Math 5329, Final

Name ______________________________

1. If $f(x) = (x - r)^m$, where $m > 1$, show that Newton's method will converge to the multiple root r, no matter where we start, but only linearly.

2. Give an example of a linear system $Ax = b$ for which the Jacobi iteration converges, although the matrix A is not diagonal dominant. (Hint: If $A = L + D + U$, the Jacobi method converges if and only if all eigenvalues of $D^{-1}(L + U)$ are less than one in absolute value.)

3. Find the weights w_1, w_2 and sample points r_1, r_2 for the Gauss 2 point quadrature rule $\int_0^h f(x)\ dx \approx w_1 h f(r_1 h) + w_2 h f(r_2 h)$. (Hint: You should make some simplifing assumptions first, based on symmetry)

4. Consider the multi-step method:
$U(t_{k+1}) = -4U(t_k) + 5U(t_{k-1}) + 4hf(t_k, U(t_k)) + 2hf(t_{k-1}, U(t_{k-1}))$

   a. Calculate the truncation error. Is the method consistent?

   b. Determine if the method is stable.

   c. Is the method implicit or explicit?

5. If $L_N(x)$ is the Lagrange polynomial of degree N which interpolates to $f(x) = e^{2x}$ at $x = 1, 2, 3, ...N+1$, prove that $L_N(0)$ does NOT converge to $f(0) = 1$, as $N \to \infty$.

6. A 3 by 3 matrix A has eigenvalues very near $2, 8$ and 9. If the shifted power method (note: not INVERSE shifted power method) $x_{n+1} = (A - pI)x_n$ is used, what value of p should be used if we want to maximize the speed of convergence to find the eigenvalue near 9? (Hint: the rate of convergence of the power method is the ratio of the second largest (in absolute value) eigenvalue to the largest.)

7. The following MATLAB program solves a linear system $Ax = b$ using the Gauss-Jordan algorithm, in which A is reduced to diagonal form rather than upper triangular form, during the forward elimination (no pivoting is done in this program). For large N, approximately how many multiplications are done by this program? How does this algorithm compare in speed to normal Gauss elimination?

```
function x = GJ(A,b,N)
for i=1:N
   for j=1:N
      if (j==i)
         continue
      end
      r = A(j,i)/A(i,i);
      for k=i:N
         A(j,k) = A(j,k) - r*A(i,k);
      end
      b(j) = b(j) - r*b(i);
   end
end
for i=1:N
   x(i) = b(i)/A(i,i);
end
```