

## 作业一

一. 已知矩阵

A:

3	1	2
2	1	2
1	2	3

B:

1	1	-1
2	0	1
1	0	1

要求

- (1) 屏幕输出A与B;
- (2) A的转置A';
- (3) 求A+B的值;
- (4) 求4A;
- (5) 求A×B;
- (6) 求|A|;
- (7) 求A<sup>-1</sup>。

% 定义

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

```
B = [1 1 -1; 2 0 1; 1 0 1];
```

% 输出

```
disp('Matrix A:');
```

```
disp(A);
```

```
disp('Matrix B:');
```

```
disp(B);
```

% A'

```
disp('A':');
```

```
disp(A');
```

% A + B

```
disp('A + B:');
```

```
disp(A + B);
```

% 4A

```
disp('4A:');
```

```
disp(4 * A);
```

% A × B

```
disp('A × B:');
```

```
disp(A * B);
```

% |A|

```
disp('|A|:');
```

```
disp(det(A));
```

```

% A^-1
disp('A^-1:');
disp(inv(A));
Matrix A:
    3    1    2
    2    1    2
    1    2    3
Matrix B:
    1    1   -1
    2    0    1
    1    0    1
A':
    3    2    1
    1    1    2
    2    2    3
A + B:
    4    2    1
    4    1    3
    2    2    4
4A:
   12    4    8
    8    4    8
    4    8   12
A × B:
    7    3    0
    6    2    1
    8    1    4
|A|:
   -1
A^-1:
    1.0000   -1.0000   -0.0000
    4.0000   -7.0000    2.0000
   -3.0000    5.0000   -1.0000

```

二. 对以下问题，编写M文件求解：

一球从100m高度自由下落，每次落地后反跳回原高度的一半，再落下。求它在第10次落地时，共经过多少米？第10次反弹有多高？

编写函数：

```
falling_ball.m x +
/MATLAB Drive/falling_ball.m
1 function [total_distance, height_of_bounce] = falling_ball(height, num_bounces)
2     total_distance = height; % 初始高度即第一次下落距离
3     height_of_bounce = height / 2; % 初始反弹高度
4
5     for i = 2:num_bounces % 从第二次反弹开始计算
6         % 每次下落的距离
7         total_distance = total_distance + 2 * height_of_bounce;
8         % 计算下一次反弹高度
9         height_of_bounce = height_of_bounce / 2;
10    end
11 end
12 |
```

% 调用函数求解

height = 100; % 初始高度为 100m

num\_bounces = 10; % 第 10 次落地

[total\_distance, height\_of\_bounce] = falling\_ball(height, num\_bounces);

% 输出结果

disp(['第', num2str(num\_bounces), '次落地时共经过 ', num2str(total\_distance),  
' 米']);

disp(['第', num2str(num\_bounces), '次反弹的高度为 ',  
num2str(height\_of\_bounce), ' 米']);

第 10 次落地时共经过 299.6094 米

第 10 次反弹的高度为 0.097656 米

三. 分别用plot,fplot绘制函数 $y=\cos(\tan(\pi x))$ 的图形。

Plot:

x = linspace(-2, 2, 1000); % 在 -2 到 2 之间生成 1000 个点

y = cos(tan(pi\*x)); % 计算函数值

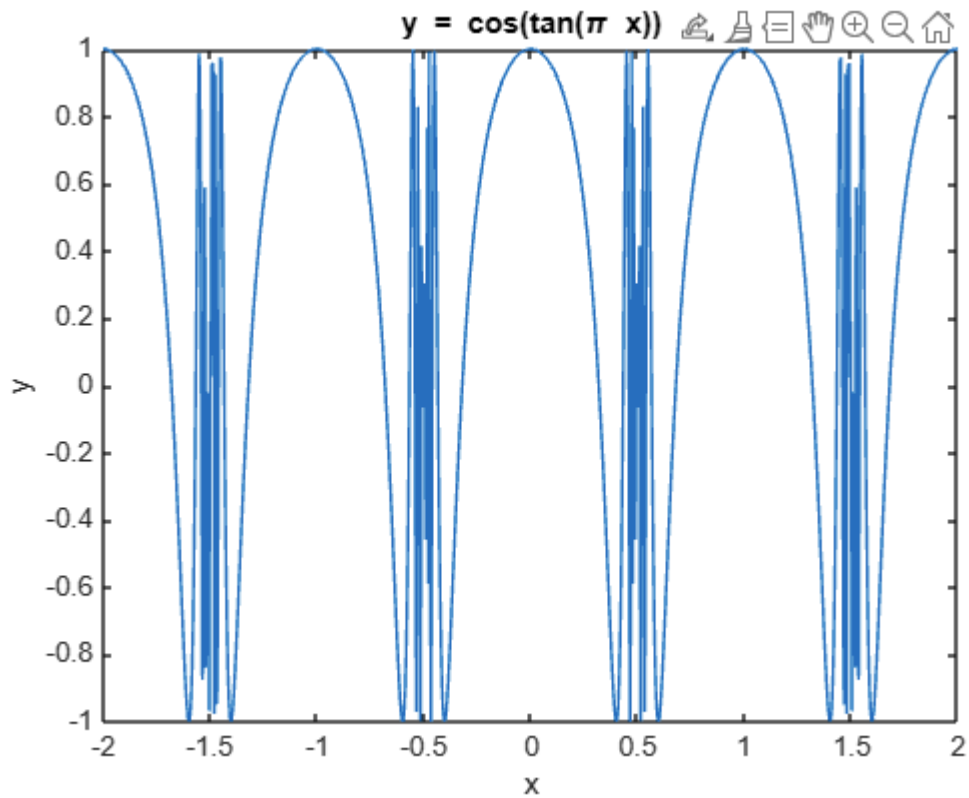
figure; % 创建新的图形窗口

plot(x, y); % 绘制图形

title('y = cos(tan(\pi x))'); % 添加标题

xlabel('x'); % 添加 x 轴标签

ylabel('y'); % 添加 y 轴标签



Fplot:

```
f = @(x) cos(tan(pi*x)); % 定义函数
```

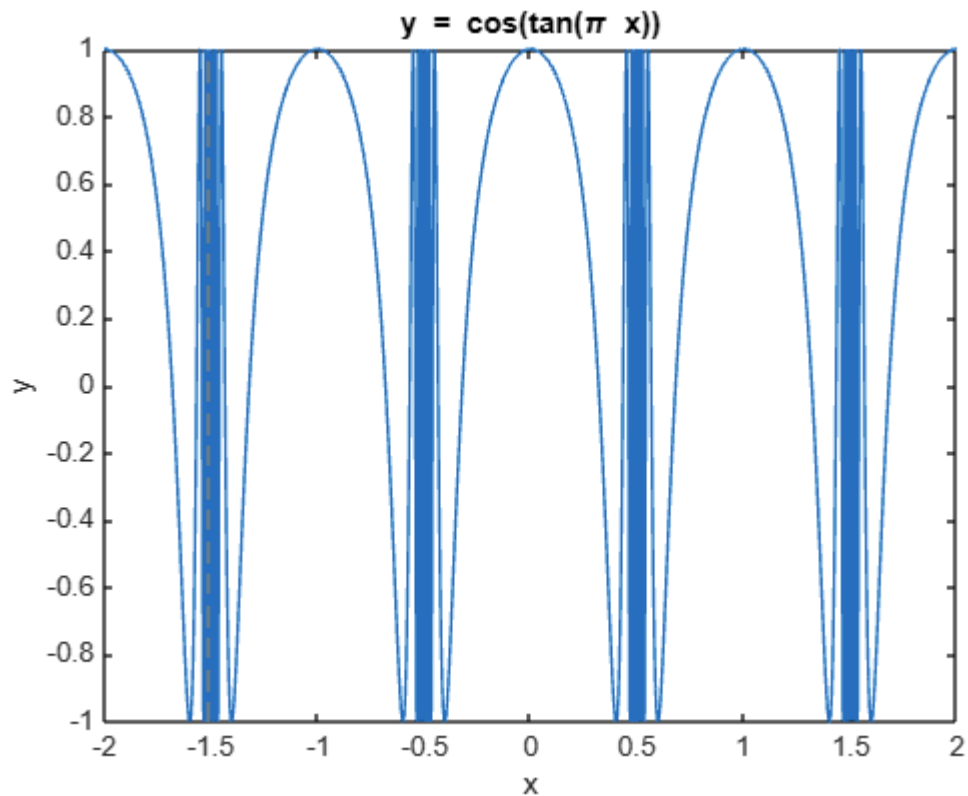
```
figure; % 创建新的图形窗口
```

```
fplot(f, [-2, 2]); % 绘制函数图形, 指定 x 范围为 -2 到 2
```

```
title('y = cos(tan(\pi x))'); % 添加标题
```

```
xlabel('x'); % 添加 x 轴标签
```

```
ylabel('y'); % 添加 y 轴标签
```



四. 在同一坐标下做出  $y_1=x^2, y_2=x^3, y_3=x^4, y_4=x^5$  这四条曲线的图形，要求在图形上加标注。

使用 plot:

```
x = -2:0.1:2;
```

```
y1 = x.^2;
```

```
y2 = x.^3;
```

```
y3 = x.^4;
```

```
y4 = x.^5;
```

```
figure;
```

```
hold on; % 将绘图保持在同一坐标下
```

```
plot(x, y1, 'r', 'LineWidth', 2);
```

```
plot(x, y2, 'g', 'LineWidth', 2);
```

```
plot(x, y3, 'b', 'LineWidth', 2);
```

```
plot(x, y4, 'm', 'LineWidth', 2);
```

```
legend('y=x^2', 'y=x^3', 'y=x^4', 'y=x^5');
```

```
title('不同指数下的函数曲线');
```

```
xlabel('x');
```

```
ylabel('y');  
grid on; % 添加网格
```

