BÁO CÁO:

**BÀI TẬP GIỮA KÌ**

*Môn học:* Một số vấn đề chọn lọc trong tính toán khoa học

*Giảng viên:* Hà Phi

*Nhóm 1:*

Trịnh Thị Yến

Bùi Thị Phượng

Lê Thành Nam

Nguyễn Thị Ngọc Như

Contents

[**Bài 1:** 3](#_Toc72441228)

[**Bài 2:** lập trình Octave 6](#_Toc72441229)

[**Bài 3:** 6](#_Toc72441230)

[**Bài 4:** 9](#_Toc72441231)

**Bài 1:** Ví dụ 2.8

Xét một xe đẩy có gán một con lắc ngược bản lền như hình 2.12.

Để đơn giản, xe đẩy và con lắc được giả định chỉ chuyển động trong một mặt phẳng và không tính ma sát, khối lượng của thanh và sức gió.

Vấn đề là giữ nguyên con lắc ở vị trí thẳng đứng. Ví dụ, nếu con lắc ngược đang rơi theo hướng như trên, xe đẩy sang phải và tác dụng một lực, quan bản lề, để đẩy con lắc trở lại vị trí thẳng đứng. Cơ chế đơn giản này có thể được sử dụng như một mô hình của một phương tiện không gian đang cất cánh.

Gọi H và V, lần lượt là lực ngang và dọc do xe đẩy tác dụng lên con lắc như hình vẽ. Áp dụng định luật Newton cho chuyển động thẳng thu được :

M= u -H

H = m(y+msinθ) = + cos - sinθ

mg – V = m(sinθ)= [ - - cos

Việc áp dụng định luật Newton vào chuyển động quay của con lắc quanh bản lề ta được:

mgsinθ = .

chúng là các phương trình phi tuyến. Vì mục tiêu thiết kế là duy trì con lắc ở vị trí thẳng đứng, nên giả thuyết rất nhỏ. Theo giả thuyết này, chúng ta có thể sử dụng xấp xỉ sinθ= và cos Bằng cách chỉ giữ lại các số hạng tuyến tính trong tương đương, bỏ qua các số hạng với , chúng ta thu được V= mg và

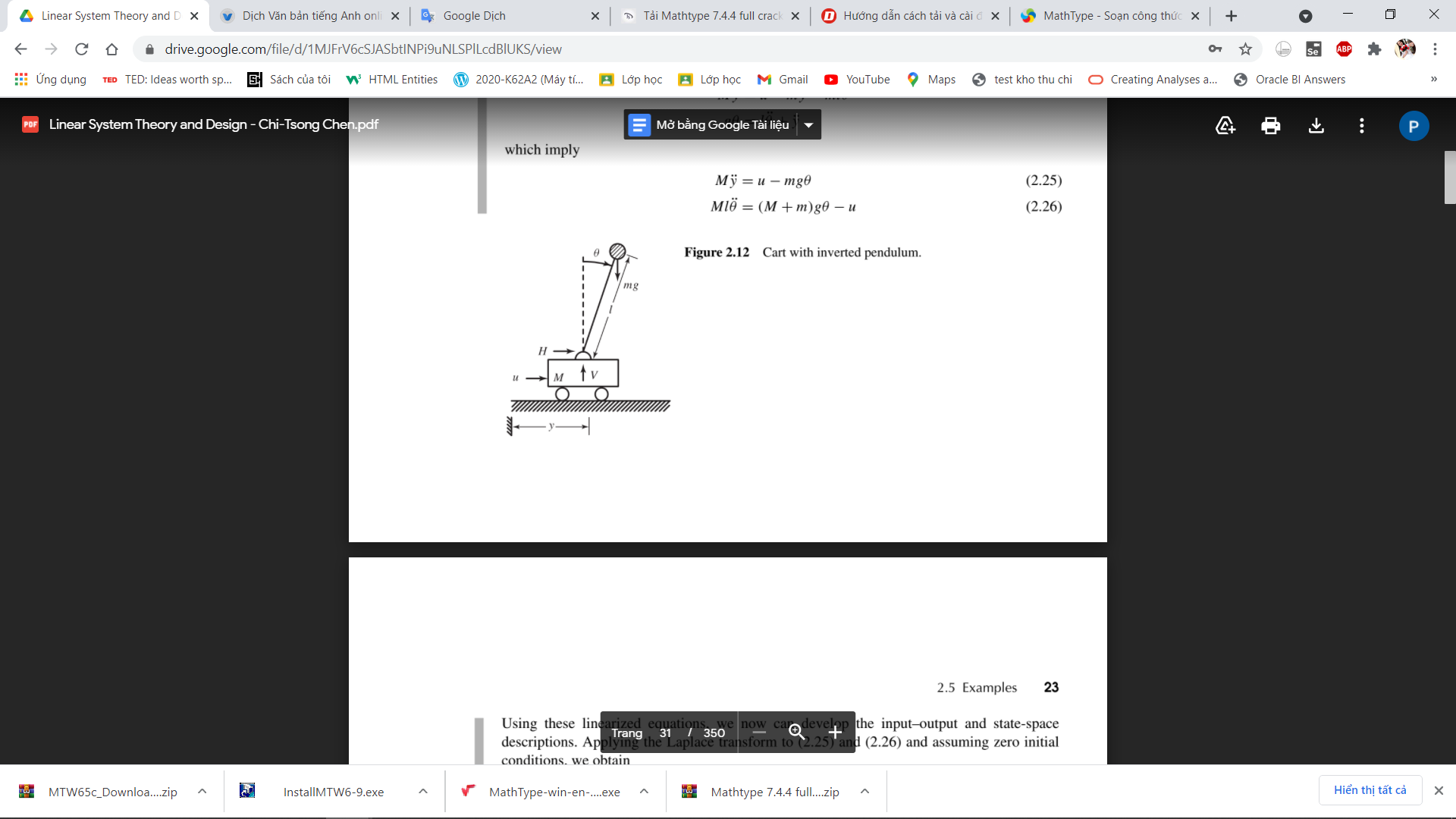
M = u – m –

g= +

**Ngụ ý:**

M = u- mgθ (2.25)

M = (M +m)g – u (2.26)

 Hình 2.12. Xe đẩy với con lắc ngược.

Sử dụng phương pháp tuyến tính hóa, bây giờ chúng ta có thể phát triển các mô tả đầu vào- đầu ra và mô tả không gian trạng thái.

Áp dụng biến đổi Laplace để (2.25) và (2.26) và giả định điều kiện ban đầu bằng không/zero, ta có được:

=

= (M +m) -

Từ những phương trình này, chúng ta có thể dễ dàng tính toán hàm truyền từ u sang y và hàm truyền từ u sang

=

=

Để phát triển một phương trình không gian trạng thái (state space system), chúng ta chọn các biến trạng thái là x1 = y, x2 = , và . Sau đó, từ sự lựa chọn này và các phương trình (2.25) và (2.26) chúng ta dễ dàng có được:

= + u

y= [1 0 0 0]x (2.27)

Phương trình không gian - trạng thái này có kích thước 4 và mô tả hệ thống với rất nhỏ.

**Số chiều của u là 1 x 1**

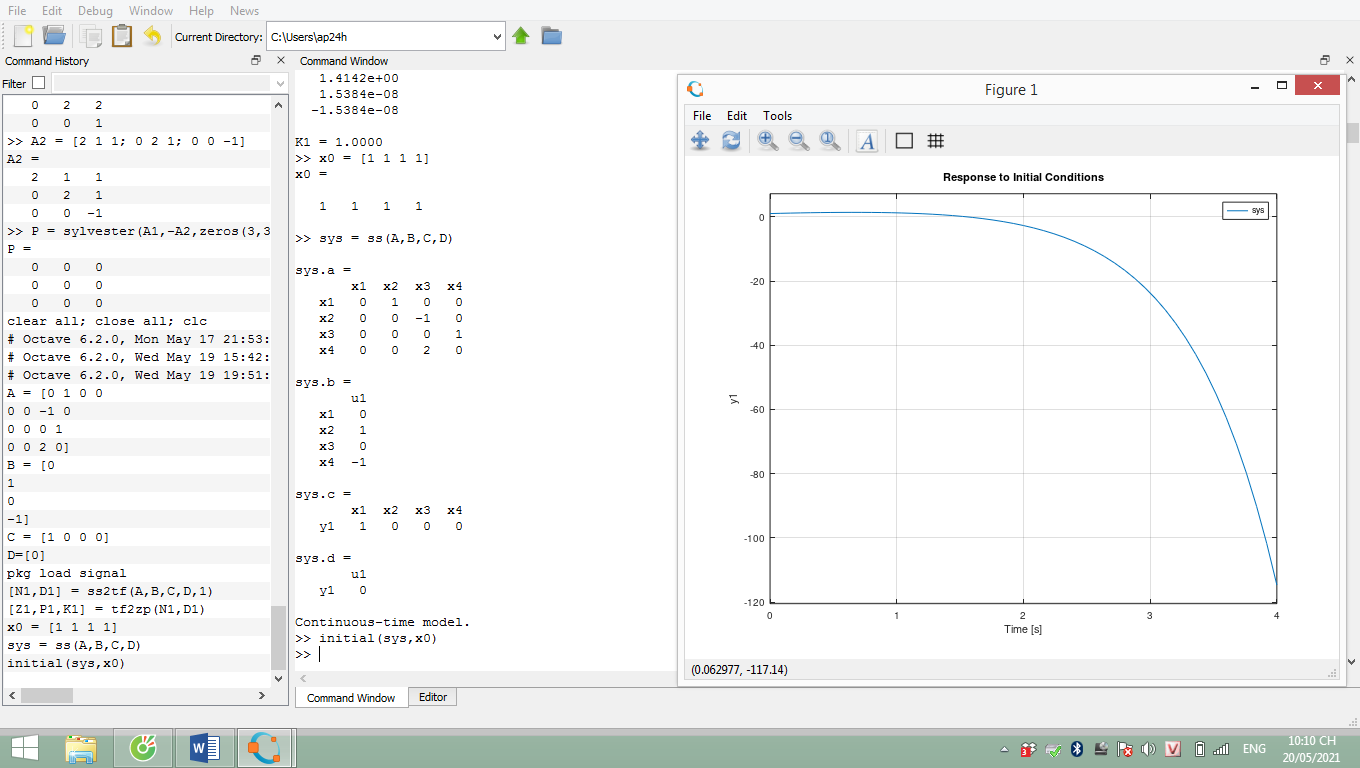
**Số chiều của x là 4 x 1**

**Số chiều của y là 1 x 1**

**Số chiều trong phép x ma trận q x 1 = (q x p) x (p x 1)**

**Vậy số chiều của hàm truyền là 1 x 1**

**Bài 2:** lập trình Octave



N1 = 1.0000e+00 1.1102e-16 -1.0000e+00

D1 = 1.0000e+00 8.8818e-16 -2.0000e+00 -1.1114e-15 4.7335e-16

Hàm truyền trong bài tập này là



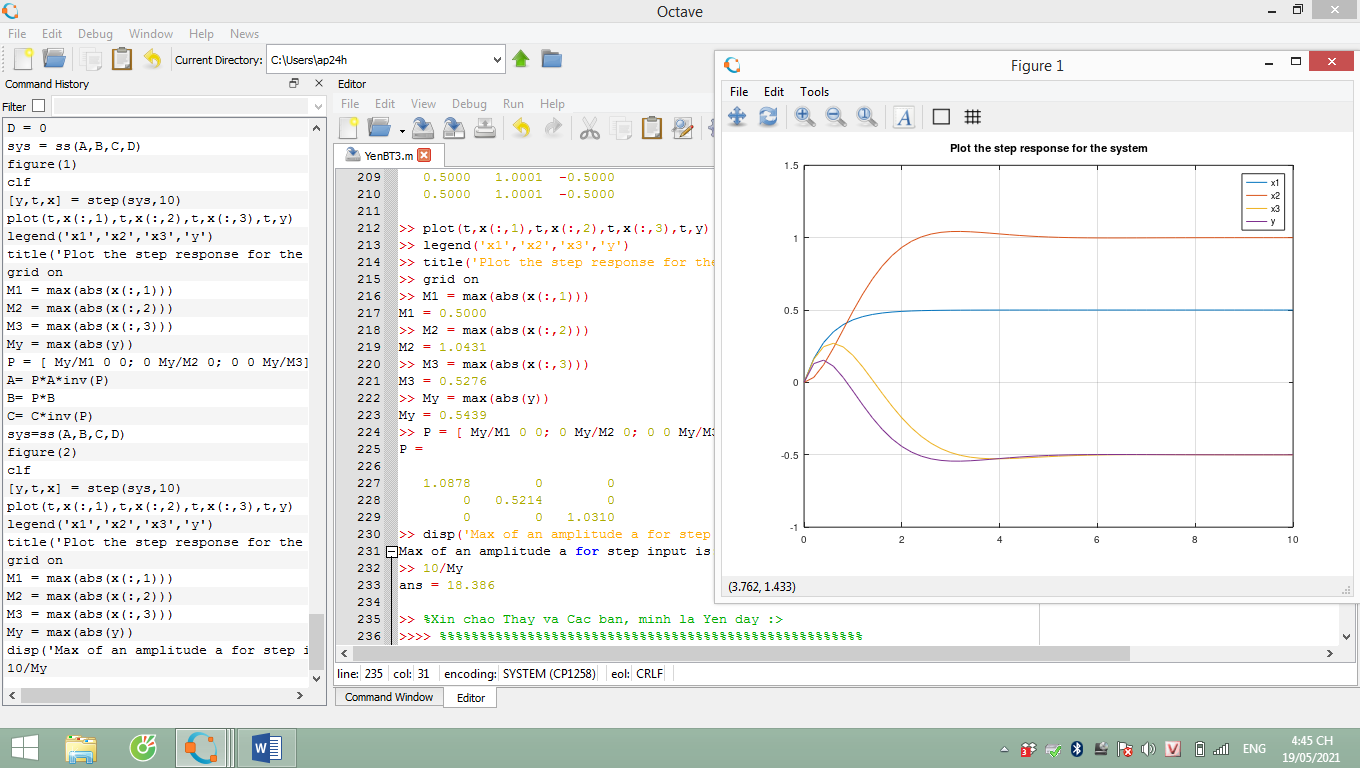
**Bài 3:**

Hãy tìm phép đổi biến số thích hợp để chia tỉ lệ (magnitude scaling) hệ sau, sao cho tất cả các biến trạng thái xi(t) đều có độ lớn bằng với độ lớn tối đa của đầu ra y(t).

= x + u, y = x

Nếu mọi tín hiệu đều phải nằm trong phạm vi ±10 V và nếu hàm đầu vào là hàm bước nhảy (step với độ lớn a) thì a tối đa có thể là bao nhiêu?

// tìm đồ thị [y,x,t] trên Octave



* Thu được:

|x1|max = 0.5

|x2|max = 1.0431

|x3|max = 0.5276

|y|max = 0.5439

Bước nhảy với độ lớn a tối đa là = = 18,386

* Thực hiện phép biến đổi số:

1 = . x1 = . x1 = 1.0878 x1 ≈ x1

2 = . x2 = . x2 = 0.5214 x1 ≈ 0.5x2

3 = . x3 = . x3 = 1.0308 x1 ≈ x3

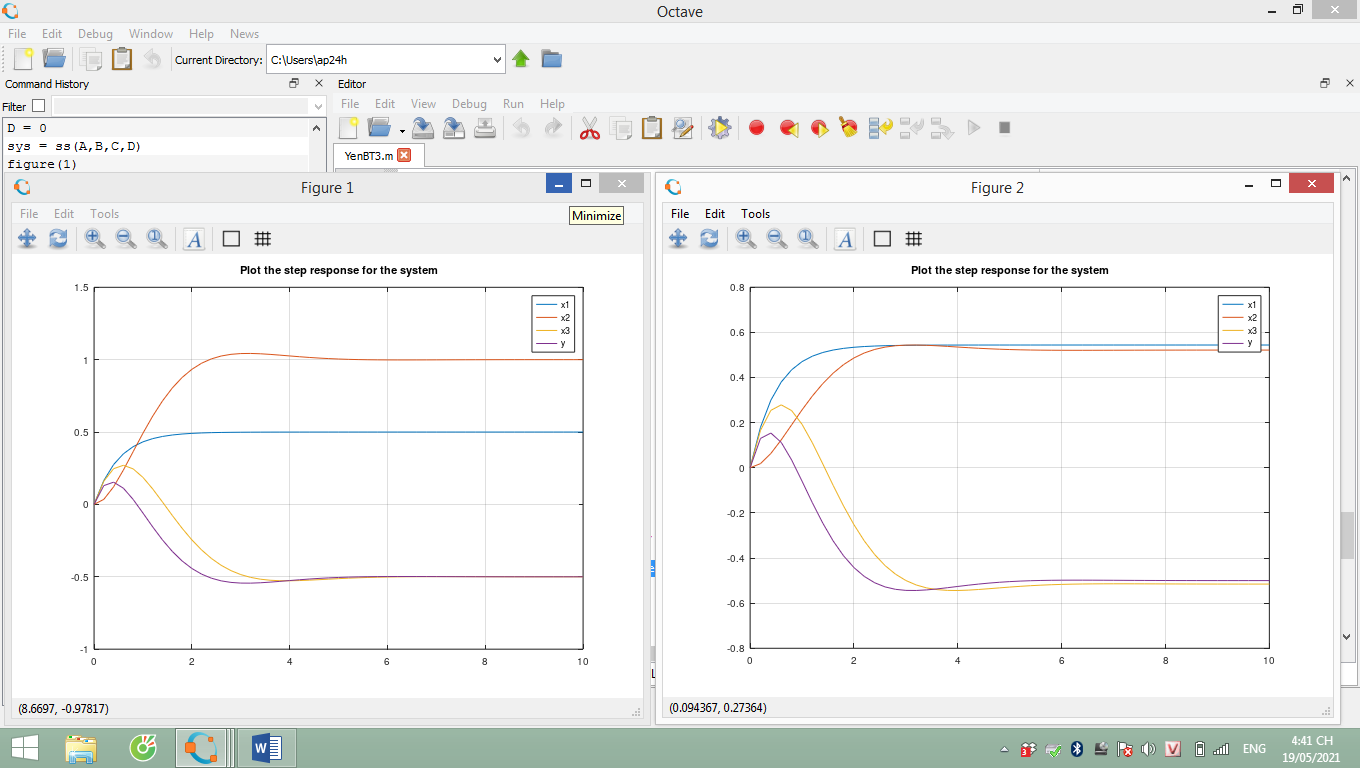
* Do:

1 = -2x1 + u ***=>*** 1 = - 21 + u

2 = x1 + x3 ***=>*** = 1 + 3 ***=>*** 2 = 0.51 + 0.53

3 = -2x2 - 2x3 + u ***=>*** 3 = - 2 3 + u ***=>*** 3 = -42 - 23 + u

y = x1 – x2 = 1 + = 1 + 22

****

**Nhìn hình bên phải, sau này khi khuếch đại u = 18.4 \* step thì tín hiệu x, y đều đạt ngưỡng thỏa mãn đề bài**

**18.4 \* +- 0.6 ~ +-10 V**

**Kết luận: = + u;**

**y = [ 1 -2 0] .**

**Bài 4:**

Hai hệ sau có tương đương hay không?

Có tương đương zero hay không?

= x + u, y= x (1)

Và:

= x + u, y= x (2)

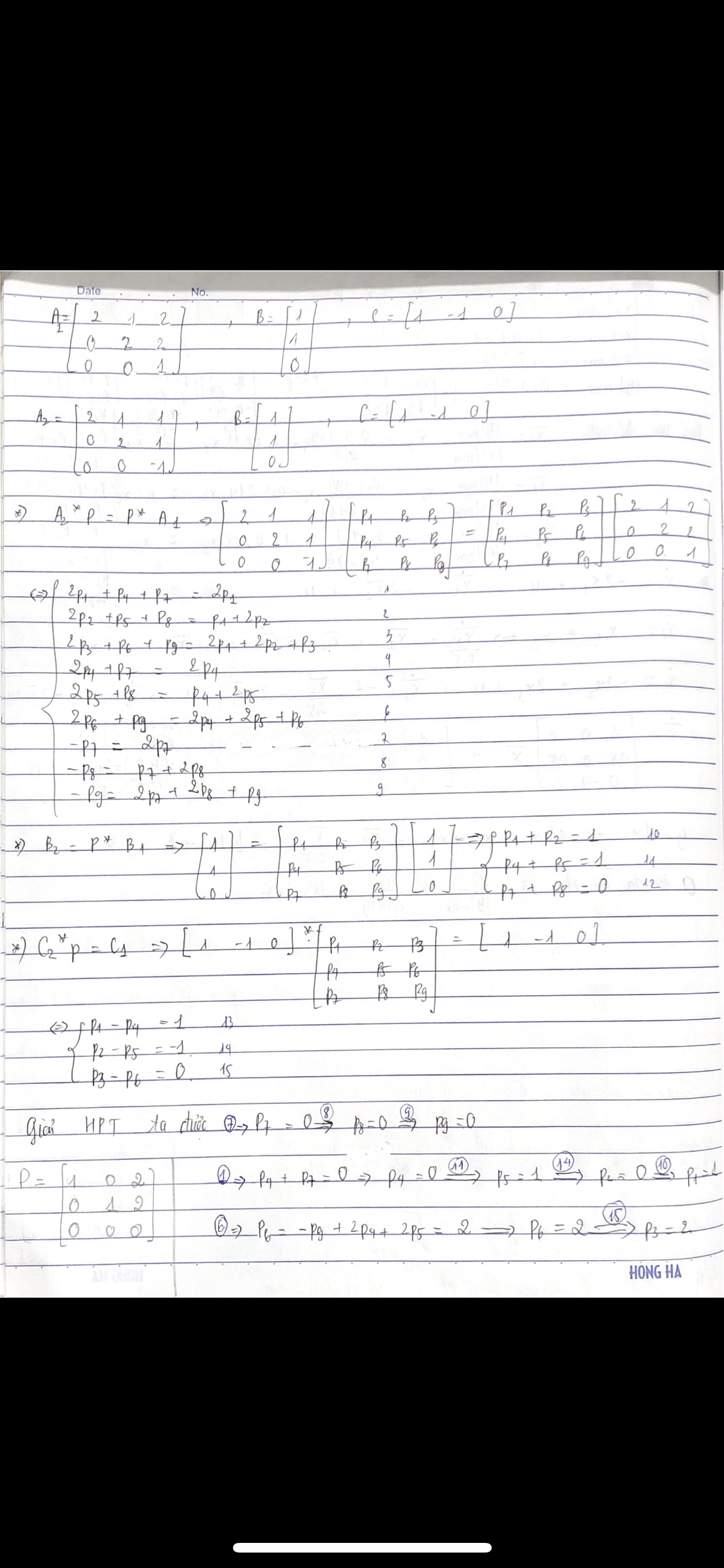
Từ (1):

**A** = ; **B** = ; **C** = ; **D** = 0.

Từ (2):

= ; = ; = ; = 0.

* Kiểm tra 2 hệ tương đương



Hint: Viet lai he dang Mp = m, M cỡ 15 x 9, p cỡ 9 x 1, n cỡ 15 x 1.

Giải hệ dùng M\m và reshape(P,n,n) thì được P

Sau đó test lại xem A2 = P \* A1 \* inv(P); B2 = P \* B1,

C2 = C1 \* inv(P)

Tìm được P =

Det (P) = 0 không tồn tại P-1

* Không thỏa mãn ĐK 2 hệ tương đương:
* Kiểm tra 2 hệ tương đương trạng thái 0

D = ( thỏa mãn)

C. Ai. B . i . với i = 0,1,2 ?

* Với i = 0:

C. A0. B = = 0

. 0 . = = 0

* C. A0. B = . 0 .
* Với i = 1:

C. A1. B = = 1

. 1 . = = 1

* C. A1. B = . 1 .
* Với i = 2:

C. A2. B = = 4

. 2 . = = 4

* C. A2. B = . 2 .
* Thỏa mãn ĐK 2 hệ tương đương trạng thái 0 .

**Kết luận: 2 hệ không tương đương**

**2 hệ có trạng thái tương đương 0 .**

**☺**