

1、按要求□行与矩□相关的操作。

(1) 构造两个矩阵。

$$A = \begin{bmatrix} 1 & 2 & 4 \\ 1 & 1 & 1 \\ 1 & 3 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \\ 1 & 2 & 3 \end{bmatrix}$$

% 你的答案

```
A = [1 2 4; 1 1 1; 1 3 1];
```

```
B = [1 2 3; 1 2 3; 1 2 3];
```

% 思考：矩阵 A 能否使用 ones() 并修改特定元素来构造

```
A = ones(3);
```

```
A(1,2) = 2;
```

```
A(1,3) = 4;
```

```
A(3,2) = 3;
```

% 思考：搜索 repmat() 如何使用，并用它来构造矩阵 B

```
B = repmat([1 2 3],3,1)
```

```
B = 3×3
```

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

(2) 将 A 和 B 中所有位置□□的元素相加和相乘，分□□□□ sum_AB 和 dpro_AB。

$$sum_AB = \begin{bmatrix} 2 & 4 & 7 \\ 2 & 3 & 4 \\ 2 & 5 & 4 \end{bmatrix} \quad dpro_AB = \begin{bmatrix} 1 & 4 & 12 \\ 1 & 2 & 3 \\ 1 & 6 & 3 \end{bmatrix}$$

% 你的答案

% 提示：点乘 由点和乘组成 尝试 .*

```
sum_AB = A + B
```

```
sum_AB = 3×3
```

```
2     4     7
2     3     4
2     5     4
```

```
dpro_AB = A .* B
```

```
dpro_AB = 3×3
```

```
1     4    12
1     2     3
1     6     3
```

(3) 将矩□ B □置并□□□ BT， □矩□ A 和 BT 两个矩□相乘并□□□ pro_ABT。

$$BT = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 2 & 2 \\ 3 & 3 & 3 \end{bmatrix} \quad pro_ABT = \begin{bmatrix} 17 & 17 & 17 \\ 6 & 6 & 6 \\ 10 & 10 & 10 \end{bmatrix}$$

% 你的答案

% 提示：在表示矩阵的变量后加一个 ' 试试

BT = B'

BT = 3×3

1	1	1
2	2	2
3	3	3

pro_ABT = A * BT

pro_ABT = 3×3

17	17	17
6	6	6
10	10	10

(4) 将矩阵 A 和 B 横向拼接在一起，得到矩阵 C；纵向拼接在一起，赋值给矩阵 VC。

% 你的答案

% 提示：将两个数组并排放入一个 [] 内。两个数组之间加上 ; 会怎样呢

C = [A B]

C = 3×6

1	2	4	1	2	3
1	1	1	1	2	3
1	3	1	1	2	3

VC = [A; B]

VC = 6×3

1	2	4
1	1	1
1	3	1
1	2	3
1	2	3
1	2	3

% 思考：搜索 cat() 如何使用，并完成上述操作

C = cat(2, A, B)

C = 3×6

1	2	4	1	2	3
1	1	1	1	2	3
1	3	1	1	2	3

VC = cat(1, A, B)

VC = 6×3

1	2	4
1	1	1
1	3	1
1	2	3
1	2	3
1	2	3

(5) 将矩阵 C 第 3 行第 2 列的元素存入 a；将矩阵的第 4 列提取出来存入 b；将矩阵第 1 行的元素每隔 1 个提取出来存入 c。

$$C = \begin{bmatrix} 1 & 2 & 4 & 1 & 2 & 3 \\ 1 & 1 & 1 & 1 & 2 & 3 \\ 1 & 3 & 1 & 1 & 2 & 3 \end{bmatrix}$$

% 你的答案

% 提示: m:n 从 m 到 n; m:s:n 从 m 到 n 步长为 s; : 从开始到结束

```
a = C(3,2)
```

```
a = 3
```

```
b = C(:,4)
```

```
b = 3x1
     1
     1
     1
```

```
c = C(1,1:2:end)
```

```
c = 1x3
     1     4     2
```

% 思考: C(1,[2,3,4,6])会提取出来什么

% [2,4,1,3]

% 思考: C([1,3],[2,5])会提取出来什么

% [2,2;3,2]

(6) 运行指令：

(i) C > 2.5

(ii) C(C > 2.5)

(iii) find(C > 2.5)

他输出的含义。

% 你的答案

```
C > 2.5
```

```
ans = 3x6 logical 数组
     0     0     1     0     0     1
     0     0     0     0     0     1
     0     1     0     0     0     1
```

```
C(C > 2.5)
```

```
ans = 5x1
     3
     4
     3
```

3
3

```
find(C > 2.5)
```

```
ans = 5x1
     6
     7
    16
    17
    18
```

- (i) 对 C 中各个元素进行判断，符合不等式输出 1，不符合输出 0，输出为 logical 类型
- (ii) 输出 C 中符合不等式的元素
- (iii) 输出 C 中符合不等式的元素的线性索引

% 思考：[row,col] = find(C > 2.5) 得到了什么结果。搜索 find() 文档

得到了符合不等式的元素对应的行列下标，分别存储在 row 和 col 中

% 思考：ind2sub 有什么功能，如何使用

将元素在某个大小的矩阵中的线性索引转化为对应的行列下标

2、以下是某名被试在完成一按反任的 RT 数据，按要求行整理。

trial	RT_1	RT_2
1	2	3
1	7.5901	10.9424
2	4.2887	8.0937
3	8.0614	4.8777
4	2.1539	4.3062
5	2.6017	0.8277

(1) 将数据“sub_001.mat”调入 matlab 中。

```
% 你的答案
% 提示：load()
load('sub_001.mat')
```

% 思考：什么是绝对路径，什么是相对路径？

绝对路径从根目录开始，相对路径从工作目录开始

% 思考：你的答案是哪一种写法？另一种写法怎么写呢？

答案为相对路径，另一种写法为 load('C:\Users\86137\Desktop\Class\MATLAB\homework\matlab_exe1')

% 思考：windows 和 mac 相比，二者 matlab 中绝对路径的表达有何不同？

(2) 数据中的□若□ 999，□代表数据缺失。□将数据缺失的□次找出并□除。

% 你的答案

% 提示：搜索 find() 文档；删除某行某列可以使其等于 []

```
[row, col] = find(data == 999)
```

```
row = 3×1
     6
    17
    12
col = 3×1
     2
     2
     3
```

```
data(row,:) = []
```

```
data = 17×3
     1.0000     7.5901    10.9424
     2.0000     4.2887     8.0937
     3.0000     8.0614     4.8777
     4.0000     2.1539     4.3062
     5.0000     2.6017     0.8277
     7.0000     6.0881     9.3463
     8.0000     5.6527     0.7706
     9.0000     4.2657     2.7290
    10.0000     7.5406     3.1000
    11.0000     1.9758     0.7143
     ⋮
```

% 思考：数据缺失的试次并不是按顺序排列，搜索 sort() 如何使用，

% 并将试次编号按升序和降序进行排列。

```
sort(row)
```

```
sort(row, "descend")
```

% 思考：如何判断是否清理干净，搜索 ismember() 文档。

```
ismember(999,data)
```

```
ans = logical
     0
```

(3) □算每个□次前后两次按□反□□之差，并将数据作□新的一列添加在最后。

% 你的答案

```
newRT = data(:,3) - data(:,2)
```

```
newRT = 17×1
     3.3522
     3.8050
    -3.1838
     2.1523
    -1.7740
     3.2582
    -4.8821
    -1.5367
    -4.4406
    -1.2615
     ⋮
```

```
data = [data newRT]
```

```
data = 17x4
    1.0000    7.5901   10.9424    3.3522
    2.0000    4.2887    8.0937    3.8050
    3.0000    8.0614    4.8777   -3.1838
    4.0000    2.1539    4.3062    2.1523
    5.0000    2.6017    0.8277   -1.7740
    7.0000    6.0881    9.3463    3.2582
    8.0000    5.6527    0.7706   -4.8821
    9.0000    4.2657    2.7290   -1.5367
   10.0000    7.5406    3.1000   -4.4406
   11.0000    1.9758    0.7143   -1.2615
      ⋮
```

(4) 将得到的数据以“sub_001_clean.mat”的名字保存在新建文件夹“data_clean”中。

```
% 你的答案
% 提示：新建文件夹 mkdir(); 保存数据 save()
mkdir('data_clean')
```

警告：目录已存在。

```
save('sub_001_clean.mat','data')
```

```
% 思考：写一个完整的绝对路径通常很繁琐，
% 代码 save('.\data_clean\sub_001_clean.mat','data') 可以达到相同效果
% 其中 . 代表什么？
% 工作目录？
% 思考：在 command window 中输入 cd .. 其中 .. 代表什么？
% 返回上一级目录
```