```
> restart;
  with( linalg ):
Warning: new definition for   norm
Warning: new definition for   trace
```

Solution of Homework Problems from Chapter 3 of Peressini, Sullivan, and Uhl

Prepared by Douglas B. Meade (meade@math.sc.edu)
9 November 1995

WWW URL: http://www.math.sc.edu/~meade/math524/worksheets/ch3hw1.ms

---

```
>
```

In a previous worksheet (newton.ms) we saw Maple implementations of Newton's method for a single function of one real variable.

```
> newtonZ := x -> x-g(x)/dg(x):
> newtonM := x -> x-df(x)/df2(x):
>
```

In this worksheet we will need the Newton iterates for a system of equations which are functions of more than one variable.

```
> NewtonZ := x -> convert( evalm(
                      x - inverse( gradg(x) ) &* g(x) ),
                      vector ):
> NewtonM := x -> convert( evalm(
                       x - inverse( Hf(x) ) &* gradf(x) ),
                       vector ):
>
```

Note that these definitions are fairly straightforward, except that special care is needed to ensure that the input and output of the Newton procedures are compatible, i.e. the output needs to be converted from an n x 1 matrix to an n-dimensional vector. In the following definitions, ignore the first and third lines of the definitinos. In each case, the second line is the one that contains the definition of the Newton iterates.

```
>
```

>

Problem #1d

This necessary Newton iterates for this problem can be found using the newtonZ function, after the function, g, and its first derivative, dg, are defined.

```
> g := x -> x^(1/3);
```

$$g := x \to x^{1/3}$$

```
> dg := x->1/3*x^(-2/3);
```

$$dg := x \to \frac{1}{3}\,\frac{1}{x^{2/3}}$$

>

The first 10 iterates for the solution of g(x)=0 are

```
> for k from 0 to 9 do
    x[k+1] := newtonZ( x[k] )
  od;
```

$$x_1 := -2\,x_0$$

$$x_2 := 4\,x_0$$

$$x_3 := -8\,x_0$$

$$x_4 := 16\,x_0$$

$$x_5 := -32\,x_0$$

$$x_6 := 64\,x_0$$

$$x_7 := -128\,x_0$$

$$x_8 := 256\,x_0$$

$$x_9 := -512\,x_0$$

$$x_{10} := 1024\,x_0$$

The pattern is obvious: x[k] = (-2)^k x[0]. Thus, this is a geometric sequence with radius r = -2. This sequence does not converge, unless x[0]=0.

>

>

Problem 2

Once we define the function, and its first two derivatives, we can use the newtonM routine to compute the Newton iterates.

> f := x -> x^(4/3);

$$f := x \to x^{4/3}$$

> df := x -> 4/3*x^(1/3);

$$df := x \to \frac{4}{3} x^{1/3}$$

> df2 := x -> 4/9*x^(-2/3);

$$df2 := x \to \frac{4}{9} \frac{1}{x^{2/3}}$$

>

The first 10 iterates for the minimizer of f are

```
> for k from 0 to 9 do
    x[k+1] := newtonM( x[k] )
  od;
```

$$x_1 := -2\, x_0$$

$$x_2 := 4\, x_0$$

$$x_3 := -8\, x_0$$

$$x_4 := 16\, x_0$$

$$x_5 := -32\, x_0$$

$$x_6 := 64\, x_0$$

$$x_7 := -128\, x_0$$

$$x_8 := 256\, x_0$$

$$x_9 := -512\, x_0$$

$$x_{10} := 1024\, x_0$$

THIS IS THE SAME SEQUENCE AS IN PROBLEM 1d!

>

>

Problem #4

To minimimze this function of two variables we will use the NewtonM function. But, first, we need to define the function, f, its gradient, gradf, and its Hessian, Hf.

>

> f := x-> 2*x[1]^4 + x[2]^2 - 4*x[1]*x[2] + 5*x[2];

$$f := x \to 2\,x_1^{\,4} + x_2^{\,2} - 4\,x_1\,x_2 + 5\,x_2$$

> gradf := x -> matrix(2,1, [[8*x[1]^3-4*x[2]], [2*x[2]-4*x[1]+5]]);

$$gradf := x \to \mathrm{matrix}\left(\left[\left[8\,x_1^{\,3} - 4\,x_2\right], \left[2\,x_2 - 4\,x_1 + 5\right]\right]\right)$$

> Hf := x -> matrix(2,2, [[24*x[1]^2,-4], [-4, 2]]);

$$Hf := x \to \mathrm{matrix}\left(\left[\left[24\,x_1^{\,2}, -4\right], [-4, 2]\right]\right)$$

> x[0] := [0,0];

$$x_0 := [0, 0]$$

> x[1] := NewtonM(x[0]);

$$x_1 := \begin{bmatrix} \frac{5}{4} & 0 \end{bmatrix}$$

> x[2] := NewtonM(x[1]);

$$x_2 := \begin{bmatrix} \frac{85}{118} & \frac{-125}{118} \end{bmatrix}$$

>

END OF WORKSHEET

>