

## MATLAB 第 2 次作业参考答案

几点反馈：

- 作业的参考答案来自于同学们的作业，只是稍作修正，为保护隐私，隐去姓名。
- 附加题是额外的加分，鼓励大家尝试。同时也说明就算是 100 分，也不一定是全对，请大家校对答案，同时也可以对比一下代码的书写，相信会有所收获。
- 若题目要求输出运行结果，请大家截图或者复制到 WORD 中。许多同学就算题目没有要求也给出了运行结果截图，值得鼓励。
- 合理运用缩进（其实 matlab 里在 for、if 等语句后换行就会有缩进），适当增加注释，便于代码的阅读与逻辑的梳理。
- 注意作业格式，细节请看之前的格式文档，也可以参考一下参考答案的格式，之后的作业在格式方面会适当扣分。

第一题

hw2\_1.m

```
clear;%清除变量
```

```
a=2;b=1;%给定 a,b 的值
```

```
x=@(theta)(a.*cos(theta)+b).*cos(theta);%x 关于 theta 的匿名函数
```

```
y=@(theta)(a.*cos(theta)+b).*sin(theta);%y 关于 theta 的匿名函数
```

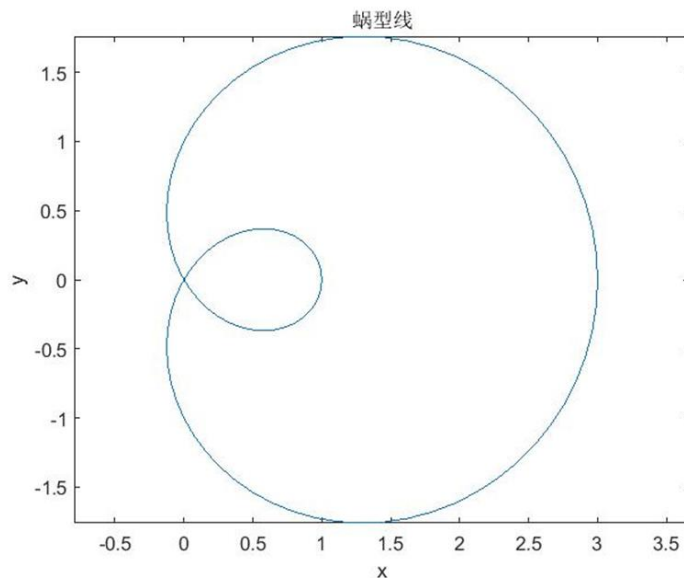
```
thetar=0:2*pi/100:2*pi;%给定要绘制的 theta 变化范围
```

```
plot(x(thetar),y(thetar));%在直角坐标系绘制图像
```

```
axis equal;%使 x-y 坐标分度值等长
```

```
title('蜗型线');%添加标题
```

```
xlabel('x');ylabel('y');%添加坐标轴标签
```



第二题

hw2.m

```
a=input('请输入一个自然数\n');
```

```
if a == fix(a) && a>0 %判断是否为自然数
```

```
    n = 0;
```

```

while a ~= 1
    if mod(a,2) == 0 %判断奇偶
        a = a/2;
    else
        a = 3*a+1;
    end
    n = n+1;
end
X = ['共计操作',num2str(n),'次'];
disp(X);
else
    disp('错误的数据类型');
end

```

实现效果如下图所示

```

>> hw2
请输入一个自然数
756433
共计操作193次
>> hw2
请输入一个自然数
2.5
错误的数据类型
>> hw2
请输入一个自然数
-24
错误的数据类型

```

### 第三题

示例 1:

hw3.m

fid = fopen('table.txt','w');%创建并打开 txt 文件

for i = 1:9

for j = 1:i

X = [num2str(j),'\*',num2str(i),'=',num2str(j\*i)];%创建要输出的算式

fprintf(fid,'%s\t',X) ;%输出并空格

end

fprintf(fid,'\r\n') ;%换行

end

fclose(fid);

```
table - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
1*1=1
1*2=2   2*2=4
1*3=3   2*3=6   3*3=9
1*4=4   2*4=8   3*4=12   4*4=16
1*5=5   2*5=10  3*5=15   4*5=20   5*5=25
1*6=6   2*6=12  3*6=18   4*6=24   5*6=30   6*6=36
1*7=7   2*7=14  3*7=21   4*7=28   5*7=35   6*7=42   7*7=49
1*8=8   2*8=16  3*8=24   4*8=32   5*8=40   6*8=48   7*8=56   8*8=64
1*9=9   2*9=18  3*9=27   4*9=36   5*9=45   6*9=54   7*9=63   8*9=72   9*9=81
```

示例二:

cheng\_fa\_biao.m

```
function cheng_fa_biao
```

```
filetext = fopen('table.txt','w');    %新建或打开文件 table.txt
```

```
for i = 1:1:9
```

```
    for j = 1:1:i
```

```
        fprintf(filetext,'%d*%d=%d ',j,i,i*j);    %i 控制行数, j 控制列数
```

```
    end
```

```
    fprintf(filetext,'\r\n');    %'\r\n'用以 txt 文件内换行
```

```
end
```

```
fclose(filetext);
```

第四题

辗转相除法

gcd\_DA.m

```
function g = gcd_DA(a,b)
```

```
%辗转相除法求 a,b 最大公约数
```

```
while(b>0)%判断余数是否非零
```

```
    %将 b 赋值给 a, 余数赋值给 b
```

```
    m=mod(a,b);
```

```
    a=b;
```

```
    b=m;
```

```
end
```

```
g=a;%输出最大公约数
```

```
end
```

● 这里的循环其实就已经覆盖了对两个数比较大小并排序的这个功能。

更相减损法

gcd\_GJ.m

```
function g = gcd_GJ(a,b)
```

```
%更相减损法求 a,b 最大公约数
```

```
g=1;%赋初值
```

```
%判断是否同时为偶数
```

```
while(mod(a,2)==0&&mod(b,2)==0)
```

```
    a=a/2;b=b/2;
```

```

        g=g*2;
    end
    %更相减损
    while(a~=b)
        if(a>b)
            a=a-b;
        else
            b=b-a;
        end
    end
    end
    g=a*g;%输出最大公约数
end

```

效率对比

```

test4.m
clear;%清除变量
a=1998;
b=324;%设置初值
tic;%辗转相除法
gcd_DA(a,b);
t_DA=toc
tic;%更相减损法
gcd_GJ(a,b);
t_GJ=toc
tic;%matlab 自带函数
gcd(a,b);
t_MAT=toc

```

t_DA	9.6000e-06
t_GJ	5.8000e-06
t_MAT	7.2900e-05

经过多次运算，发现更相减损法与辗转相除法运算时长相似，Matlab 自带函数运算时间最长。

- 这里的更相减损法其实是改进后的更相减损法，蓝色部分先进行了公因数  $2^n$  的提取，所以效率提高了。如果是原始的两种算法的比较，更相减损法在两个数的差比较大的情况下，效率会比辗转相除法低。之所以选用这个同学的代码，主要是因为其代码简洁，值得学习。

第五题

结果如下图所示

```
>> hw5
原魔方矩阵为
```

```
A =
```

35	1	6	26	19	24
3	32	7	21	23	25
31	9	2	22	27	20
8	28	33	17	10	15
30	5	34	12	14	16
4	36	29	13	18	11

```
经过质数判断后的矩阵为
```

```
B =
```

0	0	0	0	19	0
3	0	7	0	23	0
31	0	2	0	0	0
0	0	0	17	0	0
0	5	0	0	0	0
0	0	29	13	0	11

```
hw51.m (质数判断函数)
```

```
function p = hw51(n)
```

```
p=1; %先假设为质数
```

```
if n ~= 2 && n ~= 3 %去掉 n=2 n=3 的情况
```

```
    if mod(n,2) == 0 || n == 1 %如果 n 为 1 或者是偶数 不是质数
```

```
        p = 0;
```

```
    else
```

```
        for i = 3:2:floor(n^0.5) %从 3 到根号 n 之间的奇数一个一个进行判断
```

```
            if mod(n,i) == 0
```

```
                p = 0;
```

```
                break
```

```
            end
```

```
        end
```

```
    end
```

```
end
```

```
end
```

```
hw5.m
```

```
A = magic(6);
```

```
B = ones(6,6);
```

```
for i = 1:6
```

```
    for j = 1:6
```

```
        if hw51(A(i,j)) == 0
```

```
            B(i,j) = 0;
```

```
        else
```

```

        B(i,j) = A(i,j);
    end
end
end
disp('原魔方矩阵为');
A
disp('经过质数判断后的矩阵为');
B

```

- 注意 1 不是质数，很多同学都漏掉了 1。大家对其他的质数的判断都写得不错，部分同学还对输入的变量类型进行了判断，十分细致。

## 第六题

示例一：

结果如下图所示

```

>> hw62
插入排序法排序所用时间为0.15758s
>> hw63
冒泡排序法排序所用时间为0.62789s
>> tic; sort(A); time3 = toc; X = ['sort函数排序所用时间为',num2str(time3),'s']; disp(X);
sort函数排序所用时间为0.017207s

```

冒泡排序法效率最低，插入排序法效率较高，但是依然无法和 Matlab 自带的 sort 函数相比。

hw61.m（生成随机数数组并保存至 txt 文件）

```

fid = fopen('test.txt','w');
A = floor(rand(10000,1)*10000);
for i = 1:length(A)
    fprintf(fid,'%d\r\n',A(i));
end
fclose(fid);

```

hw62.m（插入排序法实现排序）

```

B = textread('test.txt');
tic;
for i = 2:length(B)
    temp = B(i);
    j = i-1;
    while j > 0
        if B(j) > temp
            B(j+1) = B(j);
            j = j-1;
        else
            break
        end %从后往前比较大小 找到新加入元素应该在的位置
    end
end

```

```

        end
        B(j+1) = temp;
    end
    time1 = toc;
    X = ['插入排序法排序所用时间为',num2str(time1),'s'];
    disp(X);

```

hw63.m（冒泡排序法实现排序）

```

C = textread('test.txt');
tic;
for i = length(C)-1:-1:1
    for j = 1:i
        if C(j) > C(j+1)
            temp = C(j+1);
            C(j+1) = C(j);
            C(j) = temp;
        end
    end
end
time2 = toc;
X = ['冒泡排序法排序所用时间为',num2str(time2),'s'];
disp(X);

```

示例二：

效率对比

命令行图片即用时如下，可见 matlab 自带函数效率最高，冒泡排序效率最低，插值排序居中。

```

命令窗口

>> tic;fid=fopen('hw2_6\test.txt','r');
A=fscanf(fid,'%d',10000);
fclose(fid);sort(A);ts=toc

ts =

    0.0308

>> tic;test6_maopao;tm=toc

tm =

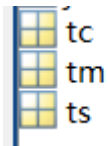
    0.4108

>> tic;test6_chazhi;tc=toc

tc =

    0.1224

```



tc  
tm  
ts  
0.1224  
0.4108  
0.0308

```
test6_random.m
clear;%清除变量
%生成随机数列
A=randi([0 10000],10000,1);
%打开文件
fid=fopen('hw2_6/test.txt','w');
%输入数据
fprintf(fid,'%d\n',A);
%关闭文件
fclose(fid);
```

```
test6_chazhi.m
clear;%清除变量
%导入数据并存放在矩阵 A 中
fid=fopen('hw2_6/test.txt','r');
A=fscanf(fid,'%d',10000);
fclose(fid);
for i=2:length(A)
    %把第 i 个数取出暂存
    temp=A(i);
    %从后向前逐个比较
    for j=i-1:-1:1
        %若比第 i 个数大，向后移动，第 i 个数放置其前
        if A(j)>temp
            A(j+1)=A(j);
            A(j)=temp;
        else
            %若比第 i 个数小，将第 i 个数放置其后
            A(j+1)=temp;
            break
        end
    end
end
end
```

```
test6_maopao.m
clear;%清除变量
%导入数据并存放在矩阵 A 中
fid=fopen('hw2_6/test.txt','r');
A=fscanf(fid,'%d',10000);
fclose(fid);
```



```

for i=1:length(A)-1
    %没进行一次下方循环，便多一个排好位置的数
    %对于 n 个数，排好 n-1 个即可，故循环 n-1 次
    for j=1:length(A)-i-1
        %比较相邻的数，使大者在后
        %这样最后一个数必定最大
        %已经有 i-1 个拍好，倒数第 i 个，循环 n-i 次
        if A(j)>A(j+1)
            temp=A(j);
            A(j)=A(j+1);
            A(j+1)=temp;
        end
    end
end
end

```

第七题

示例一：

mat\_add.m

```
function total = mat_add(varargin)
```

```
% 实现任意多矩阵的相加
```

```
total=0;% 初始化结果矩阵
```

```
%累加求和
```

```
for i=1:length(varargin)
```

```
total=total+varargin{i};
```

```
end
```

```
end
```

示例二：

mat\_add.m

```
function sum = mat_add(varargin)
```

```
sum = 0;
```

```
try
```

```
for i = 1:1:nargin
```

```
sum = sum+varargin{i};
```

```
end %将各个输入元素相加
```

```
catch
```

```
sum = NaN;
```

```
X = '输入变量的维度或格式不正确';
```

```
disp(X); %如果不能相加则输出错误
```

```
end
```

```
end
```

结果如下图所示

```

>> A1 = rand(1,4);
>> A2 = rand(1,4);
>> A3 = rand(1,4);
>> A4 = rand(1,4);
>> A5 = rand(1,5);
>> A123 = mat_add(A1,A2,A3)

A123 =

    2.0448    1.1751    1.4697    0.9607

>> A1234 = mat_add(A1,A2,A3,A4)

A1234 =

    2.2940    1.5504    2.0594    1.9559

>> A15 = mat_add(A1,A5)
输入变量的维度或格式不正确

A15 =

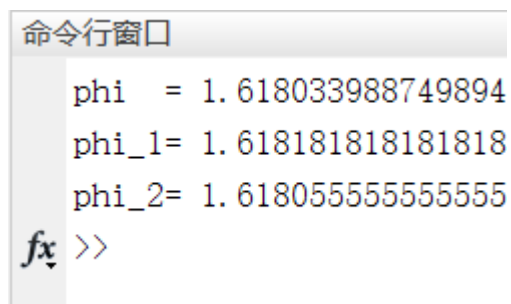
    NaN

```

- 这个同学这里用到了 `try...catch...`，值得学习

## 第八题

运算结果如图



```

命令窗口

phi    = 1.618033988749894
phi_1= 1.618181818181818
phi_2= 1.618055555555555
fx >>

```

test8.m

```
clear;clc;%清除变量
```

```
phi=(1+sqrt(5))/2;%理论值
```

```
n=10;
```

```
%Fibonacci 法
```

```
phi_1=fibonacci(n+1)/fibonacci(n);
```

```
%分式迭代法
```

```
phi_2=1;
```

```
%小数部分迭代
```

```
for i=1:n
```

```
    phi_2=1/(1+phi_2);
```

```
end
```

```
%最终结果
```

```
phi_2=1+phi_2;
```

```
%输出
```

```
fprintf('phi%s= %.15f\n',' ',phi,'_1',phi_1,'_2',phi_2);
```

● 请注意有效数字的定义是从第一个非零数字算起的。

#### 第九题

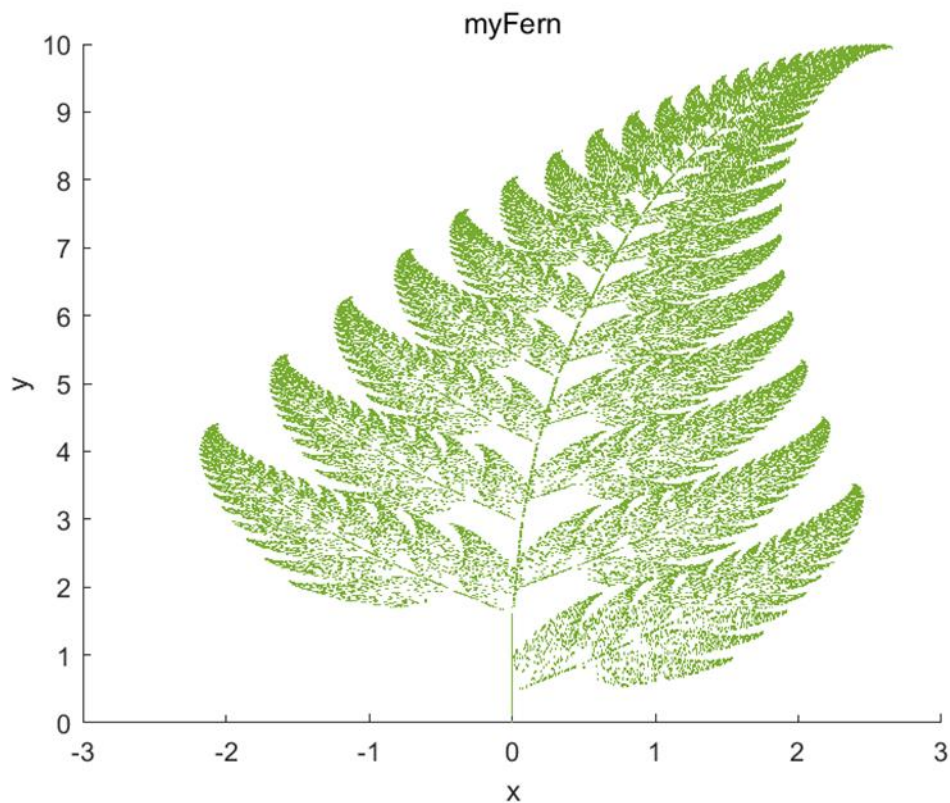
编写好函数后命令行中输入如下代码

```
>> c=[0.4660 0.6740 0.1880];[x y] = myFern(1e5);scatter(x,y,1,c,'filled');
```

```
>> hold on
```

```
>> title('myFern');xlabel('x');ylabel('y');
```

即可得到结果：



myFern.m

```
function [xt,yt] = myFern(n)
```

```
if n == 1
```

```
    xt = 0;
```

```
    yt = 0;
```

```
elseif n == fix(n) && n>1
```

```
    xt = zeros(n,1);
```

```
    yt = zeros(n,1);
```

```
    for i = 2:n
```

```
        tem = rand(1);
```

```
        if tem <= 0.01
```

```
            xt(i) = 0;
```

```
            yt(i) = 0.16*yt(i-1);
```

```
        elseif tem <= 0.08
```

```
            xt(i) = 0.2*xt(i-1)-0.26*yt(i-1);
```

```
            yt(i) = 0.23*xt(i-1)+0.22*yt(i-1)+1.6;
```

```
        elseif tem <= 0.15
```

```
            xt(i) = -0.15*xt(i-1)+0.28*yt(i-1);
```

```
            yt(i) = 0.26*xt(i-1)+0.24*yt(i-1)+0.44;
```

```
        else
```

```
            xt(i) = 0.85*xt(i-1)+0.04*yt(i-1);
```

```

            yt(i) = -0.04*xt(i-1)+0.85*yt(i-1)+1.6;
        end
    end
else
    disp('错误的函数输入');
end
end
end

```

附加题

该有向图中从点 1 走到点 7 所要花费的最小代价为 5

hw2\_10.m

```

A=[7 5 4;7 6 1; 7 3 7;5 2 2;6 2 3; 6 3 4; 6 4 2; 3 1 2; 2 1 1; 4 1 3];
cost=mincost(1,7,A);
disp(cost);

```

mincost.m

```

function cost=mincost(x,y,A)
if x==y
    cost=0;
else
    mid=[];
    mid=find(A(:,2)==x);
    tmp=[];
    for i=1:length(mid)
        tmp(i)=A(mid(i),3)+mincost(A(mid(i),1),y,A);
    end
    cost=min(tmp);
end
end

```