```
wordfile = export('hw104.mlx')
```

```
wordfile =
'C:\Users\jesse\Documents\MATLAB\Numerical Methods\Homework10.4\hw104.pdf'
```

```
clear, clc
syms g(x1, x2);
g(x1, x2) = (log((x1^2) + (x2^2))-sin(x1*x2)- log(2) - log(pi))^2 + (exp(x1-x2)+cos(x1*x2))^2;
%eval(g(1, 1)) using eval gives the actual value
k = 1;
N = 2;
x = [2, 2]';
x1 = x(1, 1);
x2 = x(2, 1);
tol = .05;
%tol = .01;
while k < N
    %step 3
    g1 = eval(g(x1, x2));
    z = gradient(g);
    %step 4
    z1 = eval(z(x1, x2));
    z0 = norm(z1);
    if z0 == 0
        break;
    end
    %step 5 make z a unit vector
    z1 = z1 / z0;
    a1 = 0;
    a3 = 1;
    g3 = eval(g(x1 - a3*z1(1, 1), x2 - a3*z1(2, 1)));
    %do step 7 and 8 while g3 is bigger than g1
    while(g3 >= g1)
        a3 = a3 / 2;
        g3 = eval(g(x1 - a3*z1(1, 1), x2 - a3*z1(2, 1)));
        if a3 < tol/2
            break; % no chance of improvement
        end
    end
    a2 = a3 / 2;
    g2 = eval(g(x1 - a2*z1(1, 1), x2 - a2*z1(2, 1)));
    %step 10
    h1 = (g2 - g1) / a2;
    h2 = (g3 -g2) / (a3 - a2);
    h3 = (h2 - h1)/a3;
    %step 11
    a0 = (.5)*(a2 - h1/h3);
    g0 = eval(g(x1 - a0*z1(1, 1), x2 - a0*z1(2, 1)));
    %step 12
    %if((g0 < g1) && (g0 < g2) && (g0 < g3))
```

```
     %   a = a0;
    %elseif((g1 < g0) && (g1 < g2) && (g1 < g3))
   %     a = a1;
   % elseif((g2 < g0) && (g2 < g1) && (g2 < g3))
    %    a = a2;
   % elseif((g3 < g0) && (g3 < g1) && (g3 < g2))
    %    a = a3;
   % end
   P = @(a)g1 + h1*a + h3*a*(a-a2);
   a = fminbnd(P, 0, 1);
    x = x - a*z1; % have this print out at the end
    x1 = x(1, 1);
    x2 = x(2, 1);
    k = k + 1;

end
disp(x);
```

```
    1.7361
    1.8044
```

```
x1guess = x1(1, 1);
x2guess = x2(1, 1);
```

same as previous, different tolerance

```
syms g(x1, x2);
g(x1, x2) = (log((x1^2) + (x2^2))-sin(x1*x2)- log(2) - log(pi))^2 + (exp(x1-x2)+cos(x1*x2))^2;
%eval(g(1, 1)) using eval gives the actual value
k = 1;
N = 2;
x = [2, 2]';
x1 = x(1, 1);
x2 = x(2, 1);
%tol = .05;
tol = .01;
while k < N
    %step 3
    g1 = eval(g(x1, x2));
    z = gradient(g);
    %step 4
    z1 = eval(z(x1, x2));
    z0 = norm(z1);
    if z0 == 0
        break;
    end
    %step 5 make z a unit vector
    z1 = z1 / z0;
    a1 = 0;
    a3 = 1;
    g3 = eval(g(x1 - a3*z1(1, 1), x2 - a3*z1(2, 1)));
```

```
    %do step 7 and 8 while g3 is bigger than g1
    while(g3 >= g1)
        a3 = a3 / 2;
        g3 = eval(g(x1 - a3*z1(1, 1), x2 - a3*z1(2, 1)));
        if a3 < tol/2
            break; % no chance of improvement
        end
    end
    a2 = a3 / 2;
    g2 = eval(g(x1 - a2*z1(1, 1), x2 - a2*z1(2, 1)));
    %step 10
    h1 = (g2 - g1) / a2;
    h2 = (g3 -g2) / (a3 - a2);
    h3 = (h2 - h1)/a3;
    %step 11
    a0 = (.5)*(a2 - h1/h3);
    g0 = eval(g(x1 - a0*z1(1, 1), x2 - a0*z1(2, 1)));

   P = @(a)g1 + h1*a + h3*a*(a-a2);
   a = fminbnd(P, 0, 1);
    x = x - a*z1; % have this print out at the end
    x1 = x(1, 1);
    x2 = x(2, 1);
    k = k + 1;
end
disp(x);
```

```
    1.7361
    1.8044
```

2, 10.4 3c with 1c's equations

```
syms f1(x1, x2) f2(x1, x2);
f1(x1, x2) = log((x1^2) + (x2^2)) - sin(x1*x2) - log(2) - log(pi);
f2(x1, x2) = exp(x1-x2) + cos(x1*x2);
f1x1 = diff(f1, x1);
f1x2 = diff(f1, x2);
f2x1 = diff(f2, x1);
f2x2 = diff(f2, x2);
k = 0;
x = [x1guess, x2guess]';
x1 = x(1, 1);
x2 = x(2, 1);
N = 3;
tol = .05;
while k < N
    J(1, 1) = eval(f1x1(x1, x2));
    J(1, 2) = eval(f1x2(x1, x2));
    J(2, 1) = eval(f2x1(x1, x2));
    J(2, 2) = eval(f2x2(x1, x2));
```

```
    F(1, 1) = eval(f1(x1, x2));
    F(2, 1) = eval(f2(x1, x2));

    y = -(J^-1)*F;
    x = x + y;
    x1 = x(1, 1);
    x2 = x(2, 1);
    if norm(y) < tol
        break;
    end
  k = k + 1;
end
disp(x);
```

```
    1.7725
    1.7725
```

10.4 5b

```
syms g(x1, x2);
g(x1, x2) = 100*((x1^2)-x2)^2 + (1-x1)^2;
k = 1;
N = 40;
x = [0, 0]';
x1 = x(1, 1);
x2 = x(2, 1);
tol = .005;
while k < N
    %step 3
    g1 = eval(g(x1, x2));
    z = gradient(g);
    %step 4
    z1 = eval(z(x1, x2));
    z0 = norm(z1);
    if z0 == 0
        break;
    end
    %step 5 make z a unit vector
    z1 = z1 / z0;
    a1 = 0;
    a3 = 1;
    g3 = eval(g(x1 - a3*z1(1, 1), x2 - a3*z1(2, 1)));
    %do step 7 and 8 while g3 is bigger than g1
    while(g3 >= g1)
        a3 = a3 / 2;
        g3 = eval(g(x1 - a3*z1(1, 1), x2 - a3*z1(2, 1)));
        if a3 < tol/2
            break; % no chance of improvement
        end
```

```
    end
    a2 = a3 / 2;
    g2 = eval(g(x1 - a2*z1(1, 1), x2 - a2*z1(2, 1)));
    %step 10
    h1 = (g2 - g1) / a2;
    h2 = (g3 -g2) / (a3 - a2);
    h3 = (h2 - h1)/a3;
    %step 11
    a0 = (.5)*(a2 - h1/h3);
    g0 = eval(g(x1 - a0*z1(1, 1), x2 - a0*z1(2, 1)));

     P = @(a)g1 + h1*a + h3*a*(a-a2);
    a = fminbnd(P, 0, 1);
    x = x - a*z1; % have this print out at the end
    x1 = x(1, 1);
    x2 = x(2, 1);
    k = k + 1;

end
disp(x);
```

```
    0.4303
    0.1801
```

```
disp(eval(g(x(1, 1), x(2, 1))))
```

```
    0.3271
```