HW Solutions from Mar 31
(Linearization and Multivariate Newton)

p. 114: 1, 3, 5, 7. Use the provided code to answer #5.

1. SOLUTIION: Using the linearization of f at $(0,1)$,

$$f(\mathbf{x}) \approx f(0,1) + \nabla f((0,1)(\mathbf{x} - (0,1))$$

Substituting $\mathbf{x} = (1/2, 3/2)$ gives us: $3 + [1,-2][1/2,1/2]^T = 5/2$.

3. SOLUTION: Increasing the fastest in the direction of the gradient. In this case,

$$f_x = y, \qquad f_y = x + 2y \quad \Rightarrow \quad \nabla f(x,y) = \langle y, x+2y \rangle \quad \Rightarrow \quad \nabla f(2,1) = \langle 1, 4 \rangle$$

5. We'll need a script file to output the gradient and Hessian:

$$\nabla f = \langle 4x^3 - 8xy, 4y^3 - 4x^2 + 2 \rangle \qquad Hf = \left[\begin{array}{rr} 12x^2 - 8y & -8x \\ -8x & 12y^2 \end{array} \right]$$

For example,

```
function [f,df,Hf]=testfunc01(x)
    temp=x;
    %Changing notation:
    x=temp(1); y=temp(2);
    f=x^4+y^4-4*x^2*y+2*y;
    df=[4*x^3-8*x*y; 4*y^3-4*x^2+2]
    Hf=[12*x^2-8*y, -8*x; -8*x, 12*y^2]
```

Then in the main script file, we might do the following, then look to see what points (if any) the algorithm converges to. After a while, you should see at least a few.

```
x=randn; y=randn;
MultiNewton(@testfunc01,[x(i);y(j)],50,1e-7)
```

7. This is really just a "notation recognition" problem. We were told that

$$\phi'(t) = \nabla f(\mathbf{a} - t\mathbf{u}) \cdot \mathbf{u} = 0$$

What is $\mathbf{u}$? We're moving in the direction of the gradient. Therefore, defining the notation using $\mathbf{x}_i, \mathbf{x}_{i+1}$, at the optimal value of t, t^*, we have:

$$\mathbf{a} = \mathbf{x}_i, \qquad \mathbf{u} = \nabla f(\mathbf{x}_i), \qquad \mathbf{x}_{i+1} = \mathbf{a} - t^*\mathbf{u} = \mathbf{x}_i - t^* \nabla f(\mathbf{x}_i)$$

Therefore, substituting these values into $\phi'(t)$, we get:

$$\nabla f(\mathbf{x}_{i+1}) \cdot \nabla f(\mathbf{x}_i) = 0$$