Homework 3

- **Finish reading Chap. 1-3 of Using MPI by Gropp et al.**

- **Look at the parallel routine to compute π (calcpip) in the Codes directory (or Examples directory on Wopr)**

- **Due Friday Oct. 25: Modify the calcpip.f routine to do a more accurate integration (Simpson's Rule) described in the next slide**
  - Provide a listing of all subroutines and test this algorithm for different numbers of processors using parallel run.qsub batch submit procedure on davistron
  - Provide the CPU time as a function of the number of processors (up to 8) and the answer

Homework 3

. A more accurate alternative to the trapezoidal rule is Simpson's rule. The basic idea is to approximate the graph of $f(x)$ by arcs of parabolas rather than line segments. Suppose that $p < q$ are real numbers, and let r be the midpoint of the segment $[p, q]$. If we let $h = (q - p)/2$, then an equation for the parabola passing through the points $(p, f(p))$, $(r, f(r))$, and $(q, f(q))$ is

$$y = \frac{f(p)}{2h^2}(x - r)(x - q) - \frac{f(r)}{h^2}(x - p)(x - q) + \frac{f(q)}{2h^2}(x - p)(x - r).$$

If we integrate this from p to q, we get

$$\frac{h}{3}[f(p) + 4f(r) + f(q)].$$

Thus, if we use the same notation that we used in our discussion of the trapezoidal rule and we assume that n, the number of subintervals of $[a, b]$, is even, we can approximate

$$\int_a^b f(x)dx \doteq \frac{h}{3}[f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + \cdots + 2f(x_{n-2}) + 4f(x_{n-1}) + f(x_n)].$$

Assuming that n/p is even, write

a. a serial program and

b. a parallel program that uses Simpson's rule to estimate $\int_a^b f(x)dx$.

From Parallel Programming with MPI by Pacheco