**Sprawozdanie**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Państwowa Wyższa Szkoła Zawodowa w Nysie | | | Wydział Nauk Technicznych | |
| Laboratorium Równoległego i Rozproszonego Przetwarzania | | | | |
| Kierunek: | Informatyka | Rok studiów nr: | 3 | Semestr nr: | 6 |
| Rok akademicki: | 2021/2022 | Grupa administracyjna: | L1 | Konto  na klastrze: | PR1g2 |

SPRAWOZDANIE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr ćwiczenia | Temat ćwiczenia | | | |
| 4 | Opracowanie i uruchomienie zrównoleglonego programu do całkowania funkcji (G) a - L1 | | | |
| Wpisz termin  złożenia  sprawozdania |
| 28.06.2022 |
| Data faktycznego  złożenia  sprawozdania |
| (nie wypełniaj) |
| Wykonawcy | Nazwisko | Imię | Nr indeksu | Ocena |
| Kusy | Oktawian | 25028 | (Nie wypełniane w trybie online) |
| Mikuła | Dominik | 25032 | (Nie wypełniane w trybie online) |
|  |  |  |  |

**Uwaga**: Umieszczenie danych osobowych wykonawców stanowi grupowe i nieodwołalne oświadczenie, że są oni/one   
(i tylko oni/one) współautorami przedstawionego sprawozdania. Późniejsza zmiana składu zespołu wykonawców nie będzie możliwa.

Nie wypełniać  
 przy składaniu online

Data i podpis prowadzącego ćwiczenia

**Wymagania typograficzne**

* Tekst główny sprawozdania należy składać czcionką normalną typu Times 12 pkt.
* Nie jest akceptowane wklejanie obrazków tabulogramów.
* Zawartość plików, nazwy ścieżek w systemie plików, polecenia wydawane z konsoli i uzyskiwane odpowiedzi systemu/aplikacji oraz kopie tabulogramów interakcji z powłoką należy składać czcionką normalną typu   
  **Courier 11 pkt**. Należy zachować wygląd, w tym pozycjonowanie tekstu.
* Nazwy pozycji menu w programach i nazwy przycisków ekranowych należy składać czcionką pogrubioną typu **Arial 11 pkt**.
* Wykluczone jest zamieszczanie ilustracji graficznych, o ile nie jest to wyraźnie wymagane w instrukcji. Tekst powinien z tłem wyraźnie kontrastować.

1. Temat ćwiczenia   
   Opracowanie i uruchomienie zrównoleglonego programu do całkowania funkcji (G) a - L1
2. Zakres ćwiczenia   
   Przedmiotem ćwiczenia jest przekształcenie programu do obliczania metodą Monte Carlo całki z funkcji do postaci wykonywalnej w trybie rozproszonym i jego uruchomienie w środowisku PBS Pro.

Co do wymagań ćwiczenia:

1. Przekształcenie wcześniej opracowanego programu sekwencyjnego do obliczenia całki metodą Monte Carlo w program zrównoleglony z wykorzystaniem biblioteki Open MPI,
2. Liczba dostępnych procesów ma być odczytana przez program za pomocą wywołania odpowiedniej funkcji MPI.
3. Liczba eksperymentów losowych do wykonania zawarta jest w parametrze wywołania programu – jak dla wersji sekwencyjnej.
4. Metoda zrównoleglenia: liczba losowań powinna być równomiernie rozłożona pomiędzy wszystkie procesy. W przypadku niezerowej reszty z dzielenia liczby losowań przez liczbę procesów, pozostałą część losowań (resztę z dzielenia) ma wykonać proces o najwyższym identyfikatorze. Do połączenia wyników cząstkowych wykorzystać funkcję MPI\_Reduce.
5. Proces master ma określić łączną liczbę trafień na podstawie danych przekazanych przez poszczególne procesy, obliczyć oszacowaną wartość całki i zredagować wyniki.
6. Wyniki mają być wyświetlone.
7. Środowisko realizacji ćwiczenia   
    CLion,

Klaster LRRP i środowisko PBS Pro

1. Przebieg ćwiczenia i uzyskane wyniki
   1. **Zadanie nr 1**
      1. Przekształć sprawdzony program sekwencyjny w program zrównoleglony z wykorzystaniem biblioteki Open MPI
      2. Przekształcenie programu sekwencyjnego w program zrównoleglony zgodnie z wymaganiami prowadzącego

// Autorzy: Oktawian Kusy; Dominik Mikula; @ Konto: PR1g2

// Cwiczenie: 4, Data opracowania: 2022-06-28, Wersja: 1

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <mpi.h>

#define srand48(x) srand((int)(x))

#define drand48() ((double)rand()/RAND\_MAX)

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

{

int rank, size, liczba\_losowan;

int wynik = 0;

int master = 0;

MPI\_Comm comm;

comm = MPI\_COMM\_WORLD;

MPI\_Init(&argc, &argv); //inicjalizacja MPI

MPI\_Comm\_rank(comm, &rank);

MPI\_Comm\_size(comm, &size);

double x, y, calka, t\_start, t\_end;

int licznik\_trafien = 0;

if (argv[1] != "") { liczba\_losowan = atoi(argv[1]); }

else { liczba\_losowan = 1000; }

t\_start = MPI\_Wtime();

srand48(t\_start);

for (int i = 0; i < liczba\_losowan / size; i++) {

x = drand48() \* 2;

y = drand48();

if (y < sqrt(x\*( 2 - x))) licznik\_trafien++;

}

if (rank == size -1) {

for (int i = 0; i < liczba\_losowan % size; i++) {

x = drand48() \* 2;

y = drand48();

if (y < sqrt(x\*(2 - x))) licznik\_trafien++;

}

}

MPI\_Reduce(&licznik\_trafien, &wynik, 1, MPI\_INT, MPI\_SUM, master, comm); //uzycie MPI\_Reduce wymusza obliczenia na procesie glownym

if (rank == 0) {

calka = wynik / (double) liczba\_losowan;

calka \*= 2;

t\_end = MPI\_Wtime() - t\_start;

printf("\n");

printf("Obliczanie oszacowania wartosci calki oznaczonej \n");

printf("z funkcji y=sqrt(x(2-x)) metoda Monte Carlo. \n");

printf("Wersja rownolegla \n");

printf("----------------------------------------------------------. \n");

printf("Autorzy: Oktawian Kusy, Dominik Mikula \n");

printf("Grupa: PR1g2 \n");

printf("----------------------------------------------------------. \n");

printf("Liczba procesow: %d\n", size); //liczba calkowita

printf("Liczba losowan: %d\n", liczba\_losowan); //liczba calkowita

printf("Oszacowanie wartosci calki: %.4f\n", calka); //liczba zmiennoprzecinkowa, 4 cyfry po przecinku

printf("Czas obliczen [s]: %.6f \n", t\_end); //liczba zmiennoprzecinkowa, 6 cyfr po przecinku

printf("\n");

}

MPI\_Finalize();

* 1. **Zadanie nr 2**
     1. Cel czynności: Kompilacja i uruchomienie zrównoleglonego programu(mpicc)

[PR1g2@p205 ~]$ mpicc 4\_z1KusyMikula.c -std=c99 -lm -o 4\_z1KusyMikula.c

[PR1g2@p205 ~]$ mpirun np 1 -np 1 4\_z1KusyMikula 1000

Obliczanie oszacowania wartosci calki oznaczonej

z funkcji y=sqrt(x(2-x)) metoda Monte Carlo.

Wersja rownolegla

----------------------------------------------------------.

Autorzy: Oktawian Kusy, Dominik Mikula

Grupa: PR1g2

----------------------------------------------------------.

Liczba procesow: 1

Liczba losowan: 1000

Oszacowanie wartosci calki: 1.5460

Czas obliczen [s]: 0.000064

* 1. **Zadanie nr 3**
     1. Cel czynności: Rozesłanie programu wykonywalnego do wszystkich pozostałych maszyn klastra. W tym celu należało użyć polecenia distribute-keys6 na maszynie głównej.

[PR1g2@p205 ~]$ distribute-file6 4\_z1KusyMikula

4\_z1KusyMikula 0% 0 0.0KB/s --:-- ETA

4\_z1KusyMikula 100% 13KB 874.7KB/s 00:00

4\_z1KusyMikula 0% 0 0.0KB/s --:-- ETA

4\_z1KusyMikula 100% 13KB 560.2KB/s 00:00

4\_z1KusyMikula 0% 0 0.0KB/s --:-- ETA

4\_z1KusyMikula 100% 13KB 764.4KB/s 00:00

4\_z1KusyMikula 0% 0 0.0KB/s --:-- ETA

4\_z1KusyMikula 100% 13KB 441.8KB/s 00:00

4\_z1KusyMikula 0% 0 0.0KB/s --:-- ETA4\_z1KusyMikula 100% 13KB 630.9KB/s 00:00

* 1. **Zadanie nr 4**
     1. Cel czynności: Przygotowanie zadania dla systemu PBS Pro, które zleci wykonanie obliczeń

Plik calka.pbs

#!/bin/bash

#PBS -N calka

#PBS -j oe

#PBS -l select=4:ncpus=8

mpirun -n 3 --host p200:8,p201:8,p202:8,p203:8,p204:8 4\_z1KusyMikula 1000

Uruchomienie:

[PR1g2@p205 ~]$ qsub calka.pbs

2197.p205

* 1. **Zadanie nr 5**
     1. Cel czynności: Sporządzenie tabeli pt. *Zarejestrowane czasy obliczeń*

|  |  |  |  |
| --- | --- | --- | --- |
| Liczba losowań ►  ▼Liczba procesów | 1000 | 1 000 000 | 1 000 000 000 |
| 3 | 0.000059 | 0.017117 | 16.817826 |
| 7 | 0.000099 | 0.013238 | 7.609719 |
| 19 | 0.004203 | 0.007542 | 5.705537 |

1. Wnioski z przeprowadzonych prac

Ćwiczenie pozwoliło przygotować program równoległy działający na klastrze systemu kolejkowego PBS Pro. Wszystkie czynności przewidziane w ćwiczeniu zostały wykonane poprawnie. Dla próby 1000 losowań użycie większej liczby procesów powoduje zwiększony czas w porównaniu do 3 losowań ze względu na wprowadzenie elementu opóźnień w komunikacji pomiędzy procesami.

1. Inne uwagi

Czasami maszyna p205 odmawiała współpracy i zwracała różne błędy. Poniżej niektóre z nich zarejestrowane w kronice.

[1656440439.415665] [p205:30780:0] sys.c:618 UCX ERROR shmget(size=2097152 flags=0xfb0) for mm\_recv\_desc failed: Operation not permitted, please check shared memory limits by 'ipcs -l'

[1656440439.416686] [p205:30779:0] sys.c:618 UCX ERROR shmget(size=2097152 flags=0xfb0) for mm\_recv\_desc failed: Operation not permitted, please check shared memory limits by 'ipcs -l'

[1656440439.425652] [p205:30778:0] sys.c:618 UCX ERROR shmget(size=2097152 flags=0xfb0) for mm\_recv\_desc failed: Operation not permitted, please check shared memory limits by 'ipcs -l'