

# Assignment #1

HPC Workshop



# Program Description

```
! This program calculates an integral of a polynomial
! Inputs:
! integer      np      = the order of the polynomial
! double precision poly = array of polynomial coefficients
! double precision a    = lower bound for integration
! double precision b    = upper bound for integration
! double precision eps  = desired accuracy
! integer      jmax    = maximum subdivisions for the
!                    trapezoidal integration
!
! Suggested input:
! 3
! 1
! 2
! 3
! 4
! 0 1000
! 1e-20 20
!
! Output:
! Integral over the range 0 to 1000 =      1.001001e+12
```

# doint.c

```
...
...
int main(int argc, char **argv) {
    FLT a,b,s,eps;
    int jmax;
    int i;
    char *aform="Integral over the range %lg to %lg = %20.7lg\n";
    /*
    ! We are integrating a polynomial read in the order (np)
    ! then the coefficients (poly).
    */
    scanf("%d",&np);
    for(i=0;i<=np;i++) {
        scanf("%lf",&poly[i]);
    }
    /*
    ! Read the lower (a) and upper (b) integrations bounds.
    */
    scanf("%lf %lf",&a,&b);
    /*
    ! Read the desired accuracy (eps) and the maximum
    ! subdivisions for the trapezoidal integration (jmax).
    */
    scanf("%lf %d",&eps,&jmax);
    s=qsimp(myfunc,a,b,eps,jmax);
    printf(aform,a,b,s);
}
```

# doint.f90

```
...
...
program doint
  use numz
  implicit none
  external myfunc
  real(b8) a,b,s,eps,myfunc
  character (len=100)::aform
  integer jmax,i
  aform=' ( "Integral over the range ",f10.5," to ",f10.5,"=",g20.7) '
  ! We are integrating a polynomial read in the order (np)
  ! then the coefficients (poly).
  read(*,*)np
  read(*,*)(poly(i),i=0,np)
  ! Read the lower (a) and upper (b) integrations bounds.
  read(*,*)a,b
  ! Read the desired accuracy (eps) and the
  ! maximum subdivisions for the trapezoidal
  ! integration (jmax).
  read(*,*)eps,jmax
  call qsimp(myfunc,a,b,s,eps,jmax)
  write(*,aform)a,b,s
end program
```

# Example Run

```
[tkaiser@ra5 assignments]$ ifort doint.f90 -o fint
```

```
[tkaiser@ra5 assignments]$ icc doint.c -o cint
```

```
[tkaiser@ra5 assignments]$ cat input
```

```
3
```

```
1
```

```
2
```

```
3
```

```
4
```

```
0 1000
```

```
1e-20 20
```

```
[tkaiser@ra5 assignments]$ ./fint < input
```

```
Integral over the range 0.00000 to 1000.00000 = 0.1001001E+13
```

```
[tkaiser@ra5 assignments]$ ./cint < input
```

```
Integral over the range 0 to 1000 = 1.001001e+12
```

# To do:

The program `doint.*` calculates the integral of a polynomial over a given range. The coefficients and the range are input along with tolerance parameters.

Rewrite as an MPI program that works on any number of processors.

(1) Each processor calculates an equal portion of the integral in parallel.

(2) Each then should print its given range and portion of the integral.

(3) The root processor then prints the value of the whole integral as the sum of the individual parts.

# Extra challenge

Have the first processor do its portion of the calculation then pass the results to the second processor which adds its portion and so on

Have the last processor print the final value.