MPI 超入門 ver 1.30 (2021.12.8)

MPI とはメッセージパッシングを用いた並列処理のためのライブラリである。並列計算モデルとしては SPMD モデルを用いている。MPI は 100 以上の関数を有するが、ここでは今回の課題で用いる為に必要最小限度の事項について説明する。

コンパイルと実行

コンパイルおよび実行方法は以下のとおりである。

コンパイル: `mpicc -O3 -o program program.c -lm`

実行方法: `mpirun -np num -machinefile machinefile program`

ここでは、`program.c` はプログラムファイル名、`program` は実行プログラム名、`num` は実行するプロセス数、`machinefile` はプログラムを実行するマシン名のリストが記述されたファイルである。

`machinefile` の例

red01.ced.cei.uec.ac.jp

red02.ced.cei.uec.ac.jp

red03.ced.cei.uec.ac.jp

red04.ced.cei.uec.ac.jp

サンプルプログラム

サンプルプログラムとして、通信手順が今回の課題と同じであるプログラムを示す。

ここでは入出力操作は全て rank=0 のプロセスで行っている。

```
#include <stdio.h>
#include <math.h>
#include "mpi.h"                                     /* MPI のヘッダファイル (必須) */

main(int argc, char* argv[])
{
    int i;
    double x[1024];
    int n, p, rank, nn, is, ie;

    MPI_Init(&argc, &argv);                          /* MPI 初期化 (必須) */
    MPI_Comm_size(MPI_COMM_WORLD, &p);              /* プロセス数を変数 p に返す */
    MPI_Comm_rank(MPI_COMM_WORLD, &rank);           /* プロセスのランクを変数 rank に返す */

    if(rank == 0)
        scanf("%d\n", &n);
    MPI_Bcast(&n, 1, MPI_INT, 0, MPI_COMM_WORLD);
```

```

/* プロセス 0 から変数 n の内容を他の全てのプロセスに伝える */

nn=n/p;
is=nn*rank;
ie=is+nn;

for(i=0 ; i < n ; i++)
    x[i]=0;

for(i=is ; i < ie ; i++)
    x[i]=i/(double)n;

for(i=0 ; i < p ; i++){
    MPI_Bcast(&x[i*nn], nn, MPI_DOUBLE, i, MPI_COMM_WORLD);
    /* プロセス i から変数 x の nn 個の内容を他の全てのプロセスに伝える */
}

if(rank == 0){
    for(i=0 ; i < n ; i++)
        printf("%5.3f ", x[i]);
    printf("\n");
}

MPI_Finalize();      /* MPI 終了操作 (必須) */
}

```

実行例:

```

mpicc -O3 -o mpi-test mpi-test.c -lm
echo 8 | mpirun -np 4 -machinefile mac ./mpi-test
0.000 0.125 0.250 0.375 0.500 0.625 0.750 0.875

```

注1: `mpirun` コマンドを実行する前に一度だけ、並列計算で利用するマシンに `ssh` でリモートログインしておくこと。

注2: この説明終了後、直ちに上記のサンプルを必ず試すこと。