

CÂU HỎI TRẮC NGHIỆM THI GIỮA KỲ MÔN ĐẠI SỐ TUYẾN TÍNH

Nội dung ôn tập

A. PHẦN I – CÁC HÀM LỆNH CỦA PYTHON.....	1
B. PHẦN II –THỰC HÀNH TÍNH TOÁN ĐẠI SỐ TUYẾN TÍNH.....	6
C. PHẦN III – ỨNG DỤNG ĐẠI SỐ TUYẾN TÍNH	9
D. PHẦN IV – TÍNH TOÁN.....	11

A. PHẦN I – CÁC HÀM LỆNH CỦA PYTHON

Là những câu hỏi yêu cầu hiểu/biết về các lệnh của Python.

Câu 1: Dữ liệu ma trận trong thư viện numpy của Python có tên là:

- a. `numpy.Array`
- b. `numpy.matrix`
- c. `numpy.Matrix`
- d. Tên khác

(đáp án: b và d, tên khác: `numpy.array` hoặc `numpy.mat`)

Câu 2: Các khai báo nào không hợp lệ để tạo ra một ma trận 2x3 bằng numpy:

- a. `mt1 = numpy.array([(1,2,3),(4,5,6)])`
- b. `mt2 = numpy.array([[1,2,3],[4,5,6]])`
- c. `mt3 = numpy.mgrid[0:3, 0:2]`
- d. `mt4 = numpy.mgrid[0:2, 0:3]`

(đáp án: c và d)

Câu 3: Chuyển vị một ma trận A của numpy bằng lệnh:

- a. `A.transpose()`
- b. `A.T`
- c. `numpy.transpose(A)`

d. `numpy.T(A)`

(đáp án: a, b, c)

Câu 4: Sau khi `import scipy` và `import numpy`, lệnh nào dưới đây để nghịch đảo một ma trận A của numpy hoặc scipy bằng lệnh:

- a. `A.I`
- b. `A.inv`
- c. `numpy.linalg.inv(A)`
- d. `scipy.linalg.inv(A)`

(đáp án: a, c, d)

Câu 5: Để sử dụng được lệnh **A.H**. Các câu lệnh khai báo A nào dưới đây đúng?

- a. `A = numpy.mat ([[1,2,3],[1,2,3]])`
- b. `A = numpy.array([[1,2,3],[1,2,3]])`
- c. `A = numpy.matrix(np.random.random((2,2)))`
- d. `A = numpy.asmatrix(numpy.array([[1,2,3],[1,2,3]]))`

(Đáp án: a, c, d)

Hướng dẫn: kiểu array không có hàm tính .H

Câu 6: Tạo ma trận đơn vị I_3 bằng lệnh numpy/scipy trừ các lệnh:

- a. `i3 = numpy.eye(3, k=1)`
- b. `i3 = numpy.identity(3)`
- c. `i3 = numpy.mat(numpy.identity(3))`
- d. `i3 = scipy.linalg.identity(3)`

(đáp án: a và d)

Câu 7: Trong Python 3, hàm nào dưới đây để tính ma trận bình phương của $A = \begin{bmatrix} 10 & 11 \\ 3 & -5 \end{bmatrix}$:

- a. `A.dot(A)`
- b. `numpy.dot(A, A)`

- c. `numpy.multiply(A, A)`
- d. `A @ A`

(đáp án: a, b, d)

Câu 8: Để giải phương trình $Ax=b$, với $A = A = \begin{bmatrix} 10 & 11 \\ 3 & -5 \end{bmatrix}$ và $b=\begin{bmatrix} 2 \\ 3 \end{bmatrix}$. Thư viện `numpy/scipy` sử dụng lệnh sau:

- a. `np.solve(A,b)`
- b. `numpy.linalg.solve(A,b)`
- c. `scipy.linalg.solve(A,b)`
- d. `np.divide(A,b)`

(đáp án: b, c)

Câu 9: Lệnh nào dưới đây hợp lệ để tính hạng của ma trận A :

- a. `numpy.matrix_rank(A)`
- b. `scipy.matrix_rank(A)`
- c. `scipy.linalg.rank(A)`
- d. Lệnh khác.

(đáp án: d; lệnh khác: hướng dẫn: `np.linalg.matrix_rank(A)`)

Câu 10: Để phân tách ma trận A thành tích 2 ma trận tam giác dưới L và tam giác trên U . Lệnh nào dưới đây hợp lệ:

- a. `numpy.lu(A)`
- b. `numpy.linalg.lu(A)`
- c. `numpy.linalg.matrix_lu(A)`
- d. Lệnh khác

(đáp án: d lệnh khác: **`scipy.linalg.lu(A)`**)

Câu 11: Ma trận A thành tích 2 ma trận tam giác dưới L và tam giác trên U , sau khi thực hiện lệnh, phương thức để nhận dữ liệu (các ma trận L và U) là:

- a. $L, U = \text{<lệnh tạo } L, U \text{ từ ma trận>}$
- b. $U, L = \text{<lệnh tạo } L, U \text{ từ ma trận>}$
- c. $L, U, I = \text{<lệnh tạo } L, U \text{ từ ma trận>}$

d. Lệnh khác

(đáp án: d lệnh khác: Hướng dẫn: vì lệnh **scipy.linalg.lu(A)** sẽ trả về theo thứ tự là **P, L, U**)

Câu 12: Trong numpy, lệnh để so sánh hai ma trận A và B có bằng nhau không là:

- a. `numpy.equal(A, B)`
- b. `numpy.equals(A,B)`
- c. `numpy.array_equal(A, B)`
- d. `numpy.matrix_equal(A, B)`

(đáp án: c)

Câu 13: Hiện tại, chúng ta có thể sử dụng thư viện sympy trực tuyến tại địa chỉ nào dưới đây:

- a. <https://live.sympy.com>
- b. <https://live.sympy.org>
- c. <https://online.sympy.org>
- d. Địa chỉ khác

(đáp án: B; lưu ý: đề thi có thể thay đổi tên địa chỉ)

Câu 14: Để tìm phương trình mặt phẳng từ 3 điểm trong không gian, gói phần mềm nào có thể hỗ trợ trực tiếp câu lệnh cho chúng ta:

- a. Thư viện numpy
- b. Thư viện scipy
- c. Thư viện sympy
- d. Thư viện math mặc định của Python.

(đáp án: c; vì sympy hỗ trợ tính toán hình thức. Lưu ý: thứ tự và tên phần mềm có thể thay đổi)

Câu 15: Hiện tại, lệnh solve để giải phương trình được trang bị ở gói thư viện nào?

- a. Thư viện numpy
- b. Thư viện scipy
- c. Thư viện sympy
- d. Cả 3 thư viện trên.

(Đáp án: c; chỉ có thư viện sympy mới có lệnh solve giải phương trình)

Câu 16: Kiểu dữ liệu số nào của các phần tử ma trận làm cho ma trận $.H$ và $.T$ của ma trận đó khác nhau?

- a. Số thực, số phức.
- b. Số nguyên, số thực, số phức.
- c. Số phức.
- d. $.H$ và $.T$ giống nhau.

(Đáp án: c)

Hướng dẫn: ví dụ:

```
>>> import numpy as np
```

```
>>> A = np.matrix([[1+2j, 3+5j],[2+4j, 9]])
```

```
>>> A.H
```

```
matrix([[1.-2.j, 2.-4.j],  
        [3.-5.j, 9.-0.j]])
```

```
>>> A.T
```

```
matrix([[1.+2.j, 2.+4.j],  
        [3.+5.j, 9.+0.j]])
```

```
>>> B = np.matrix([[1, 3],[2, 9]])
```

```
>>> B.H
```

```
matrix([[1, 2],  
        [3, 9]])
```

```
>>> B.T
```

```
matrix([[1, 2],  
        [3, 9]])
```

```
>>>
```

B. PHẦN II –THỰC HÀNH TÍNH TOÁN ĐẠI SỐ TUYẾN TÍNH

Là những câu hỏi yêu cầu thực hành tính toán đại số tuyến tính để tìm ra một giá trị:

Câu 1: Cho 2 ma trận sau, ma trận nào có hạng (rank) cao hơn:

A = `np.array([[1,1,0],[0,1,1],[1,0,1]])`

B = `np.array([[1,1,0],[1,1,1],[0,0,1]])`

- a. Rank(A) > rank(B)
- b. Rank(A) < rank(B)
- c. Rank(A) = rank(B)
- d. Dạng dữ liệu trên không thể tính toán được hạng!

(đáp án: C)

(Hướng dẫn giải: lệnh xử lý: `np.linalg.matrix_rank(A)`, gói `linalg` này thuộc về `numpy` # khi thi không cho biết lệnh; khi thi các ma trận có thể thay đổi)

Câu 2: Tìm ma trận L và U cho ma trận sau: $A = \begin{bmatrix} 10 & 11 \\ 3 & -5 \end{bmatrix}$

- a. $L = \begin{bmatrix} 1 & 0 \\ 0.3 & 1 \end{bmatrix}$ và $U = \begin{bmatrix} -10 & 11 \\ 0 & -8.3 \end{bmatrix}$
- b. $L = \begin{bmatrix} 1 & 0 \\ 0.3 & 1 \end{bmatrix}$ và $U = \begin{bmatrix} 10 & 11 \\ 0 & -8.3 \end{bmatrix}$
- c. $L = \begin{bmatrix} 1 & 0 \\ -0.3 & 1 \end{bmatrix}$ và $U = \begin{bmatrix} -10 & 11 \\ 0 & -8.3 \end{bmatrix}$
- d. $L = \begin{bmatrix} 1 & 0 \\ -0.3 & 1 \end{bmatrix}$ và $U = \begin{bmatrix} 10 & 11 \\ 0 & -8.3 \end{bmatrix}$

(Đáp án: b)

(Hướng dẫn giải: lệnh xử lý: `P, L, U = linalg.lu(A)`, gói `linalg` thuộc `scipy` # khi thi không cho biết lệnh và ma trận có thể thay đổi)

Câu 3: Tìm giá trị định thức của ma trận U với U được xác định là ma trận trên $LU = A$, $A = \begin{bmatrix} 10 & 11 \\ 3 & -5 \end{bmatrix}$:

- a. 0
- b. -8.3
- c. 10
- d. Một giá trị khác

(đáp án: d, = -83)

(Hướng dẫn giải:

```
>>> A = np.array([[10, 11],[3, -5]]) # đề thi sẽ không có dòng này mà chỉ mô tả ma trận A
>>> P, L, U = scipy.linalg.lu(A) # đề thi sẽ không có dòng này)
```

Câu 4: Giải phương trình $Ax=b$, với $A = \begin{bmatrix} 10 & 11 \\ 3 & -5 \end{bmatrix}$ và $b=\begin{bmatrix} 2 \\ 3 \end{bmatrix}$. Chúng ta được nghiệm:

- a. $\begin{bmatrix} 0.41807229 & -0.28915663 \end{bmatrix}$
- b. $\begin{bmatrix} 0.51807229 & -0.28915663 \end{bmatrix}$
- c. $\begin{bmatrix} -0.28915663 & 0.41807229 \end{bmatrix}$
- d. $\begin{bmatrix} -0.28915663 & 0.51807229 \end{bmatrix}$

(Đáp án: b)

Câu 5: Ma trận nào dưới đây hội tụ? Hội tụ được định nghĩa là $A^k=A'$ khi k tiến đến vô cùng.

- a. $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
- b. $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
- c. $\begin{bmatrix} 0 & -1 \\ 1 & 2 \end{bmatrix}$
- d. $\begin{bmatrix} 0 & 0.5 \\ 1 & 0.5 \end{bmatrix}$

(đáp án: a, d)

Câu 5: Ma trận nào dưới đây phân kỳ? (Ma trận phân kỳ được định nghĩa như sau: $A^{2k+1}=A^{2k+3} \nRightarrow A^{2k}=A^{2k+2}$)

- a. $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
- b. $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
- c. $\begin{bmatrix} 0 & -1 \\ 1 & 2 \end{bmatrix}$
- d. $\begin{bmatrix} 0 & 0.5 \\ 1 & 0.5 \end{bmatrix}$

(đáp án: b, c; tuy nhiên lưu ý: b: $A^k=A^{k+2} \nRightarrow A^{k+1}=A^{k+3}$; còn c là ma trận thay đổi mãi)

Câu 6: Tính giá trị A^{1000} cho ma trận sau: $A = \begin{bmatrix} 0.3 & 0.8 \\ 0.7 & 0.2 \end{bmatrix}$

- a. $A^{1000} = \begin{bmatrix} 0 & 9.85 \\ 8.7 & 0 \end{bmatrix}$
- b. $A^{1000} = \begin{bmatrix} 0.47 & 0.47 \\ 0.53 & 0.53 \end{bmatrix}$
- c. $A^{1000} = \begin{bmatrix} 8.7 & 0 \\ 0 & 9.85 \end{bmatrix}$
- d. $A^{1000} = \begin{bmatrix} 0.53 & 0.53 \\ 0.47 & 0.47 \end{bmatrix}$

(đáp án: d, lưu ý: lệnh nhân là $A.dot(A)$)

Câu 7: Hãy tính giá trị số Fibonacci thứ 20, 21, 22. Biết rằng chuỗi Fibonacci bắt đầu theo thứ tự thứ tự 0 (F_0): là số 0; $F_1 = 1$; $F_2 = 1$; $F_3 = 2$; $F_4 = 3$; $F_5 = 5$, ..., $F_{n+1} = F_n + F_{n-1}$.

- a. $F_{20} = 10946$; $F_{21} = 17711$; $F_{22} = 28657$;
- b. $F_{20} = 6765$; $F_{21} = 10946$; $F_{22} = 17711$;
- c. $F_{20} = 4181$; $F_{21} = 6765$; $F_{22} = 10946$;
- d. Một bộ giá trị khác.

(đáp án: b)

Câu 8: Một phân số $r = \frac{p_n}{q_n}$ được biểu diễn dưới dạng liên phân số như sau:

$$\frac{p_n}{q_n} = c_0 + \frac{1}{c_1 + \frac{1}{c_2 + \frac{1}{\ddots + \frac{1}{c_n}}}}$$

Người ta chứng minh được rằng có một cách xác định p_n và q_n như sau:

$$\begin{pmatrix} c_0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} c_1 & 1 \\ 1 & 0 \end{pmatrix} \dots \begin{pmatrix} c_n & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} p_n & p_{n-1} \\ q_n & q_{n-1} \end{pmatrix}, n = 0, 1, 2, \dots$$

Cho dãy số $c_0 = 3, c_1 = 7, c_2 = 15, c_3 = 1, c_4 = 292, c_5 = c_6 = c_7 = 1$. Giá trị của r gần giống với giá trị của số nào dưới đây nhất:

- a. 3.1415926536
- b. 3.1415926536189
- c. 3.1415926536189365
- d. 3.14159265361893653

(đáp án c. Hướng dẫn: đây là cách tính số Pi)

Code hỗ trợ tính toán:

```
>>> import numpy as np

>>> c = [3,7,15,1,292,1,1,1]

>>> M = np.mat([[1,2],[3,4]]) # khởi tạo ma trận 2x2

>>> for i in range(len(c)):

    ci = np.mat([[1,1],[1,0]])

    ci[0][0] = c[i]
```



```

    if (i==0):
        M = ci
    else:
        M = M.dot(ci)

>>> print(M)

>>> print (M[0,0]/M[1,0])

3.1415926536189365

```

Câu 9: Cho hai ma trận A, B có kích thước 3x3. Quy trình nào để tính toán $(AB)^{-1}$ dưới đây là đúng bằng thư viện numpy/scipy:

- $C = A^{-1} \rightarrow D = B^{-1} \rightarrow E = CD \rightarrow$ kết quả là E.
- $C = B^{-1} \rightarrow D = A^{-1} \rightarrow E = CD \rightarrow$ kết quả là E.
- $C = AB \rightarrow D = C^{-1} \rightarrow$ kết quả là D.
- $C = BA \rightarrow D = C^{-1} \rightarrow$ kết quả là D.

(đáp án: câu b và c; lưu ý: $(AB)^{-1} = B^{-1}A^{-1}$)

Câu 10: Cho ma trận $A = \begin{bmatrix} 3,5,-8 \\ 4, 12, -1 \\ 2,5,3 \end{bmatrix}$. Ma trận $A2 = A*2$. Hãy tính định thức 2 ma trận trên:

- Định thức: $A=75$, định thức $A2 = 150$
- Định thức: $A=75$, định thức $A2 = 600$
- Định thức: $A=85$, định thức $A2 = 170$
- Định thức: $A=85$, định thức $A2 = 680$

(Đáp án: d; lưu ý: $\det(2*A) = 2^n \det(A)$, tính det bằng lệnh: `>>> np.linalg.det(A)`)

C. PHẦN III – ỨNG DỤNG ĐẠI SỐ TUYẾN TÍNH

Là những câu hỏi ứng dụng đại số tuyến tính. Các câu hỏi này liên quan đến việc sử dụng đại số tuyến tính, đôi khi cần tính toán, chủ yếu là các kiến thức đại số tuyến tính (lý thuyết):

Câu 1: Tìm giá trị định thức của ma trận L với L được xác định là ma trận dưới thỏa $LU = A$, với $A = \begin{bmatrix} 10, 11 \\ 3, -5 \end{bmatrix}$

- 0
- 0.3
- 1

d. Ma trận A không tính được ma trận L.

(đáp án: c, ma trận L luôn có định thức bằng 1! Hướng dẫn:

```
>>> A = np.array([[10, 11],[3, -5]]) # đề thi sẽ không có dòng này mà chỉ mô tả ma trận A
```

```
>>> P, L, U = linalg.lu(A) # đề thi sẽ không có dòng này)
```

Câu 2: Chọn biểu thức nào dưới đây có kết quả là True, với ma trận $A = \begin{bmatrix} 10 & 11 \\ 3 & -5 \end{bmatrix}$:

- a. $\text{linalg.det}(3*A) == \text{linalg.det}(A) * \text{pow}(3,2)$
- b. $\text{linalg.det}(3*A) == \text{linalg.det}(A) * \text{pow}(2,3)$
- c. $\text{linalg.det}(3*A) == 3 * \text{linalg.det}(A)$
- d. $\text{linalg.det}(3*A) == \text{pow}(\text{linalg.det}(A), 3)$

(đáp án: a, vì $\det(kA) = (k^n)\det(A)$ với A là ma trận $n \times n$)

(Hướng dẫn: trước đó khai báo `>>> A = np.array([[10, 11],[3, -5]])` # đề thi sẽ không có dòng này mà chỉ mô tả ma trận A)

Câu 3: Cho ma trận $A = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$. Sau khi tính: A^{100} , giá trị của ma trận $B=A^{100}$ như sau (xét chỉ số ma trận bắt đầu từ 1, nghĩa là ma trận gồm 4 phần tử bắt đầu từ dòng 1, cột 1):

- a. $B_{21} = B_{12}$
- b. $B_{11} = B_{21} + B_{22}$
- c. $B_{12} = B_{21} + B_{22}$
- d. $B_{11} = B_{12} + B_{22}$

(đáp án: a, b, d: lưu ý: đây là chuỗi Fibonacci: $[[F_{n+1}, F_n], [F_n, F_{n-1}]]$)

Câu 4: Cho A là ma trận truyền tin $A_{ij} = 1$ nghĩa là từ i truyền được tin sang j, nếu không truyền được thì $A_{ij}=0$. Cho biết $A = ((0,1,0,1),(0,0,1,0),(1,0,0,1),(1,1,0,0))$. Hỏi số lượng thông tin người thứ $i=3$ có thể nhận được từ người thứ $j=2$ thông qua nhiều nhất 1 người trung gian là bao nhiêu?

Giải thích: Ma trận A gồm 4 dòng và 4 cột tương ứng với 4 người. Trong đó, dòng đầu tiên của ma trận (0,1,0,1) nghĩa là người thứ 1 có thể truyền trực tiếp cho người 2 và 4.

- a. 0
- b. 1
- c. 2

d. 3

(Đáp án: 2)

(Hướng dẫn: xem Bài toán truyền tin, tính toán A^2 , giá trị $A^2[3][2] = 2$)

Câu 5: Mô tả ngắn về định thức V để tính thể tích hình tứ diện trong không gian với 4 điểm trong không gian $(x_1, y_1, z_1), (x_2, y_2, z_2), (x_3, y_3, z_3), (x_4, y_4, z_4)$.

- a. V là $1/3$ định thức của ma trận 4×4 , mỗi dòng lần lượt là các x_i, y_i, z_i và dòng cuối là số 1.
- b. V là $1/3$ định thức của ma trận 4×4 , mỗi cột lần lượt là các x_i, y_i, z_i và cột cuối là số 1.
- c. V là $1/6$ định thức của ma trận 4×4 , mỗi dòng lần lượt là các x_i, y_i, z_i và dòng cuối là số 1.
- d. V là $1/6$ định thức của ma trận 4×4 , mỗi cột lần lượt là các x_i, y_i, z_i và cột cuối là số 1.

(đáp án: d)

Câu 6: Đa giác lồi P có 4 đỉnh lần lượt theo thứ tự $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$. Diện tích S của đa giác P là:

- a. $S = 0.5 * [(x_1 y_2 - x_2 y_1) + (x_2 y_3 - x_3 y_2) + (x_3 y_4 - x_4 y_3)]$
- b. $S = 0.5 * [(x_1 y_2 - x_2 y_1) + (x_2 y_3 - x_3 y_2) + (x_3 y_4 - x_4 y_3) + (x_4 y_1 - x_1 y_4)]$
- c. $S = 0.5 * [(x_1 y_2 + x_2 y_1) + (x_2 y_3 + x_3 y_2) + (x_3 y_4 + x_4 y_3)]$
- d. $S = 0.5 * [-(x_1 y_2 - x_2 y_1) - (x_2 y_3 - x_3 y_2) - (x_3 y_4 - x_4 y_3) - (x_4 y_1 - x_1 y_4)]$

(Đáp án: b hoặc d)

Câu 7: Cho tập hợp $A = \{1, 2, 3, 4, 5, 6, 7\}$. Dấu (sgn) của phép thế $\sigma = \{7, 6, 5, 4, 3, 2, 1\}$ là:

- a. 1
- b. -1
- c. 0
- d. Giá trị khác.

(Đáp án: b)

D. PHẦN IV – TÍNH TOÁN

Câu 1: Tìm ma trận nghịch đảo của ma trận $A = \begin{bmatrix} -2 & 1 \\ 3/2 & -0.5 \end{bmatrix}$

- a. $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$
- b. $\begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$
- c. $\begin{bmatrix} 1 & 3 \\ 3 & 4 \end{bmatrix}$
- d. Đáp án khác

Đáp án: a $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$

Hướng dẫn: Để tính được, sinh viên phải sử dụng các lệnh sau

```
>>> import numpy as np
```

```
>>> from numpy import linalg
```

```
>>> A = np.array([[-2, 1], [3/2, -0.5]]) # hoặc >>> A = np.matrix([[-2, 1], [3/2, -0.5]])
```

```
>>> linalg.inv(A)
```

Câu 2: Dòng đầu tiên của nghịch đảo ma trận sau: $B = \begin{bmatrix} 0 & -1 & 2 & 3 \\ -4 & 5 & 3 & 2 \\ 1 & 6 & -7 & 2 \\ 3 & 9 & 2 & 1 \end{bmatrix}$ là:

- a. [-0.06813757, 0.05126541, 0.02011681, 0.06164828]
- b. [0.08500973, -0.16872161, -0.0155743 , 0.11356262]
- c. [0.03569111, 0.02076574, -0.10577547, 0.06294614]
- d. Giá trị khác.

Đáp án: b.

Hướng dẫn: Quy trình tính như câu 1. Ma trận nghịch đảo của B là:

```
>>> B = np.array([[-0, -1, 2, 3], [-4, 5, 3, 2], [1, 6, -7, 2], [3, 9, 2, 1]])
```

```
>>> linalg.inv(B)
```

```
array([[ 0.08500973, -0.16872161, -0.0155743 , 0.11356262],  
       [-0.06813757, 0.05126541, 0.02011681, 0.06164828],  
       [ 0.03569111, 0.02076574, -0.10577547, 0.06294614],  
       [ 0.28682674, 0.00324465, 0.07722258, -0.02141467]]))
```

Câu 3: Tính giá trị của định thức $B = \begin{bmatrix} 0 & -1 & 2 & 3 \\ -4 & 5 & 3 & 2 \\ 1 & 6 & -7 & 2 \\ 3 & 9 & 2 & 1 \end{bmatrix}$

- a. 1040.99999
- b. 1540.99999
- c. 1650.99999
- d. 1840.99999

Đáp án: b.

Hướng dẫn: sử dụng thư viện linalg như câu 2 và 3. Tính định thức bằng hàm linalg.det()

```
>>> B = np.array([[0, -1, 2, 3], [-4, 5, 3, 2], [1, 6, -7, 2], [3, 9, 2, 1]])
```

```
>>> linalg.det(B)
```

```
1540.99999999999982
```

Câu 4: Giải hệ phương trình sau:

$$x_1 - x_2 - 2x_3 = 1$$

$$2x_1 - 3x_2 - 5x_3 = 3$$

$$-x_1 + 3x_2 + 5x_3 = -2$$

Nghiệm là:

- a. $x_1 = 1; x_2 = -2; x_3 = 1$
- b. $x_1 = 1; x_2 = -1; x_3 = 0.5$
- c. $x_1 = 2; x_2 = 1; x_3 = -1$
- d. Nghiệm khác

Đáp án: câu a.

Hướng dẫn: Sinh viên giải bằng các lệnh sau:

```
>>> import numpy as np
```

```
>>> A = np.array([[1,-1,-2],[2,-3,-5],[-1,3,5]])
```

```
>>> b = np.array([1,3,-2])
```

```
>>> from numpy import linalg
```

```
>>> linalg.solve(A,b)
```

Câu 5: Tìm phương trình ellipse biết rằng ellipse đi qua 5 điểm sau: (10, 0), (-10, 0), (4, 2), (-4, 2), (4, -2):

- a. $x^2/100 + y^2/(100/21) = 1$
- b. $5120x^2 + 107520y^2 - 512000 = 0$
- c. $5120x^2 + 507520y^2 - 112000 = 0$
- d. Phương trình khác.

Đáp án: a và b vì b rút gọn, chia 2 về cho 5120 và tiếp tục chia cho 100 sẽ ra a.

Hướng dẫn: Sinh viên có thể thử qua các lệnh sau

Module tính phương trình ellipse:

```
>>> import sympy as sp

>>> def pt_ellip(xi, yi):

    x, y = sp.symbols('x y')

    x1, x2, x3, x4, x5 = xi[0],xi[1],xi[2],xi[3],xi[4]

    y1, y2, y3, y4, y5 = yi[0],yi[1],yi[2],yi[3],yi[4]

    ellip = sp.Matrix([ [x**2, x*y, y**2,x,y,1],

                        [x1*x1, x1*y1, y1*y1,x1,y1,1],

                        [x2*x2, x2*y2, y2*y2,x2,y2,1],

                        [x3*x3, x3*y3, y3*y3,x3,y3,1],

                        [x4*x4, x4*y4, y4*y4,x4,y4,1],

                        [x5*x5, x5*y5, y5*y5,x5,y5,1]])

    return ellip.det()
```

```
>>> xi = [10,-10, 4, -4, 4]
```

```
>>> yi = [ 0, 0, 2, 2, -2]
```

```
>>> print(pt_ellip(xi, yi))
```

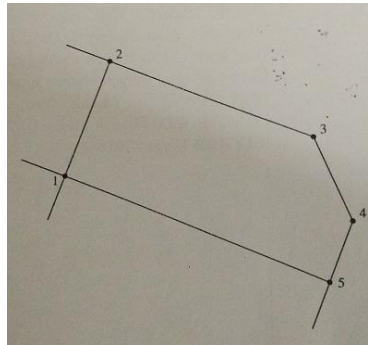
Kết quả ra câu b. Sau đó, sinh viên có thể tìm ước số chung để đơn giản hóa phương trình:

```
>>> import math
```

```
>>> math.gcd(5120, 107520) # ra 5120. Lấy 107520/5120 = 21
```

Suy ra phương trình: $x^2 + 21.y^2 = 100$

Câu 7: Diện tích của thửa đất trong một giấy chứng nhận dưới đây là bao nhiêu mét vuông?



BẢNG KÊ TỌA ĐỘ		
Điểm	X(mét)	Y(mét)
1	1181128.25	603263.70
2	1181136.18	603266.68
3	1181130.97	603279.69
4	1181125.77	603281.91
5	1181122.06	603280.42

- a. 116 m²
- b. 130 m²
- c. 164 m²
- d. Diện tích khác

Đáp án: diện tích khác ?

Hướng dẫn: xem bài 4 về tính diện tích:

```
>>> X = [128.25, 136.18, 130.97, 125.77, 122.06, 128.25]
```

```
>>> Y = [63.70, 66.68, 79.69, 81.91, 80.42, 63.70]
```

```
>>> S = 0
```

```
>>> for i in range(len(X)-1):
```

```
    S = S + (X[i]*Y[i+1]-X[i+1]*Y[i])
```

```
>>> print (S/2.0)
```

```
139.56789999999955
```