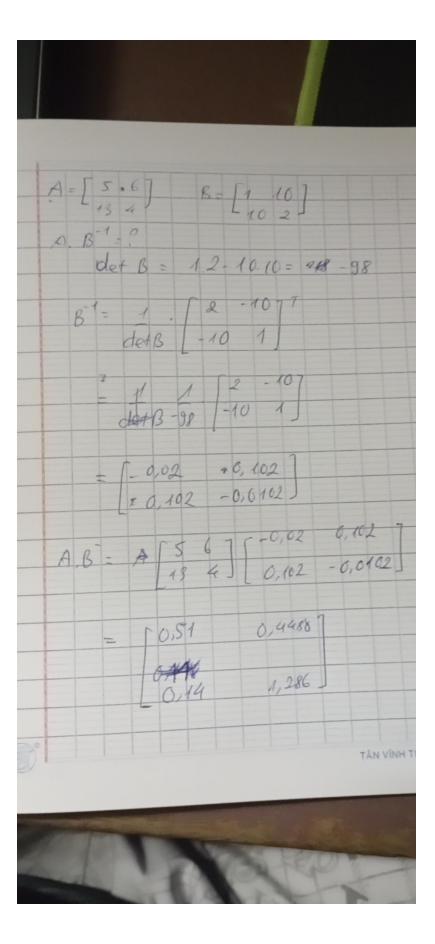
```
x = [1,10]
y = [5,6]
z = [10,2]
t = [13,4]
```

Đầu ra:

```
Cộng:
[[ 6 16]
[23 6]]
Trừ:
[[ 4 -4]
[ 3 2]]
Kết quả sau khi nhân 2 ma trận:
[[ 65 62]
[ 53 138]]
Ma trận đảo:
[[-0.02040816 0.10204082]
[ 0.10204082 -0.01020408]]
Kết quả sau khi chia 2 ma trận:
[[0.51020408 0.44897959]
 [0.14285714 1.28571429]]
PS D:\PythonApplication1\PythonApplication1\Bài 6>
```

Giải tay:





Bài 2: Viết chương trình sử dụng numpy thực hiện các phép cộng, trừ, nhân, chia 2 ma trận được lưu trong 2 file .csv, kết quả in ra màn hình và lưu ở 1 file .csv khác. Thực hiện phép tính nhân, chia tương tự bằng tay, so sánh kết quả. Code:

```
import hammatran as cal
import csv
import numpy as np
matrix1 = open("matrix1.csv", "r", encoding='utf-8-sig')
matrix2 = open("matrix2.csv", "r", encoding='utf-8-sig')
csv.reader(matrix1)
csv.reader(matrix2)
a = []
b = []
for row in matrix1:
   1 = []
   x = row.strip().split(",")
    for i in range(0,len(x)):
        1.append(int(x.pop(0)))
    a.append(1)
a = np.array(a)
print(a)
for row in matrix2:
    1 = []
    x = row.strip().split(",")
    for i in range(0,len(x)):
        1.append(int(x.pop(0)))
    b.append(1)
b = np.array(b)
print(b)
try:
    cal.add(a,b)
    cal.sub(a,b)
    cal.mult(a,b)
    cal.div(a,b)
except ValueError:
    print("Ma trân không hợp lệ")
```

Đầu vào: tạo 2 file.csv "matrix1", "matrix2" chứa thông tin ma trận

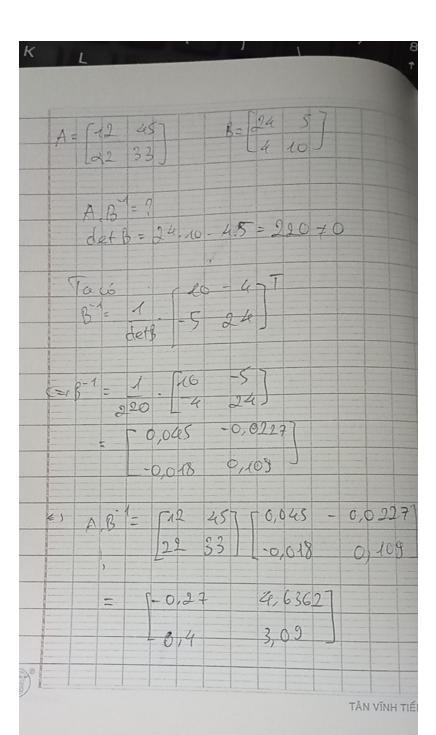
```
C: > Users > Admin > Downloads > III matrix2.csv

1 24,5
2 4,10
3
```

#### Đầu ra:

## Giải tay:

12 22	45	245
AFB.	12+24	45+5
	22,4	83410
	= (36	43
D-5 =	18	23)
A.B -	C = ( C11 ( C21 (	22
e <sub>11</sub> = (	(12 45) (8	18-5 + 45.4 = 468
C12 = C C22 = C	12 45 ) (20 2 2 33) (24 2 2 33) (5)	)=27.24+33.4=660 )= 440
=) A.	B = (46°)	8 510
Hard times sho		Trong khổn khô mởi biết bạn tốt VIBOOK INTERNATIONAL





Bài 3: Xây dựng 1 class có 2 biến dạng numpy array, trong class xây dựng các hàm tính định thức, ma trận nghịch đảo. Thực hiện phép tính tương tự bằng tay, so sánh kết quả.

Code:

```
import numpy as np
class cau3:
   a = np.array([[2,3],[4,5]])
   b = np.array([[3,8],[7,8]])
   t1 = a.transpose()
   t2 = b.transpose()
    def dinhthuc(self):
        self.c = np.float32(np.linalg.det(self.a))
        self.d = np.float32(np.linalg.det(self.b))
        print("Định thức của a: \n",self.c)
        print("Định thức của b: \n",self.d)
    def nghichdao(self):
        self.md = np.matrix(self.a)
         self.md = self.md.I
        print(self.md)
var1 = cau3()
var1.dinhthuc()
var1.nghichdao()
```

Đầu vào:

```
a = np.array([[2,3],[4,5]])
b = np.array([[3,8],[7,8]])
```

Đầu ra:

```
Dinh thức của a:
-2.0
Dịnh thức của b:
-32.0
[[-2.5 1.5]
[ 2. -1. ]]
PS C:\Users\Admin> [
```

Câu 4: Xây dựng 1 class, trong đó có 1 biến X được khởi tạo ngẫu nhiên dạng ma trận NxM có giá trị trong khoảng (min, max) cho trước sử dụng numpy; có 1 hàm để đọc file .csv gán giá trị cho X, 1 hàm để lưu giá trị của X vào 1 file .csv, đầu vào của hàm để đường dẫn tới file đó.

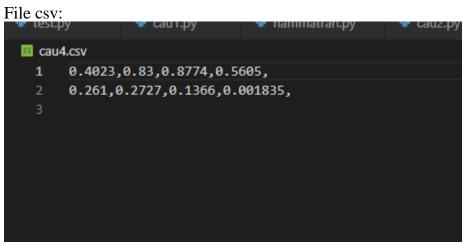
```
import csv
import numpy as np
class cau4:
    min = 2
    max = 4
    X = np.float16(np.array(np.random.rand(min,max)))
    K = np.float16(np.array(np.random.rand(min,max)))
    def csvsave(self,address):
        with open(address, "w+", encoding='utf-8-sig') as f:
            csv.reader(f)
            for self.list in self.X:
                self.g = [0]*len(self.list)
                for self.i in range (0, len(self.list)):
                    self.g[self.i] = str(self.list[self.i])
                for self.string in self.g:
                    f.write(self.string+",")
                f.write("\n")
    def csvread(self,address):
        with open(address, "r+", encoding='utf-8-sig') as f:
            csv.reader(f)
            self.a = []
            for self.row in f:
                self.1 = []
                self.x = self.row.strip().split(",")
                self.x.remove('')
                for self.i in range(0,len(self.x)):
                    self.l.append(float(self.x.pop(0)))
                self.a.append(self.1)
            self.X = np.array(self.a)
            print("Ma trận X:\n",self.X)
    def update(self):
        print("Trước update:\n",self.X)
        self.X = self.X + 5*self.K
        print("Sau update:\n",self.X)
c4 = cau4()
c4.csvsave('D:\\PythonApplication1\\PythonApplication1\\Bài 6\\cau4.csv')
c4.csvread('D:\\PythonApplication1\\PythonApplication1\\Bài 6\\cau4.csv')
c4.update()
```

#### Đầu vào:

```
class cau4:
    min = 2
    max = 4
    X = np.float16(np.array(np.random.rand(min,max)))
```

## Kết quả:

```
Ma trận X:
 [[0.4023
            0.83
                     0.8774
                              0.5605 ]
 [0.261
           0.2727
                    0.1366
                            0.001835]]
```



Câu 5: Viết thêm hàm update vào class ở câu 4, hàm update() như sau: X = X - alpha\*K, trong đó K là

ma trận số ngẫu nhiên có kích thước giống X, alpha là một hệ số thực. In ra màn hình kết quả của

X trước và sau update.

#### Code:

```
import csv
import numpy as np
class cau4:
   min = 2
    max = 4
    X = np.float16(np.array(np.random.rand(min,max)))
    K = np.float16(np.array(np.random.rand(min,max)))
    def csvsave(self,address):
        with open(address, "w+", encoding='utf-8-sig') as f:
            csv.reader(f)
            for self.list in self.X:
                self.g = [0]*len(self.list)
for self.i in range (0, len(self.list)):
                    self.g[self.i] = str(self.list[self.i])
                for self.string in self.g:
                    f.write(self.string+",")
                f.write("\n")
    def csvread(self,address):
        with open(address, "r+", encoding='utf-8-sig') as f:
           csv.reader(f)
            self.a = []
            for self.row in f:
                self.x = self.row.strip().split(",")
                self.x.remove('')
                for self.i in range(0,len(self.x)):
                    self.l.append(float(self.x.pop(0)))
                self.a.append(self.1)
            self.X = self.a
           print(self.X)
    def update(self):
        print("Trước update\n",self.X)
        self.X = self.X + 5*self.K
        print("Sau update\n",self.X)
c4.update()
```

### Đầu vào:

```
class cau4:
    min = 2
    max = 4
    X = np.float16(np.array(np.random.rand(min,max)))
    K = np.float16(np.array(np.random.rand(min,max)))
    def csysave(self address):
```

Kêt quả:

```
Trước update
[[0.3843 0.3984 0.8477 0.2678]
[0.1956 0.6787 0.6777 0.9644]]
Sau update
[[4.5     1.102 1.533 1.167]
[3.707 4.05     3.266 2.27 ]]
PS D:\PythonApplication1\PythonApplication1\Bài 6> [
```

## Câu nâng cao:

Tính số tiền tiết kiệm sau N tháng lãi suất kép với số tiền ban đầu và mỗi tháng gửi vào ngân hàng thêm. Viết chương trình dạng class. Số tiền gửi vào hàng tháng có thể thêm vào qua file .csv. Biết lãi suất ngân hàng là 7%/năm, tìm lãi suất theo tháng sao cho 12 tháng số tiền chênh lệch so với lãi suất năm không quá 1%.