Problem Set (Chapter 2)

1 Write a user-defined function with function call `val = function_eval(f,a,b)` where f is an inline function, and a and b are constants such that $a<b$. The function calculates the midpoint m of the interval $[a,b]$ and returns the value of $f(a)+\frac{1}{2}f(m)+f(b)$. Execute the function for $f=e^{x/2}$, $a=-2$, $b=4$.

Solution

```
function val = function_eval(f,a,b)
m = (a+b)/2;
val = f(a) + f(m)/2 + f(b);
```

```
>> f = inline('exp(x/2)');
>> val = function_eval(f,-2,4)

val =
    8.5813
```

2 Write a user-defined function with function call `m = midpoint_seq(a,b,tol)` where a and b are constants such that $a<b$, and `tol` is a specified tolerance. The function first calculates the midpoint m_1 of the interval $[a,b]$, then the midpoint m_2 of $[a,m_1]$, then the midpoint m_3 of $[a,m_2]$, and so on. The process terminates when two successive midpoints are within `tol` of each other. Allow a maximum of 20 iterations. The output of the function is the sequence $m_1, m_2, m_3, \ldots$. Execute the function for $a=-4$, $b=10$, $\text{tol}=10^{-3}$.

Solution

```
function m = midpoint_seq(a,b,tol)
m(1)=(a+b)/2;
for i = 2:20,
    m(i) = (a+m(i-1))/2;
    if abs(m(i) - m(i-1)) < tol, break; end
end
```

```
>> m = midpoint_seq(-4,10,1e-3)

m =

  Columns 1 through 8

    3.0000    -0.5000    -2.2500    -3.1250    -3.5625    -3.7813    -3.8906    -3.9453

  Columns 9 through 14

   -3.9727    -3.9863    -3.9932    -3.9966    -3.9983    -3.9991
```

3 Write a user-defined function with function call `C = temp_conv(F)` where F is temperature in Fahrenheit, and C is the corresponding temperature in Celsius. Execute the function for $F=87$.

Solution

```
function C = temp_conv(F)
C = (F-32)*100/180;
```

```
>> C = temp_conv(87)

C =
   30.5556
```

4 Write a user-defined function with function call `P = partial_eval(f,a)` where f is a function defined symbolically, and a is a constant. The function returns the value of $f' + f''$ at $x = a$. Execute the function for $f = 3x^2 - e^{x/3}$, and $a = 1$.

Solution

```
function P = partial_eval(f,a)
del = diff(sym(f),'x') + diff(sym(f),2,'x');
x = a;
P = subs(del);
```

```
>> f = sym('3*x^2-exp(x/3)');
>> P = partial_eval(f,1)

P =
   11.3797
```

5 Write a user-defined function with function call `P = partial_eval(f,g,a)` where f and g are functions defined symbolically, and a is a constant. The function returns the value of $f' + g'$ at $x = a$. Execute the function for $f = x^2 + e^{-x}$, $g = \sin(0.3x)$, and $a = 0.8$.

Solution

```
function P = partial_eval(f,g,a)
del = diff(f,'x') + diff(g,'x');
x = a;
P = subs(del);
```

```
>> f = sym('x^2+exp(-x)');
>> g = sym('sin(0.3*x)');
>> P = partial_eval(f,g,0.8)

P =
    1.4421
```

6 Write a script file that employs any combination of the flow control commands to generate

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 2 & 0 & -1 & 0 & 0 \\ 2 & 0 & 3 & 0 & -1 & 0 \\ 0 & 2 & 0 & 4 & 0 & -1 \\ 0 & 0 & 2 & 0 & 5 & 0 \\ 0 & 0 & 0 & 2 & 0 & 6 \end{bmatrix}$$

Solution

```
clear
clc
A = zeros(6,6);
for i = 1:6,
  for j = 1:6,
      A(i,i) = i;
    if j == i+2,
      A(i,j) = -1;
elseif i == j+2,
    A(i,j) = 2;
    end
  end
end
```

```
>> A

A =
     1     0    -1     0     0     0
     0     2     0    -1     0     0
     2     0     3     0    -1     0
     0     2     0     4     0    -1
     0     0     2     0     5     0
     0     0     0     2     0     6
```

7 Write a script file that employs any combination of the flow control commands to generate

$$\mathbf{A} = \begin{bmatrix} 4 & 1 & -2 & 3 & 0 & 0 \\ 0 & 4 & -1 & 2 & 3 & 0 \\ 0 & 0 & 4 & 1 & -2 & 3 \\ 0 & 0 & 0 & 4 & -1 & 2 \\ 0 & 0 & 0 & 0 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 4 \end{bmatrix}$$

Solution

```
clear
clc
A = 4*eye(6);
for i = 1:6,
  for j = 1:6,
    if j == i+1,
      A(i,j) = (-1)^(i+1);
elseif j == i+2,
    A(i,j) = 2*(-1)^i;
    elseif j == i+3,
        A(i,j) = 3;
    end
  end
end
```

```
>> A

A =
     4     1    -2     3     0     0
     0     4    -1     2     3     0
     0     0     4     1    -2     3
     0     0     0     4    -1     2
     0     0     0     0     4     1
     0     0     0     0     0     4
```

8 Plot $\int_1^t e^{t-2x} \sin x dx$ versus $-1 \le t \le 1$, add grid and label.

Solution

```
>> syms t x
>> integ = int(exp(t-2*x)*sin(x),x,1,t);
>> ezplot(integ,[-1,1])
```

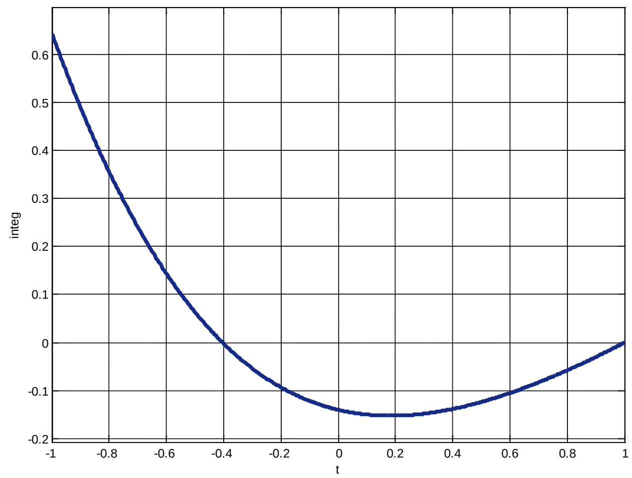

Figure Solution of Problem 8

9 Plot $\int_0^t (x+t)^2 e^{-(t-x)} dx$ versus $-2 \le t \le 1$, add grid and label.

Solution

```
>> syms x t
>> integ = int((x+t)^2*exp(-(t-x)),x,0,t);
>> ezplot(integ,[-2,1])
```

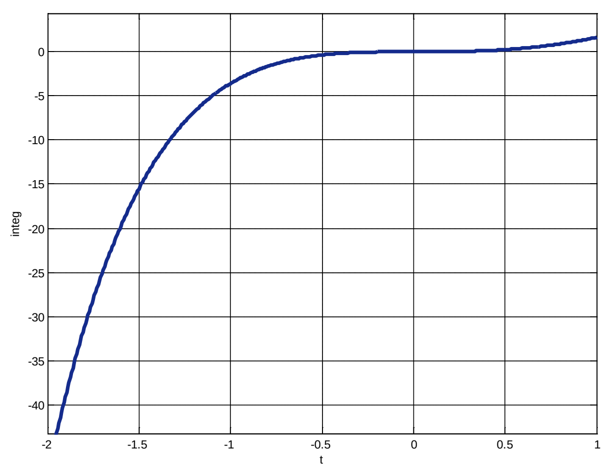

Figure Solution of Problem 9

10 Evaluate $\int_0^\infty \frac{\sin \omega}{\omega} d\omega$.

Solution

```
>> syms w
>> int(sin(w)/w,w,0,inf)

ans =

pi/2
```

11 Differentiate $f(t) = (t-1)\ln(t+1) - t\cos(t/2)$ with respect to t, make the outcome an inline function, and evaluate at $t = 1.3$.

Solution

```
syms t
f = sym('(t-1)*log(t+1)-t*cos(t/2)');
fp = inline(char(diff(f)));
fp(1.3)

ans =

    0.5606
```

12 Differentiate $g(x) = 2^{x-2}\sin x - e^{3-2x}$ with respect to x, make the outcome an inline function, and evaluate at $x = 0.9$.

Solution

```
syms x
g = sym('2^(x-2)*sin(x)-exp(3-2*x)');
gp = inline(char(diff(g)));
gp(0.9)

ans =

    7.1835
```

13 Plot $y_1 = \frac{1}{3}e^{-t}\sin\left(t\sqrt{2}\right)$ and $y_2 = e^{-t/2}$ versus $0 \le t \le 5$ in the same graph. Add grid, and label.

Solution

```
syms t
y1 = sym('(1/3)*exp(-t)*sin(sqrt(2)*t)');
y2 = sym('exp(-t/2)');
ezplot(y1,[0,5])
hold on
ezplot(y2,[0,5])
```

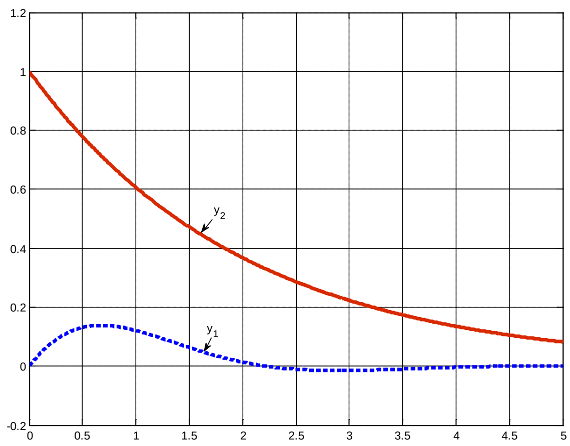

Figure Solution of Problem 13

14 Generate 100 points for each of the two functions in Problem 13 and plot versus $0 \le t \le 5$ in the same graph. Add grid, and label.

Solution

```
y1 = inline('(1/3)*exp(-t)*sin(sqrt(2)*t)');
y2 = inline('exp(-t/2)');
t = linspace(0,5);
for i = 1:100,
    yy1(i) = y1(t(i));
    yy2(i) = y2(t(i));
end
plot(t,yy1,t,yy2)
```

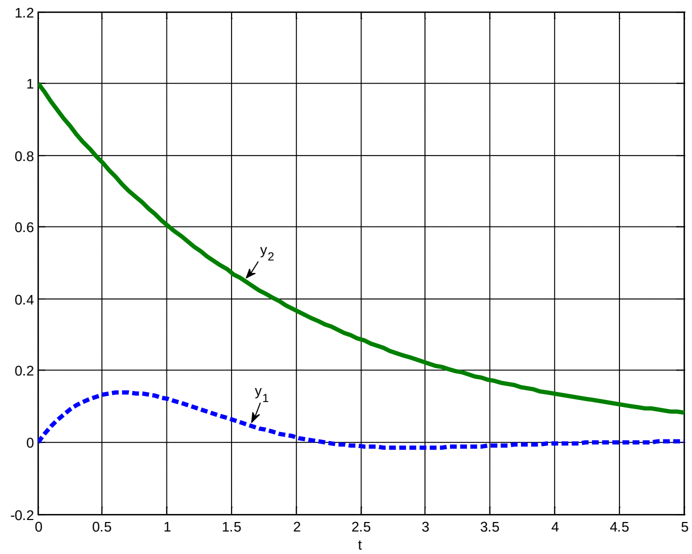

Figure Solution of Problem 14

15 Plot $u(x,t) = \cos(1.7x)\sin(3.2t)$ versus $0 \le x \le 5$ for four values of $t = 1, 1.5, 2, 2.5$ in a 2×2 tile. Add grid and title.

Solution

```
x = linspace(0,5); t = 1:0.5:2.5;
for i = 1:4,
    for j = 1:100,
    u(j,i) = cos(1.7*x(j))*sin(3.2*t(i));    % Generate 100 values of u for each t
    end
end

% Initiate figure
subplot(2,2,1), plot(x,u(:,1)), title('t = 1')
subplot(2,2,2), plot(x,u(:,2)), title('t = 1.5')
subplot(2,2,3), plot(x,u(:,3)), title('t = 2')
subplot(2,2,4), plot(x,u(:,4)), title('t = 2.5')
```

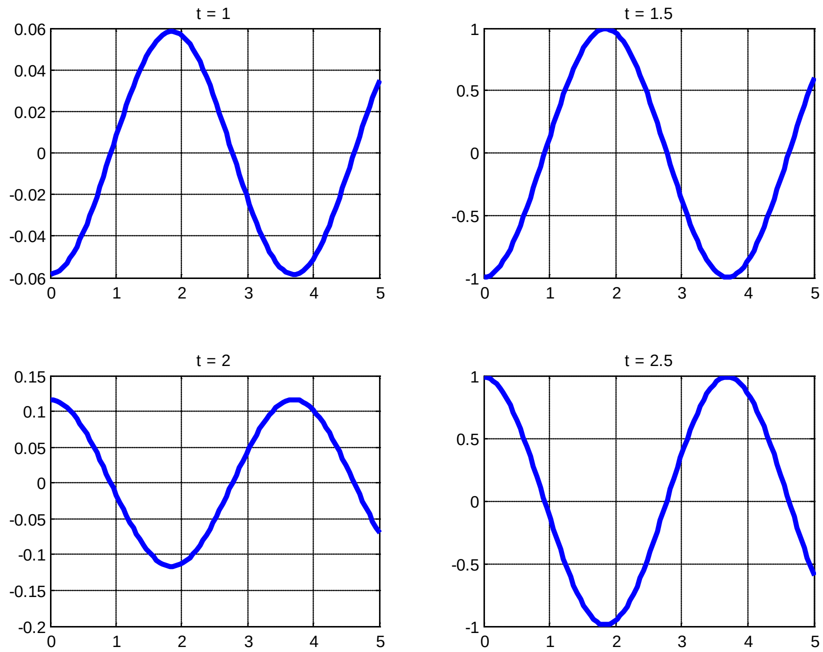

Figure Solution of Problem 15

16 Plot $u(x,t)=(1-\sin x)e^{-(t+1)}$ versus $0 \le x \le 5$ for two values of $t = 1,3$ in a 1×2 tile. Add grid and title.

Solution

```
x = linspace(0,5); t = [1,3];
for i = 1:2,
    for j = 1:100,
    u(j,i) = (1-sin(x(j)))*exp(-(t(i)+1));    % Generate 100 values of u for each t
    end
end

% Initiate figure
subplot(1,2,1), plot(x,u(:,1)), title('t = 1')
subplot(1,2,2), plot(x,u(:,2)), title('t = 3')
```

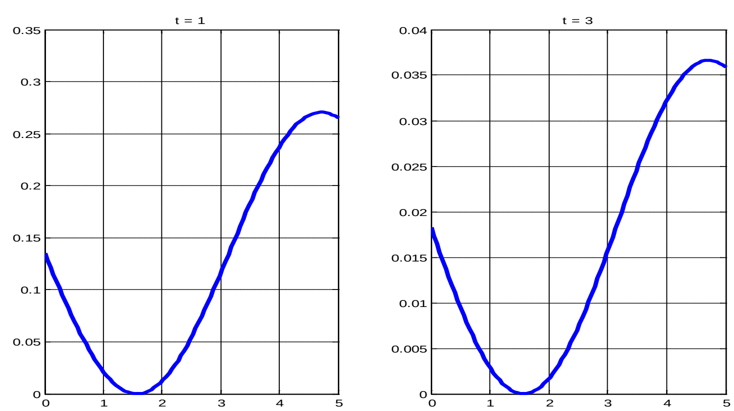

Figure Solution of Problem 16

17 Write a user-defined function with function call `[r k] = root_finder(f,x0,kmax,tol)` where f is an inline function, x_0 is a specified value, kmax is the maximum number of iterations, and tol is a specified tolerance. The function sets $x_1 = x_0$, calculates $|f(x_1)|$, and if it is less than the tolerance, then x_1 approximates the root r. If not, it will increment x_1 by 0.01 to obtain x_2, repeat the procedure, and so on. The process terminates as soon as $|f(x_k)| < \text{tol}$ for some k. The outputs of the function are the approximate root and the number of iterations it took to find it. Execute the function for $f(x) = x^2 - 3.3x + 2.1$, $x_0 = 0.5$, $\text{kmax} = 50$ $\text{tol} = 10^{-2}$.

Solution

```
function [r k] = root_finder(f,x0,kmax,tol)
x(1) = x0;
if abs(f(x(1)))<tol,
    r = x(1);
end
for k = 2:kmax,
    x(k) = x(k-1) + 0.01;
    if abs(f(x(k)))<tol,
        r = x(k); break, end
end
```

```
>> f = inline('x^2-3.3*x+2.1');
>> [r k] = root_finder(f,0.5,50,1e-2)

r =
    0.8600

k =
    37
```

18 Write a user-defined function with function call `[opt k] = opt_finder(fp,x0,kmax,tol)` where fp is the derivative (as an inline function) of a given function f, x_0 is a specified value, kmax is the maximum number of iterations, and tol is a specified tolerance. The function sets $x_1 = x_0$, calculates $|\text{fp}(x_1)|$, and if it is less than the tolerance, then x_1 approximates the critical point opt at which the derivative is near zero. If not, it will increment x_1 by 0.1 to obtain x_2, repeat the procedure, and so on. The process terminates as soon as $|\text{fp}(x_k)| < \text{tol}$ for some k. The outputs are the approximate optimal point and the number of iterations it took to find it. Execute the function for $f(x) = x + (x-2)^2$, $x_0 = 1$, $\text{kmax} = 50$ $\text{tol} = 10^{-2}$.

Solution

```
function [opt k] = opt_finder(fp,x0,kmax,tol)
x(1) = x0;
if abs(fp(x(1)))<tol,
    opt = x(1);
end
for k = 2:kmax,
    x(k) = x(k-1) + 0.1;
    if abs(fp(x(k)))<tol,
        opt = x(k); break, end
end
```

```
>> f = sym('x + (x-2)^2');
>> fp = inline(char(diff(f)))

fp =
     Inline function:
     fp(x) = 2*x - 3
```

```
>> [opt k] = opt_finder(fp,1,50,1e-2)

opt =
    1.5000

k =
     6
```

19 Evaluate $\left[x^2+e^{-a(x+1)}\right]^{1/3}$ when $a=-1$, $x=3$ by using an anonymous function in another anonymous function.

Solution

```
>> A = @(a,x)(x^2+exp(-a*(x+1)));
>> B = @(a,x)(A(a,x)^(1/3));
>> B(-1,3)

ans =

    3.9916
```

20 Evaluate $\sqrt{\left|x+\ln\left|1-e^{(a+2)x/3}\right|\right|}$ when $a=-3$, $x=1$ by using an anonymous function in another anonymous function.

Solution

```
>> A = @(a,x)(x+log(abs(1-exp((a+2)*x/3))));
>> B = @(a,x)(sqrt(abs(A(a,x))));
>> B(-3,1)

ans =

    0.5105
```

Problem Set (Chapter 3)

Bisection method (Section 3.2)

✍ In Problems 1–6, the given equation has a root in the indicated interval. Using bisection method, generate the first four midpoints and intervals (besides the original interval given) containing the root.

1 $\cos x+\sin x=0$, $[2,4]$

Solution

$f(2)>0$ and $f(4)<0$ so a root is indeed inside $[2,4]$. The first four midpoints and intervals are obtained as follows:

$c=\frac{1}{2}(2+4)=3$	$f(3)<0$	$[2,3]$	
$c=\frac{1}{2}(2+3)=2.5$	$f(2.5)<0$	$[2,2.5]$	
$c=\frac{1}{2}(2+2.5)=2.25$	$f(2.25)>0$	$[2.25,2.5]$	
$c=\frac{1}{2}(2.25+2.5)=2.375$	$f(2.375)<0$	$[2.25,2.375]$	Actual solution is 2.3562.