

Programowanie współbieżne i rozproszone	PWIR07	
24.05.2022	P2	Szymon Zwoliński

Zaimplementuj rozproszone obliczenie przybliżenia liczby π . Skrypt wraz z dokumentacją opisującą go umieść w repozytorium w odpowiednim folderze.

```

#include <stdlib.h>
#include <stdio.h>
#include <math.h>
#include <string.h>
#include <stdio.h>
#include <cstdlib>
#include <ctime>
#include "mpi.h"

int main(int argc, char* argv[])
{
    std::srand(time(NULL));
    int niter = 100;
    double x, y;
    int i, j, count = 0, mycount;
    double z;
    double pi;
    int myid, numprocs, proc;
    MPI_Status status;
    int master = 0;
    int tag = 123;

    MPI_Init(&argc, &argv);
    MPI_Comm_size(MPI_COMM_WORLD, &numprocs);
    MPI_Comm_rank(MPI_COMM_WORLD, &myid);

    mycount = 0;
    for (i = 0; i < niter; i++) {
        x = (double)(rand() / float(RAND_MAX));
        y = (double)(rand() / float(RAND_MAX));
        z = x * x + y * y;
        if (z <= 1)
        {
            mycount++;
        }
    }
    if (myid == 0) {
        count = mycount;
        for (proc = 1; proc < numprocs; proc++) {
            MPI_Recv(&mycount, 1, MPI_REAL, proc, tag, MPI_COMM_WORLD, &status);
            count += mycount;
        }
        pi = 4.*((double)count / (niter * numprocs));
        printf("\n # liczba prob= %d , przyblizenie pi = %g \n", niter * numprocs, pi);
    }
    else {
        printf("Procesor %d przesyła rezultat= %d do glownego procesu\n", myid, mycount);
        MPI_Send(&mycount, 1, MPI_REAL, master, tag, MPI_COMM_WORLD);
    }

    MPI_Finalize();
    return 0;
}

```

```

G:\PWIR\PWIR07\proj\ConsoleApplication1\x64\Debug>mpiexec -n 4 ConsoleApplication1.exe 100 100
Procesor 1 przesyła rezultat= 77 do glownego procesu
Procesor 3 przesyła rezultat= 77 do glownego procesu
Procesor 2 przesyła rezultat= 77 do glownego procesu

# liczba prob= 400 , przyblizenie pi = 3.08

```

Program oblicza przybliżoną liczbę PI stosując metodę Monte Carlo. Każdy wątek oblicza liczbę "trafień" (≤ 1), a następnie po wykonaniu obliczeń wysyła liczbę "trafionych" do procesu głównego (master), który je sumuje, a następnie mnoży wynik przez 4.