

ASSIGNMENT 3

HPC1 Fall 2008

Due Date: *Friday, October 24*

(please submit your report electronically to the instructor, in one file, PDF preferred, named *hw2-youruserid.pdf*)

Problem 1: Write a program to compute π by the summation:

$$\frac{\pi}{4} = \sum_{i=0}^{N \rightarrow \infty} \frac{(-1)^i}{2i+1},$$

and use **OpenMP** to parallelize the code. Determine the performance of your code as a function of N terms in the sum, and P processors. Note that, depending on the granularity of your timer, it may be necessary to repeat the calculation (say j times, such that jN is a convenient timing interval) and time the total to get reliable average times, especially for smaller values of N . Plot the performance, execution time, parallel speedup, and efficiency as a function of P (note the utility of logarithmic scales!). On U2 compute nodes you will be limited to 2 processors, but there are a couple of machines that you can use for larger SMPs. **u2-dev** is dual-socket quad-core 'Clovertown' Xeon machine with 8 cores total that you can log into directly (watch out for interference with your timings coming from other users by using something like "top," as this machine is not part of the PBS pool). If you are interested in running on an even larger shared memory system, let me know and you can also use the 64 processor SGI Altix (Lennon), but that is not a requirement for this problem. **Hint:** simple pseudo-code for splitting the sum into P partial sums might look something like:

```
myID = MyProcNumber()
Np = TotalProcNumber()
mySum = 0
do i=myID*(NsumTerms/Np)+1, (myID+1)*(NsumTerms/Np), 2
    mySum = mySum + 1.0/(2*i-1)
    mySum = mySum - 1.0/(2*i+1)
end do
CollectPartialSums(S)
```

Problem 2: Repeat problem 1 using **MPI** instead of **OpenMP**.