Matlab control II

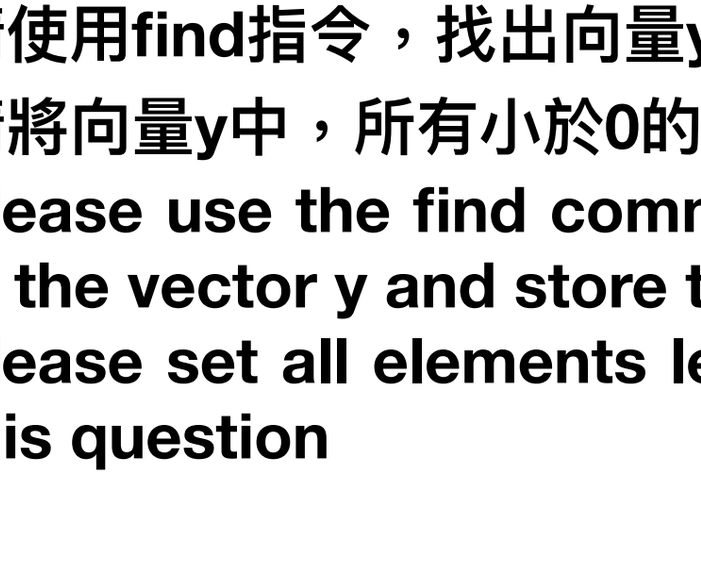
Dice為包含n個整數亂數的向量，亂數值介於1到4之間，本題將dice向量轉換為包含n個字元的gene向量

Dice is a vector containing n integer random numbers. The random value is between 1 and 4. This question converts the dice vector into a gene vector containing n characters.

```
>> n = 2000;
dice = ceil(rand(1, n) * 4);
ind = find(dice == 1);
gene(ind) = "A";
ind = [redacted];
gene(ind) = "T";
ind = find(dice == 3);
gene(ind) = "C";
ind = find(dice == 4);
gene[redacted];
gene(1:10)
dice(1:10)

ans =

    'CTCTTTCCAA'


ans =

     3     2     3     2     2     2     3     3     1     1
```

請使用find指令，找出向量y中所有小於0的元素的位置，並儲存在索引向量ind中
請將向量y中，所有小於0的元素設為0．本題繪製Relu函數
Please use the find command to find the positions of all elements less than 0 in the vector y and store them in the index vector ind.
Please set all elements less than 0 in vector y to 0. Draw the Relu function in this question

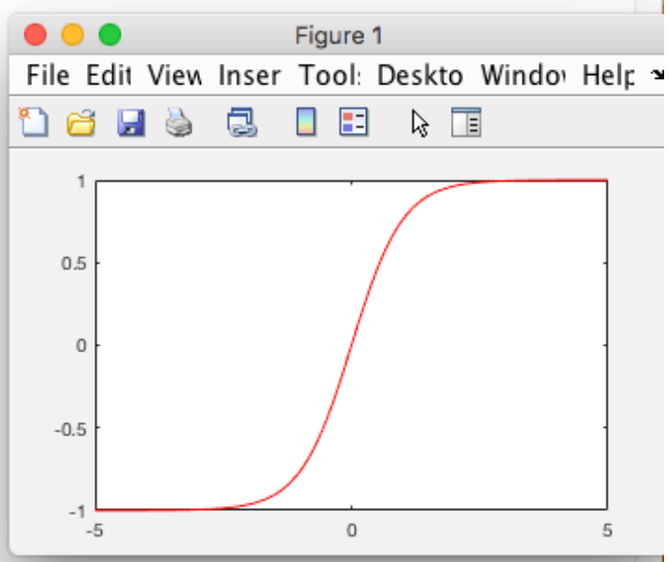
```
x = linspace(-5, 5);
y = x;
ind = [redacted];
[redacted] = 0;
plot(x, y)
```



本題使用switch指令，執行三種不同函數，分別是sign函數，tanh函數，以及Relu函數

This question uses the switch instruction to execute three different functions, namely the sign function, the tanh function, and the Relu function.

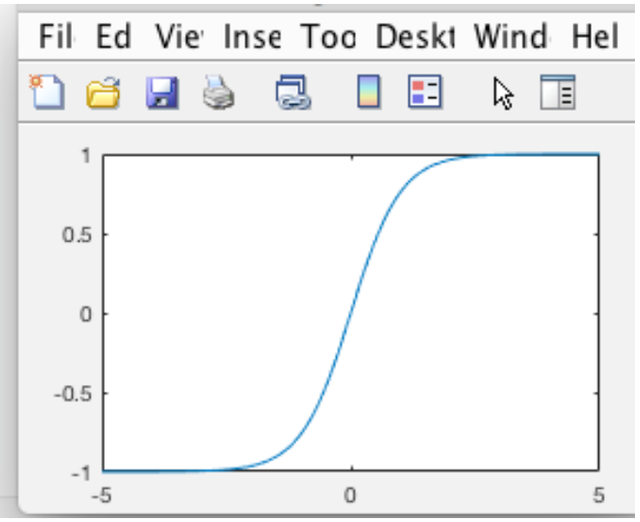
```
x = linspace(-5, 5);
n = 3;
select = ceil(rand * n);
[redacted]
    case 1
        y = sign(x);
    [redacted]
        y = [redacted];
    otherwise
        y = x;
        ind = find(y < 0);
        y(ind) = 0;
end
plot(x, y,'r')
```



本題使用inline指令，建立代表tanh的函數f，並以plot指令，繪製tanh函數

This question uses the inline command to create a function f representing tanh, and uses the plot command to draw the tanh function.

```
x = sym('x');
f =                    ;
a = linspace(-5,5);
plot(       )
>>
```



ss為字串，請使用字串附加，將statement指定為字串”diff(tanh(x))”，將statement代入eval，即可呼叫diff函數，求得tanh函數的一階導數

ss is a string, please use string append, specify the statement as the string "diff(tanh(x))", substitute the statement into eval, you can call the diff function to find the first derivative of the tanh function

```
ss = "tanh(x)"
statement =
(statement)

ss =

    "tanh(x)"


statement =

    "diff(tanh(x))"


ans =

1 - tanh(x)^2
```

dice包含n個骰子點數，本題要找出第100個6點位置．進入while迴圈前，countSix設為100，pos設為0．每進入一次迴圈，pos加1，代表移到下一個位置檢視dice的點數，如果位置pos所對應的點數為6，則變數countSix減一，只要countSix大於而且pos小於n，迴圈就要繼續執行．

```
n = 2000;
dice = ceil(rand(1, n) * 6);
countSix = 100; pos = 0;
while
    pos = pos + 1;
    if          == 6
        countSix =          ;
    end
end
ind = find(dice(1 : pos) == 6);
pos
fprintf('%d * 6 is %d\n',length(ind), sum(dice(ind)));

pos =

   558

100 * 6 is 600
```

Dice contains n dice points. This question is to find the position of the 100th 6. Before entering the while loop, countSix is set to 100 and pos is set to 0. Every time a loop is entered, pos increases by 1, which means moving to the next position to view the points of the dice. If the number of points corresponding to the position pos is 6, the variable countSix is reduced by one. As long as countSix is greater than 0 and pos is less than n, the loop will continue. implement.

demo函數是本題主要執行的函數，程式的檔案名稱可以設定為demo.m．
Append函數是在demo函數範圍內部可以呼叫的函數，功能是附加兩個橫列向量，並傳回附加後的內容．函數輸入參數為a、b，ab代表輸出參數，函數append執行結束後，回傳輸出參數內容給呼叫指令
The demo function is the main function executed in this question. The file name of the program can be set to demo.m.
The append function is a function that can be called within the scope of the demo function. Its function is to append two row vectors and return the appended content. The input parameters of the function are a and b, and ab represents the output parameter. After the function append is executed, the content of the output parameters is returned to the calling command.

```
1   function demo()
2     u = 1 : 3;
3     v = 4 : 6;
4     uv =
5
6   function ab = append(a, b)
7     ab =    ;
```

```
>> demo

uv =

     1     2     3     4     5     6

>>
```

demo函數是本題主要執行的函數，請在第四行呼叫函數innerProduct，計算向量u與向量v的內積．
innerProduct函數是在demo函數範圍內部可以呼叫的函數，功能是計算兩個橫列向量內積，並傳回內積值，請在函數內使用矩陣乘法與transpose指令，求a向量與b向量的內積

The demo function is the main function of this question. Please call the function innerProduct on the fourth line to calculate the inner product of vector u and vector v. The innerProduct function is a function that can be called within the scope of the demo function. Its function is to calculate the inner product of two column vectors and return the inner product value. Please use the matrix multiplication and transpose instructions within the function to find the inner product of the a vector and the b vector.

```
1   function demo()
2       u = 1 : 3;
3       v = 4 : 6;
4       p =
5
6   function c = innerProduct(a, b)
7       c =                        ;
```

```
>> demo

p =

    32

>>
```

demo函數是本題主要執行的函數，第四行呼叫函數outerProduct，計算向量u與向量v的外積．outerProduct函數是在demo函數範圍內部可以呼叫的函數，功能是計算兩個橫列向量外積，並回傳外積值，請在函數內使用transpose指令與矩陣乘法，求向量a與向量b的外積

The demo function is the main function executed in this question. The fourth line calls the function outerProduct to calculate the outer product of vector u and vector v.
The outerProduct function is a function that can be called within the scope of the demo function. Its function is to calculate the outer product of two column vectors and return the outer product value. Please use the transpose instruction and matrix multiplication within the function to find the outer product of vector a and vector b.

```
1   function demo()
2     u = 1 : 3;
3     v = 4 : 6;
4     p = outerProduct(u, v)
5
6   function c =
7     c =  * b;
```

```
p =

    4    5    6
    8   10   12
   12   15   18
```

demo函數是本題主要執行的函數，第五行呼叫函數product，計算矩陣U與矩陣V的乘積．product函數是在demo函數範圍內部可以呼叫的函數，主要使用巢狀迴圈計算兩個矩陣的乘積，並回傳乘積值，請設定內迴圈的索引範圍，並設計迴圈主體，目的是要將A矩陣的第i個橫列與B矩陣的第j個直列相乘，並將結果儲存在C(i, j)

The demo function is the main function executed in this question. The fifth line calls the function product to calculate the product of the matrix U and the matrix V.
The product function is a function that can be called within the demo function scope. It mainly uses nested loops to calculate the product of two matrices and returns the product value. Please set the index range of the inner loop and design the loop body. The purpose is to multiply the i-th column of matrix A by the j-th column of matrix B, and store the result in C(i, j)

```
 1   function demo()
 2     m = 2; n = 3;
 3     U = reshape(1 : m * n, m, n);
 4     V = reshape(1 : m * n, n, m);
 5     P =
 6     sum(sum(P - U * V))
 7
 8   function C = product(A, B)
 9     [m n] = size(A);
10   for i = 1 : m
11       for j =
12           C(i, j) =                ;
13       end
14   end
```

```
>> demo

P =

    22    49
    28    64


ans =

     0

>>
>>
```

第四行呼叫函數det2，計算2x2矩陣A的行列式．
det2函數是在demo函數範圍內部可以呼叫的函數，主要使用索引取出2x2陣列B的各元素，將左對角乘積值減去右對角乘積值，並將結果儲存在輸出參數ans，回傳給呼叫指令
The fourth line calls the function det2 to calculate the determinant of the 2x2 matrix A.
The det2 function is a function that can be called within the scope of the demo function. It mainly uses the index to retrieve each element of the 2x2 array B, subtracts the right diagonal product value from the left diagonal product value, and stores the result in the output parameter ans, and returns it to call instructions

```
1   function demo()
2     p = randperm(4);
3     A = reshape(p, 2, 2);
4     d =      (A);
5     d - det(A)
6
7   function ans = det2(B)
8     ans =                                                  ;
```

```
>> demo2

ans =

     0

fx >>
```

本題計算3x3矩陣的行列式，從A矩陣的第一列展開，有三個分別對應到A(1，1)、A(1，2)與A(1，3)的2x2矩陣行列式需要計算，例如，第五行計算A(1，2)與刪去第一橫列第二直列後所形成2x2矩陣行列式的乘積，依此方式計算d1與d3值，並在第七行將合併的答案儲存在變數d，代表矩陣A的行列式

This question calculates the determinant of a 3x3 matrix. Expand the first column of matrix A. There are three 2x2 matrix determinants corresponding to A(1, 1), A(1, 2) and A(1, 3) that need to be calculated. For example, the fifth line calculates the product of A(1, 2) and the determinant of the 2x2 matrix formed by deleting the first row and the second column. In this way, the d1 and d3 values are calculated, and the combined answer is stored in the variable on the seventh line d, represents the determinant of matrix A

```
1   function demo()
2     p = randperm(9);
3     A = reshape(p, 3, 3);
4     d1 =                                          ;
5     d2 = A(1, 2) * det2([A(2 : 3, 1) A(2 : 3, 3)]);
6     d3 =                                          ;
7     d =                ;
8     d - det(A)
9
10  function ans = det2(B)
11    ans =                                         ;
12
```

```
>> demo

ans =

     0

>>
>>
>>
>>
```