# COSC 462 Fall 2024 : Programming Assignment 2

**Date: Oct 7, 2024** **Total points: 90** **Due: 11:59 PM, Oct 20, 2024**

(40 points) The following is a polynomial in a single unknown variable $x$ with degree $n-1$ where $a_1 \cdots a_{n-1}$ are known constants.

$$P(x) = a_1 x + a_2 x^2 + a_3 x^3 + \cdots a_n x^n$$

The goal here is to compute the value of the polynomial at a given value of $x$, say, $x_0$, that is, evaluate $P(x_0)$. Your programming assignment is to implement a parallel prefix algorithm and evaluate this polynomial with degree 64 at a value of $x = 0.5$ using process counts $P = 1, 2, 4, 8, 16, 32$ and 64. Specifically:

(i) (5 points) Assign 64 coefficients $a_1, \cdots, a_{64}$ by generating 64 random numbers between 0 and 1.

(ii) (35 points) Using MPI_Scan, write a parallel prefix-based code to evaluate this polynomial at $x = 0.5$ using $P =$ 64, 32, 16, 8, 4, 2 and 1.

Note that the value of the polynomial $P(x = 0.5)$ must remain the same irrespective of the number of processes used.

(50 points) **Numerical Integration**
An integral representation of $\pi$ is:

$$\int_0^1 dx \frac{4}{1+x^2} = \pi$$

The discrete version of this integral is:

$$\sum_{i=0}^{n-1} \frac{4}{1+x_i^2} \Delta \approx \pi$$

where $\Delta = \frac{1}{n}$ and $x_i = (i + 0.5)\Delta$. Using this notation, a sequential code to numerically compute the value of $\pi$ is:

```
# include <stdio.h>
# nsteps 10000000
void main()
{
  double x, sum=0.0;
  double step = 1.0/nsteps;
  for (int i=0; i<nsteps; i++)
    {
      x = (i+0.5)*step;
      sum += 4.0/(1.0+x*x);
    }
  double pi = sum*step;
  printf("The value of pi = %f\n",pi);
}
```

(i) (10 points) Write a code that parallelizes the $for$ loop and takes $nsteps$ as an input variable.

(ii) (20 points) Keeping *nsteps* fixed at $10^7$, compute the value of $\pi$ using 1, 2, 4, 8, 16 and 32 MPI processes. Plot the parallel execution times against the number of MPI processes ($P$).

(iii) (20 points) For $P = 16$ fixed, compute the value of $\pi$ using $nsteps = 10^3, 10^4, 10^5, 10^6$ and $10^7$. Plot the parallel execution times against the value of MPI *nsteps*.

In parts (ii) and (iii), the value of $\pi$ should be computed to at least 6 decimal places.