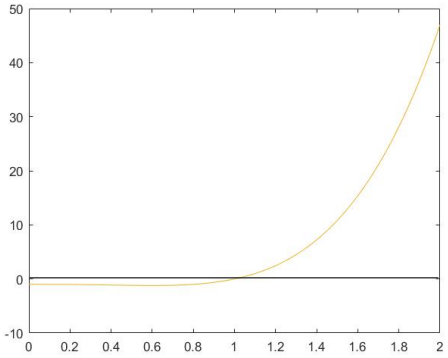
Homework 2
Jingmin Sun
661849071

1. We can observe that around $x = 1$, the function touches x axis, but not cross it, so we cannot use bisection method to find this root directly. Instead of that, we can find the local minimum of the function, and the local minimum will be the one that touches x axis. To find the minimum, we can differentiate the function since it's a nice smooth function. If the derivative equals 0 at some point, then it will be the local minimum $\bar{x}$, and we can find the value of $f(\bar{x})$, if it is positive, then the original graph does not have solution around $x = 1$, and if it is zero, then we get the answer, and if it is negative, we can use bisection method directly to the original function.

   So, our goal now is find the root for the derivative of the original function by bisection method, and the derivative of the original function is $5x^4 - 4x^3 - 1$, and we can observe that it cross the x-axis around 1, so we can apply bisection method to find the root for it around 1.



2. (a) To compute the square root of a positive constant α, it's same to solve the equation $x = \sqrt{\alpha}$, since we can only involve $+, -, *, /$, so we can choose to find the root of

$$f(x) = x^2 - \alpha$$

   (b) When $\alpha = 3$, we need to find the root for $f(x) = x^2 - 3$, suppose $[a_0, b_0] = [1, 2]$, since

$$\begin{aligned} f(a_0) = f(1) = 1 - 3 = -2 \\ f(b_0) = f(2) = 4 - 3 = 1 \\ \therefore f(a_0)f(b_0) = -2 \cdot 1 = -2 < 0 \end{aligned}$$

the choice of $[a_0, b_0]$ is valid. Let $c_0 = \dfrac{a_0 + b_0}{2} = \dfrac{3}{2}$, and

$$\begin{aligned} f(c_0) = f\left(\frac{3}{2}\right) &= \frac{9}{4} - 3 = -\frac{3}{4} \\ \because f(a_0)f(c_0) &= -2 \cdot -\frac{3}{4} = \frac{3}{2} > 0 \\ f(b_0)f(c_0) &= 1 \cdot -\frac{3}{4} = -\frac{3}{4} < 0 \\ \therefore [a_1, b_1] &= [c_0, b_0] = \left[\frac{3}{2}, 2\right] \\ c_1 &= \frac{a_1 + b_1}{2} = \frac{\frac{3}{2} + 2}{2} = \frac{7}{4} \end{aligned}$$

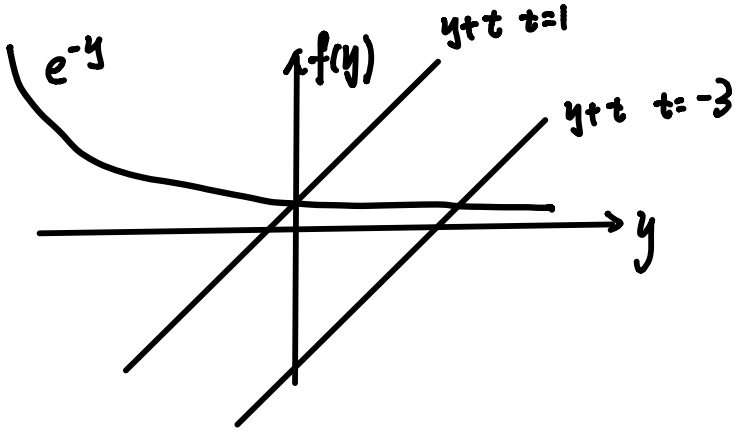
(c) The update formula for Newton's method is

$$\begin{aligned} x_{j+1} &= x_j - \frac{f(x_j)}{f'(x_j)} \\ &= x_j - \frac{x_j^2 - 3}{2x_j} \qquad \forall j = 1, 2, 3 \cdots \end{aligned}$$

Choose $x_0 = \frac{3}{2}$, since $f(x_0) = -\frac{3}{4}$, which is near zero, so it's a good choice.

$$\begin{aligned} x_1 &= x_0 - \frac{x_0^2 - 3}{2x_0} \\ &= x_0 - \frac{x_0}{2} + \frac{3}{2x_0} \\ &= \frac{x_0}{2} + \frac{3}{2x_0} \\ &= \frac{\frac{3}{2}}{2} + \frac{3}{2\frac{3}{2}} \\ &= \frac{7}{4} \\ x_2 &= x_1 - \frac{x_1^2 - 3}{2x_1} \\ &= x_1 - \frac{x_1}{2} + \frac{3}{2x_1} \\ &= \frac{x_1}{2} + \frac{3}{2x_1} \\ &= \frac{\frac{7}{4}}{2} + \frac{3}{2\frac{7}{4}} \\ &= \frac{97}{56} \end{aligned}$$

3. (a) Sketch:

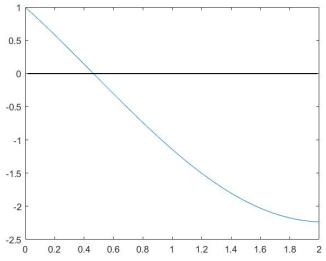


Since the solution for $y + t = e^{-y}$ has exactly the same solution as $y + t - e^{-y} = 0$, let $f(y) = y + t - e^{-y}$, and $f'(y) = 1 + e^{-y} > 0$ follows, so $f(y)$ is a monotony increasing function, which means it has only one intersection to the $f(y) = 0$, so, no matter what t is, there is only one solution.

(b) The updated formula for secant method is

$$\begin{aligned} y_{j+1} &= y_j - \frac{f(y_j)(y_j - y_{j-1})}{f(y_j) - f(y_{j-1})} \\ &= y_j - \frac{(y_j + t - e^{-y_j})(y_j - y_{j-1})}{(y_j + t - e^{-y_j}) - (y_{j-1} + t - e^{-y_{j-1}})} \\ &= y_j - \frac{(y_j + t - e^{-y_j})(y_j - y_{j-1})}{(y_j - y_{j-1}) + (e^{-y_{j-1}} - e^{-y_j})} \end{aligned}$$

4. To find the root for $\cos(x) = 2\sin(x)$, it's same to compute the root for $f(x) = \cos(x) - 2\sin(x)$. Since we aim to find the smallest positive root, we can draw the graph first,



and we can observe that the smallest positive root lies between 0 and 1, which means we can set the initial interval to $[0, 1]$, and for stopping criterion, we need to make the computational root to lies in the 10^{-6} neighborhood of $\bar{x}$, so we need the interval of final iteration no larger than 10^{-6}, so the stopping criterion is $\frac{b-a}{2} \leq 0.5 \cdot 10^{-6}$

Listing 1: bisection.m function

```
%% This program is to use the bisection method to solve for the root
%% of function cos(x)-2sin(x) and produce a table to show the value
%% of each iteration
```

```
%%%%%%%%

f=@(x) cos(x)-2*sin(x);
T = bisection1(f,0,1,0.5*10^(-6));

function c = bisection1(f,a,b,tol)
fprintf(1,[' i ', '     c\n']);
if f(a) == 0 %If a is already 0, then stop
      c = a;
    fprintf(1,'%2.1i %1.10f\n',[0, c]);
elseif f(b) == 0 %If b is already 0, then stop
       c = b;
     fprintf(1,'%2.1i %1.10f\n',[0, c]);
elseif f(a)* f(b) > 0
   %If they are at the same side of x axis, then not applied
   error('The initial condition is not satisfied');
else
   i = 0; % index of current iteration
   while (b-a)/2 > tol
%stopping criteria: If the distance between a and b greater than 2tol, stop
          i = i+1; % to the next iteration
          c = (a+b)/2;
          fc = f(c); % Store it to accerlerate the program
     if fc == 0
        fprintf(1,'%2.1i %1.10f\n',[i, c]);
        % If c is answer, then add it to the table
         break         % and break the loop
     end
     if fc * f(a) <0
       b=c;
     else
       a=c;
     end
      fprintf(1,'%2.1i %1.10f\n',[i, c]);
      % Add c to the table and go to next iteration
   end
end
end
```

Listing 2: output

```
>> bisection
 i        c
```

```
 1   0.5000000000
 2   0.2500000000
 3   0.3750000000
 4   0.4375000000
 5   0.4687500000
 6   0.4531250000
 7   0.4609375000
 8   0.4648437500
 9   0.4628906250
10   0.4638671875
11   0.4633789063
12   0.4636230469
13   0.4637451172
14   0.4636840820
15   0.4636535645
16   0.4636383057
17   0.4636459351
18   0.4636497498
19   0.4636478424
20   0.4636468887
```

5. My initial guess is $x_0 = 1$, and there are three stopping criteria in my algorithm:

   - If the difference between two iteration is relatively small, then we stop
   - If we have large amount iteration, we stop.
   - Since we wanna find the solution that lies between 1 and 3, so if $x > 3$, then we stop

Listing 3: newton.m function

```
%% This progeam applies newton's method to function f1
%% And make a table to see the value in each iteration
%% %%%%%


f1=@(x)94*(cos(x))^3-24*cos(x)+177*(sin(x))^2- ...
108*(sin(x))^4-72*(cos(x))^3*(sin(x))^2-65;
fd1=@(x) -282*(cos(x))^2*sin(x)+24*sin(x)+354*sin(x)*cos(x)- ...
432*(sin(x))^3*cos(x)+216*(cos(x))^2*(sin(x))^3-144*(cos(x))^4*sin(x);

a = newtons(f1,fd1,1,1000,3,0.5*10^(-6));

function x = newtons(f,fd, x, imax, xmax, tol)
fprintf(1,[' i ', '     x_i\n']);
```

```
if f(x) == 0 % If the input is already the root
     fprintf(1,'%2.1i %1.10f\n',[0, x]);
else
    i = 0; % index of iteration
    err = 10*tol; % Set the initial value of error
    while err > tol
        % if the difference between two iteration is less that tol, stop
        z = f(x)/fd(x); % original/derivative
        x = x -z; % The value for the new iteration
        err = abs(z); %the difference between two iteration
        i = i+1;
        if(i > imax)
            % If the number of iteration is sufficiently large, stop
            break
        end
        if(x > xmax) % If the answer go too far from what we expect, stop
            break
        end
      fprintf(1,'%2.1i %1.10f\n',[i, x]);
    end
end
end
```

Listing 4: output

```
>> newton
 i        x_i
 1   1.0176444552
 2   1.0281375444
 3   1.0347368040
 4   1.0389904857
 5   1.0417682978
 6   1.0435961292
 7   1.0448044592
 8   1.0456055968
 9   1.0461377623
10   1.0464916944
11   1.0467272762
12   1.0468841665
13   1.0469886908
14   1.0470583545
15   1.0471047512
16   1.0471357425
```

```
17  1.0471563415
18  1.0471698424
19  1.0471786645
20  1.0471815856
21  1.0471877173
22  1.0471769430
23  1.0471830768
24  1.0471905368
25  1.0472011250
26  1.0472826969
27  1.0472544034
28  1.0472352319
29  1.0472234944
30  1.0472149821
31  1.0472081249
32  1.0472127839
33  1.0472127839
```

---

6. (a) First, let's compute the derivatives of f

$$
\begin{aligned}
f'(x) &= \frac{dx^2 \sin x^2}{dx} \\
&= 2x \sin x^2 + 2x^3 \cos x^2 \\
f'(0) &= 0 \\
f''(x) &= \frac{d2x \sin x^2 + 2x^3 \cos x^2}{dx} \\
&= 2 \sin x^2 + 4x^2 \cos x^2 + 6x^2 \cos x^2 - 4x^4 \sin x^2 \\
&= 2 \sin x^2 + 10x^2 \cos x^2 - 4x^4 \sin x^2 \\
f''(0) &= 0 \\
f'''(x) &= \frac{d2 \sin x^2 + 10x^2 \cos x^2 - 4x^4 \sin x^2}{dx} \\
&= 4x \cos x^2 + 20x \cos x^2 - 20x^3 \sin x^2 - 16x^3 \sin x^2 - 8x^4 \cos x^2 \\
&= 4x \cos x^2 + 20x \cos x^2 - 36x^3 \sin x^2 - 8x^4 \cos x^2 \\
f'''(0) &= 0 \\
f^{(4)}(x) &= \frac{d4x \cos x^2 + 20x \cos x^2 - 36x^3 \sin x^2 - 8x^4 \cos x^2}{dx} \\
&= 4 \cos x^2 + 8x^2 \sin x^2 + 20 \cos x^2 - 40x^2 \sin x^2 - 108x^2 \sin x^2 - 72x^4 \cos x^2 - 32x^3 \cos x^2 + 16x^4 \sin x^2 \\
f^{(4)}(0) &= 24
\end{aligned}
$$

Thus, the multiplicity of the root is 4.

(b) The backward error is

$$\begin{aligned}|f(x_a)| &= |(0.01)^2 \sin(0.01^2)| \\ &= 9.9999999983 \times 10^{-9}\end{aligned}$$

And the forward error is

$$\begin{aligned}|r - x_a| &= |0 - 0.01| \\ &= 0.01\end{aligned}$$

7. We can calculate the derivative for the function first, which is

$$\begin{aligned}f'(x) &= (x-2)(x-3)(x-4) + (x-1)(x-3)(x-4) + (x-1)(x-3)(x-4) + (x-1)(x-2)(x-3) \\ f'(4) &= (x-1)(x-2)(x-3) \\ &= 3 \cdot 2 \cdot 1 \\ &= 6\end{aligned}$$

And we can get

$$\begin{aligned}\Delta x &= -\frac{h(x)}{f'(x)}\epsilon \\ &= -\frac{-4^6}{6}\epsilon \\ &= \frac{2048}{3}\epsilon \\ &\approx 682.67\epsilon\end{aligned}$$

When $\epsilon = 10^{-6}$

$$\begin{aligned}\Delta x &= 6.8267 \times 10^{-4} \\ \hat{x} &= 4 + 6.8267 \times 10^{-4}\end{aligned}$$

When $\epsilon = 10^{-8}$

$$\begin{aligned}\Delta x &= 6.8267 \times 10^{-6} \\ \hat{x} &= 4 + 6.8267 \times 10^{-6}\end{aligned}$$

And we can observe the answer calculated by MATLAB:

Listing 5: output

```
f11=@(x,ep) (x-1)*(x-2)*(x-3)*(x-4)-ep*x^6;
fzero(@(x) f(x, 10^(-6)),4)
```

```
ans =

   4.000682511531720

fzero(@(x) f(x, 10^(-8)),4)

ans =

   4.000006826651132

diary off
```