模块二：函数的迭代

教学要求：要求学生掌握迭代、混沌的判断方法，以及利用迭代思想解决实际问题。

2.1 (1)求出分式线性函数 $f(x)=\dfrac{2x+1}{x-m}$ 的不动点；

(2)编程判断 $f(x)$ 产生的迭代序列是否收敛？若收敛，收敛值与初值是否有关？与不动点又有何关系？

(1) 代码：
```
syms x
m=1011;
y=(2*x+1)/(x-m)-x;
solve(y,x)
```
结果：
```
ans=
1013/2-1026173^(1/2)/2
1026173^(1/2)/2+1013/2
```

(2) iteration.m文件：
```
function p=interation(f,x0,n)
p=x0;
for i=2:n
    p(i)=f(p(i-1));
end
end
```

命令：
```
iteration(@(x)(2*x+1)/(x-1011), 初始值 ,10)
```
取初始值为不动点1、不动点2和其他点

结论：该函数收敛.

初值不是第二个不动点，收敛于第一个不动点；

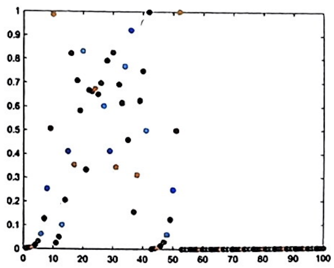
初值是第二个不动点，收敛于第二个不动点.

2.2 初值分别取 $\frac{1}{4},\frac{1}{10},\frac{1}{1000},\frac{1}{m}$，通过离散点图判断下面函数产生的迭代序列会出现哪些结果？

$$f(x)=\begin{cases}2x, & 0\le x\le \frac{1}{2}\\ 2(1-x), & \frac{1}{2}<x\le 1\end{cases}$$

代码（选一个初值写出）为：

```
f= @(x)1-2*abs(x-1/2);
x0 = 1/1011;
for i=1:100
    plot(i,f(x0),'.','MarkerSize',20);
    x0=f(x0);
    hold on
end
hold off
```



结论：无论初值为何值，该迭代序列总是趋向于0.

2.3 对于 Martin 迭代，任选下表中参数 a,b,c 的四组值，并将迭代次数分别选取为 5000，10000，15000，20000，画出相应的图形（每个图形均需标注对应的参数）。

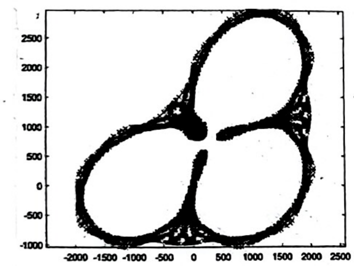
a	b	c
m	m	m
−m	−m	m
−m	m/1000	−m
m/1000	m/1000	0.5
m/1000	m	−m
m/100	m/10	−10
−m/10	17	4

Martin 函数代码：

```
function Martin(a,b,c,N)
f=@(x,y)(y-sign(x)*sqrt(abs(b*x-c)));
g=@(x)(a-x);
m=[0;0];
for n=1:N
    m(:,n+1)=[f(m(1,n),m(2,n)),g(m(1,n))];
end
plot(m(1,:),m(2,:),'kx')
axis equal
```

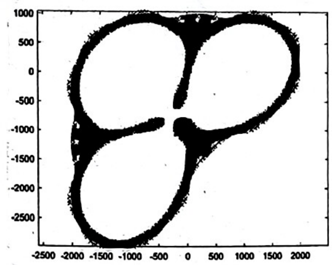
调用 Martin 函数代码：

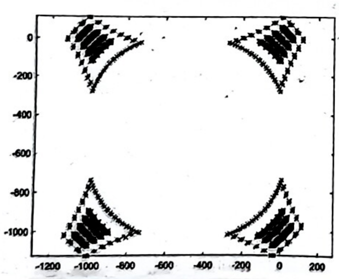
```
m=1011;
Martin(m,m,m,5000)
Martin(-m,-m,m,10000)
Martin(-m,m/1000,-m,15000)
Martin(m/1000,m/1000,0.5,20000)
```



① (m,m,m,5000)

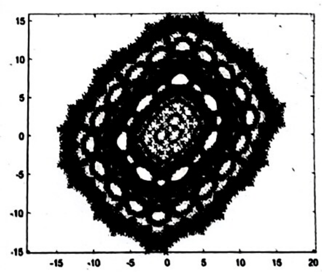
② (−m,−m,m,10000)





③ (−m,m/1000,−m,15000)

④ (m/1000,m/1000,0.5,20000)



2.4 试找出分式函数 $f(x)=\dfrac{ax+b}{cx^2+dx+e}$（其中 a,b,c,d,e 是整数），使它产生的迭代序列（迭代的初始值也是整数）收敛到 $\sqrt[3]{m}$（对于 $\sqrt[3]{m}$ 为整数的学号，请改为求 $\sqrt[3]{10m}$），具体要求如下：

（1）选取 a,b,c,d,e 确定分式函数 $f(x)$ 使得：

① $\sqrt[3]{m}$ 或者 $\sqrt[3]{10m}$ 是 $f(x)$ 的不动点；

② $\left|f'\left(\sqrt[3]{m}\right)\right|<1$ 或者 $\left|f'\left(\sqrt[3]{10m}\right)\right|<1$ 以满足收敛的充分性条件；

（2）编程验证迭代序列是否收敛到 $\sqrt[3]{m}$ 或 $\sqrt[3]{10m}$，写出代码和运行结果；

（3）如果迭代收敛，那么迭代的初值与收敛的速度有什么关系. 写出你做此题的体会.（可以教材 66 页练习 4 的一些分析）

(1) m=1011 计算得出一种结果如下：

a= 100

b= 1011

c= 1

d= 0

e= 100

(2) 代码：
```
f=@(x)(100*x+1011)/(x^2+100);
x0=15;
for i=1:15
    x0=f(x0);
    fprintf('%g. %g\n',i,x0);
end
```
结果：

1. 7.72615
2. 11.169
3. 9.468
4. 10.3236
5. 9.98153
6. 10.1098
7. 9.99949

8.	10.0553
9.	10.0271
10.	10.0413
11.	10.0341
12.	10.0378
13.	10.0359
14.	10.0368
15.	10.0364
16.	10.0366
17.	10.0365
18.	10.0366
19.	10.0365
20.	10.0365
21.	10.0365
22.	10.0365
23.	10.0365
24.	10.0365
25.	10.0365

(3) 结论：迭代的初值离不动点差距越大，所收敛的步长越大，但同时到达不动点附近所需迭代次数也随之增多.

体会：若函数有着有限个不动点，则该函数的迭代序列必定收敛.

2.5 对于函数 $f(x)=\sin x$，记 $f_n(x)$ 为 $f(x)$ 在 $x=0$ 处的 n 阶泰勒展开式.

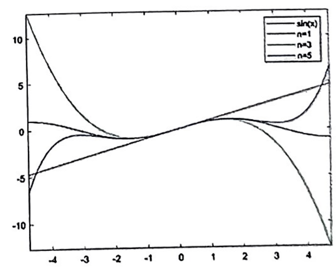
（1）在同一个坐标系中，用不同颜色画出区间 $\left(-\frac{3\pi}{2},\frac{3\pi}{2}\right)$ 内 $f(x)$ 及 $f_n(x)(n=1,3,5)$ 图形；

（2）在同一个坐标系中，用不同颜色画出区间 $\left(-\frac{3\pi}{2},\frac{3\pi}{2}\right)$ 内 $f(x)$ 及 $f_n(x)(n=7,9,11)$ 图形.

（3）将（1）和（2）中的各函数图形加以比较，写出你的体会.

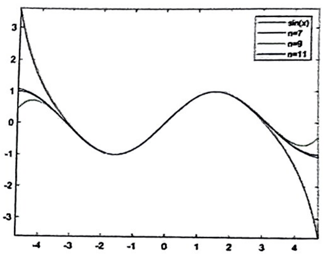
(1) 代码：

```
syms x
y= sin(x);
y1= taylor (sin(x), x, 'Order', 2);
y2= taylor (sin(x), x, 'Order', 4);
y3 = taylor (sin(x), x, 'Order', 6);
fplot (y, 'k');
hold on;
fplot (y1, 'r');
fplot (y2, 'g');
fplot (y3, 'b');
xlim([-3/2*pi, 3/2*pi]);
grid on;
legend('sin(x)', 'n=1', 'n=3', 'n=5');
```



(2) 代码：

```
syms x
y=sin(x);
y1=taylor(sin(x),x,'Order',8);
y2=taylor(sin(x),x,'Order',10);
y3=taylor(sin(x),x,'Order',12);
fplot(y,'k');
hold on;
fplot(y1,'r');
fplot(y2,'g');
fplot(y3,'b');
xlim([-3/2*pi,3/2*pi]);
grid on;
legend('sin(x)','n=7','n=9','n=11');
```



(3) 体会：泰勒展开的阶数越高，其图形越接近原函数.