Gauss Quadrature

Gaussian quadrature is a quadrature rule constructed to yield an exact result for polynomials of degree 2*n* – 1.

MATLABCODE:

```
function [integral, k] = gaussquad(fn, x0, x1, pts)
  % This function calculates the Gauss Numerical Integration
  % for finite integrals
  % INPUTS:
  %     fn = the function in question
  %     x0, x1 = the interval of the integration
  %     pts = the number of points
  % OUTPUTS:
  %     integral
  %     k = number of iterations needed to adapt
  integral = 1;
  lastintegral = 0;
  n = pts;
  [xg, c] = gausswt(n);
  k = 0;
  while abs(lastintegral - integral) > 10^-5;
    k = k+1;
    integral = lastintegral;
    lastintegral =0;
    x = linspace(x0,x1,n);
    for k = 1:pts-1:n-1
      if pts == 2,
        xm = (x(k)+x(k+1))/2;
        h = (x(k+1)-x(k));
      else
        xm = x(k+1);
        h = (x(k+2)-x(k));
      end
      lastintegral = lastintegral + (c(1)*fn(xm+(h*xg(1)/2))*(h/2)
                                  + c(2)*fn(xm+(h*xg(2))/2)*(h/2)
                                  + c(3)*fn(xm+(h*xg(3))/2)*(h/2));
    end
    n = n+pts-1;
   end
end
```

```
function [x, c] = gausswt(n)
  % this function is used to return the weights and arguments
  % of gauss quadrature given the number of points "n"
  %   INPUTS:
  %       n = number of points
  %   OUTPUTS:
  %       x = gauss arguments
  %       c = weighting factors

  if n == 2
    x = [-0.577350269, 0.577350269, 0];
    c = [1, 1, 0];
  end
  if n == 3
    x = [-0.774596669, 0, 0.774596669];
    c = [0.5555556,0.8888889, 0.5555556];
  end
end
```

TEST CODE:

```
>> [I,k] = gaussquad(fn,0,pi,2)
I = -4.9348
k =  16
>> [I,k] = gaussquad(fn,0,pi,3)
I = -4.9348
k =  11
```