Numerical Computing HW2

Yunfan Gao

February 15, 2021

Problem 1

Calculate step by step: $f(0) = -1$ negative
$f(1) = 1 - cos(1)$ positive
$f(0.5) = \sqrt{0.5} - \cos(0.5)$ negative
$f(0.75) = \sqrt{0.75} - \cos(0.75)$ positive
$f(0.625) = \sqrt{0.625} - \cos(0.625)$ negative
$f(0.6875) = \sqrt{0.6875} - \cos(0.6875)$ positive
$f(0.65625) = \sqrt{0.65625} - \cos(0.65625)$ positive
$f(0.640625)$ negative
$f(0.6484375)$ positive
...
0.642578125
0.6416015625
0.64208984375
0.641845703125
0.6417236328125
The answer is around 0.6417
Verify using bisect code:

```
%Program 1.1 Bisection Method
%Computes approximate solution of f(x)=0
%Input: function handle f; a,b such that f(a)*f(b)<0,
% and tolerance tol
%Output: Approximate solution xc

function xc=bisect(f,a,b,tol)
if sign(f(a))*sign(f(b)) >= 0
    error('f(a)f(b)<0 not satisfied!') %ceases execution
end
fa=f(a);
fb=f(b);
while (b-a)/2>tol
    c=(a+b)/2;
    fc=f(c);
```

```
    if fc == 0 %c is a solution, done
        break
    end
    if sign(fc)*sign(fa)<0 %a and c make the new interval
        b=c;fb=fc;
    else %c and b make the new interval
    a=c;fa=fc;
    end
end
xc=(a+b)/2; %new midpoint is best estimate
```

```
>> f=@(x) sqrt(x)-cos(x)

f =

  function_handle with value:

    @(x)sqrt(x)-cos(x)

>> xc=bisect(f,0,1,0.00005)

xc =

    0.6417
```

Thus using the bisection method to find xc with four digits: 0.6417

Problem 2

The zeros can only be -2 -1 0 1 2
$f(-1.5) = p * p * n * n * n$ negative
$f(2.5) = p * p * p * p * p$ positive
$f(0.5) = p * p * p * n * n$ positive
$f(-0.5) = p * p * n * n * n$ negative
Thus x = 0 is the zero f will converge
$f(-0.5)$ negative
$f(2.4)$ positive
$f(0.95) = p * p * p * n * n$ positive
Thus it will converge to the zero at x = 0 when k goes to infinity.

Problem 3

$f(x) = x^4 + 2x^2 - x - 3$
$g_1(x) = \left(3 + x - 2x^2\right)^{1/4}$
when f(x) = 0, $0 = x^4 + 2x^2 - x - 3$
$x + 3 - 2x^2 = x^4 => x = \left(3 + x - 2x^2\right)^{1/4}$

$g_2(x) = \left(\frac{x+3-x^4}{2}\right)^{1/2}$
$0 = x^4 + 2x^2 - x - 3$
$x + 3 - x^4 = 2x^2$
$\frac{x+3-x^4}{2} = x^2 => x = \left(\frac{x+3-x^4}{2}\right)^{1/2}$
Thus both g1 and g2 have fixed point when f = 0

Problem 4

$g(x) = x/2 + 1/x$ has fixed point $x = x/2 + 1/x$
$x^2 = 2 => x = \pm\sqrt{2}$
which are the solution of $f(x) = x^2 - 2 = 0$
To prove that g(x) has a unique fixed point in [1, 2]: First need to check g(x) is in the range of [1, 2].
$g'(x) = \frac{1}{2} - \frac{1}{x^2}$
$g'(x) < 0$ for $x < \sqrt{2}$ and $g'(x) > 0$ for $x > \sqrt{2}$
Thus
$\min g(x) = g(\sqrt{2}) = \sqrt{2}$
$\max g(x) = g(1) = g(2) = 1.5$
$Thus$
$\sqrt{2} \leq g(x) \leq \frac{3}{2}$ for x in [1, 2]
$\max|g'(x)| = |g'(x)| = \frac{1}{2} < 1$
Thus there exists a unique fixed point for g(x) in [1, 2]

Problem 5

The bisect code is same as shown in first question: With the following entry commands and outputs

```
>> f=@(x) x-cos(x);
>> xc=bisect(f,0,pi/2,0.00000001)


xc =


    0.7391
```

```
>> f=@(x) exp(x)+x-7;
>> xc=bisect(f,1,2,0.00000001)

xc =

    1.6728
```

Problem 6

```
%Program 1.2 Fixed-Point Iteration
%Computes approximate solution of g(x)=x
%Input: function handle g, starting guess x0,
% number of iteration steps k
%Output: Approximate solution xc
function xc=fpi(g, x0, k)
x(1)=x0;
for i=1:k
x(i+1)=g(x(i));
end
xc=x(k+1);
```

```
>> g=@(x) cos(x)

g =

  function_handle with value:

    @(x)cos(x)

>> xc=fpi(g,pi/4,100)

xc =

    0.7391
```

```
>> g=@(x) log(7-x)

g =

  function_handle with value:

    @(x)log(7-x)

>> xc=fpi(g,1.5,100)

xc =

    1.6728
```