Q1: factorial using n processes

#include <stdio.h>

#include <mpi.h>

int factorial(int x)

{

int ans = 1;

for(int i = x; i > 0; i--)

ans = ans\*i;

return ans;

}

void main(int argc, char\* argv[])

{

int rank,size, c;

int N = 5;

int a[N], b[N];

MPI\_Init(&argc,&argv);

MPI\_Comm\_size(MPI\_COMM\_WORLD,&size);

MPI\_Comm\_rank(MPI\_COMM\_WORLD,&rank);

if(rank == 0)

{

printf("Enter %d values: ",N);

for(int i=0; i<N; i++)

{

scanf("%d", &a[i]);

}

}

MPI\_Scatter(a, 1, MPI\_INT, &c, 1, MPI\_INT, 0, MPI\_COMM\_WORLD);

c = factorial(c);

MPI\_Gather(&c, 1, MPI\_INT, b, 1, MPI\_INT, 0, MPI\_COMM\_WORLD);

if(rank == 0)

{

for(int i=0; i<N; i++)

{

printf("Root recieved %d from process %d \n", b[i], i);

}

}

MPI\_Finalize();

}

Q2: Average of NxM array

#include <stdio.h>

#include <mpi.h>

float avg(int \*x, int M)

{

float ans = 0.0;

for(int i = 0; i < M; i++)

ans = ans+x[i];

return ans/M;

}

void main(int argc, char\* argv[])

{

int rank,size;

float c;

int N = 5;

int M;

MPI\_Init(&argc,&argv);

MPI\_Comm\_size(MPI\_COMM\_WORLD,&size);

MPI\_Comm\_rank(MPI\_COMM\_WORLD,&rank);

if(rank == 0)

{

printf("Enter value of M: ");

scanf("%d", &M);

}

MPI\_Bcast(&M, 1, MPI\_INT, 0, MPI\_COMM\_WORLD);

int b[M];

float x[N];

int a[N\*M];

if(rank == 0)

{

printf("Enter %d values: ", N\*M);

for(int i=0; i<N\*M; i++)

{

scanf("%d", &a[i]);

}

}

MPI\_Scatter(a, M, MPI\_INT, b, M, MPI\_INT, 0, MPI\_COMM\_WORLD);

c = avg(b, M);

printf("value of average in rank %d : %f \n", rank, c);

MPI\_Gather(&c, 1, MPI\_FLOAT, x, 1, MPI\_FLOAT, 0, MPI\_COMM\_WORLD);

if(rank == 0)

{

float ans = 0.0;

for(int i = 0; i < N; i++)

ans = ans+x[i];

ans = ans/N;

printf("\nFinal Avg = %f \n", ans);

}

MPI\_Finalize();

}

Q3: Non vowels

#include <stdio.h>

#include <mpi.h>

#include <string.h>

void main(int argc, char\* argv[])

{

int rank,size;

int N = 5;

char s[100];

char c[100];

int b[N];

MPI\_Init(&argc,&argv);

MPI\_Comm\_size(MPI\_COMM\_WORLD,&size);

MPI\_Comm\_rank(MPI\_COMM\_WORLD,&rank);

if(rank == 0)

{

printf("Enter String: ");

scanf("%s", s);

}

int l = strlen(s)/N;

MPI\_Bcast(&l, 1, MPI\_INT, 0, MPI\_COMM\_WORLD);

MPI\_Scatter(s, l, MPI\_CHAR, c, l, MPI\_CHAR, 0, MPI\_COMM\_WORLD);

int x = 0;

for(int i = 0; i<l; i++)

{

if(c[i] != 'a' && c[i] != 'e' && c[i] != 'i' && c[i] != 'o' && c[i] != 'u' && c[i] != 'A' && c[i] != 'E' && c[i] != 'I' && c[i] != 'O' && c[i] != 'U')

x++;

}

MPI\_Gather(&x, 1, MPI\_INT, b, 1, MPI\_INT, 0, MPI\_COMM\_WORLD);

if(rank == 0)

{

x = 0;

for(int i=0; i<N; i++)

{

x+=b[i];

printf("Root recieved %d from process %d \n", b[i], i);

}

printf("Total non-vowel count : %d \n", x);

}

MPI\_Finalize();

}

Q4: Alternating string concat

#include <stdio.h>

#include <mpi.h>

#include <string.h>

void main(int argc, char\* argv[])

{

int rank,size;

int N = 5;

char s1[100];

char s2[100];

char b1[30];

char b2[30];

char ans[100];

MPI\_Init(&argc,&argv);

MPI\_Comm\_size(MPI\_COMM\_WORLD,&size);

MPI\_Comm\_rank(MPI\_COMM\_WORLD,&rank);

if(rank == 0)

{

printf("Enter String 1: ");

scanf("%s", s1);

printf("Enter String 2: ");

scanf("%s", s2);

}

int l = strlen(s1)/N;

MPI\_Bcast(&l, 1, MPI\_INT, 0, MPI\_COMM\_WORLD);

MPI\_Scatter(s1, l, MPI\_CHAR, b1, l, MPI\_CHAR, 0, MPI\_COMM\_WORLD);

MPI\_Scatter(s2, l, MPI\_CHAR, b2, l, MPI\_CHAR, 0, MPI\_COMM\_WORLD);

char c[2\*l];

for(int i = 0; i < l; i++)

{

c[2\*i] = b1[i];

c[2\*i+1] = b2[i];

}

printf("process %d returns: %s \n", rank, c);

MPI\_Gather(c, 2\*l, MPI\_CHAR, ans, 2\*l, MPI\_CHAR, 0, MPI\_COMM\_WORLD);

if(rank == 0)

{

printf("\n\nFinal String : %s \n", ans);

}

MPI\_Finalize();

}