**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 | | | |
| 5 | Pomiar czasu obrotu komunikatów MPI | | | |
| Wpisz termin  złożenia  sprawozdania |
| 30.06.2022 |
| 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

Pomiar czasu obrotu komunikatów MPI

1. Zakres ćwiczenia

Przedmiotem ćwiczenia jest:

a) opracowanie programu do obliczeń rozproszonych, którego zadaniem jest określenie czasu potrzebnego do przesłania krótkiego komunikatu od procesu-nadawcy do procesu-odbiorcy i odesłanie takiego samego komunikatu przez odbiorcę do nadawcy,

b) kompilacja i sprawdzenie działania oraz uruchomienie opracowanego programu w konfigura-cji rozproszonej,

c) wykonanie obliczeń w trybie rozproszonym ustalając konfigurację PBS Pro tak, aby procesy MASTER i SLAVE działały w odrębnych maszynach.

d) zestawienie w tabeli średnich czasów obiegu komunikatów dla obydwu trybów uruchomienia i różnych wartości liczby cykli,

e) zinterpretowanie wyników porównania czasów z tabeli.

1. Środowisko realizacji ćwiczenia

* CLion,
* Klaster LRRP i środowisko PBS Pro

1. Przebieg ćwiczenia i uzyskane wyniki  
   1. **Zadanie nr 1**
      1. Cel czynności – opracowanie w języku C programu do pomiaru czasu obrotu komunikatów MPI

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

// Cwiczenie 5, Data opracowania 2022-06-29, Wersja 1

#include <stdio.h>

#include <stdlib.h>

#include <mpi.h>

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

{

double t\_start, t\_end;

int liczba\_komunikatow, ilosc\_procesow, id;

double t3 = 0;

int ilosc\_raportowanych = 10;

int MASTER = 0, SLAVE = 1;

char komunikat = 255; // wartość komunikatu - 0xFF

if (argc != 2 && argc != 3) //sprawdzenie ilosci parametrow

{

printf("Zla liczba parametrow: nazwa ilosc\_komunikatow ilosc\_raportowanych\n");

return 21;

}

liczba\_komunikatow = atoi(argv[1]);

if (liczba\_komunikatow < 1)

{

printf("Ujemna liczba komunikatow!\n");

return 22;

}

if (argc == 3) ilosc\_raportowanych = atoi(argv[2]);

MPI\_Init(&argc, &argv);

MPI\_Comm\_size(MPI\_COMM\_WORLD, &ilosc\_procesow);

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

if (ilosc\_procesow > 2) printf("Za dużo procesow!\n\n");

if (ilosc\_procesow < 2) MPI\_Abort(MPI\_COMM\_WORLD, 20);

if (id == MASTER)

{

for (int i = 0; i < liczba\_komunikatow; i++)

{

t\_start = MPI\_Wtime();

if (MPI\_Send(&komunikat, 1, MPI\_BYTE, 1, 0, MPI\_COMM\_WORLD) != MPI\_SUCCESS)

{

printf("ERROR 00");

MPI\_Abort(MPI\_COMM\_WORLD, 0);

}

if (MPI\_Recv(&komunikat, 1, MPI\_BYTE, 1, 0, MPI\_COMM\_WORLD, MPI\_STATUS\_IGNORE) != MPI\_SUCCESS)

{

printf("ERROR 01");

MPI\_Abort(MPI\_COMM\_WORLD, 1);

}

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

t\_end \*= 1000000000; //zamiana na nanosekundy

if (i < ilosc\_raportowanych) printf("RAPORT: cykl %10d, czas trwania %.4f ns.\n", i, t\_end);

t3 += t\_end;

}

printf("\n");

printf("Sredni czas komunikatow: %f ns,\nSredni czas przeslania w jedna strone: %f ns\n", t3 / liczba\_komunikatow, t3 / (2 \* liczba\_komunikatow));

}

if (id == SLAVE)

{

for (int i = 0; i < liczba\_komunikatow; i++)

{

if (MPI\_Recv(&komunikat, 1, MPI\_BYTE, 0, 0, MPI\_COMM\_WORLD, MPI\_STATUS\_IGNORE) != MPI\_SUCCESS)

{

printf("ERROR 10");

MPI\_Abort(MPI\_COMM\_WORLD, 10);

}

if (MPI\_Send(&komunikat, 1, MPI\_BYTE, 0, 0, MPI\_COMM\_WORLD) != MPI\_SUCCESS)

{

printf("ERROR 11");

MPI\_Abort(MPI\_COMM\_WORLD, 11);

}

}

}

MPI\_Finalize();

return 0;

}

* 1. **Zadanie nr 2**
     1. Cel czynności – kompilacja programu oraz jego dystrybucja do pozostałych maszyn klastra

[PR1g2@p200 ~]$ mpicc 5\_z1\_KusyMikula.c -o 5\_z1\_KusyMikula.c -lm -std=c99

[PR1g2@p200 ~]$ distribute-file6 5\_z1\_KusyMikula

5\_z1\_KusyMikula 0% 0 0.0KB/s --:-- ETA

5\_z1\_KusyMikula 100% 13KB 593.2KB/s 00:00

5\_z1\_KusyMikula 0% 0 0.0KB/s --:-- ETA

5\_z1\_KusyMikula 100% 13KB 735.1KB/s 00:00

5\_z1\_KusyMikula 0% 0 0.0KB/s --:-- ETA

5\_z1\_KusyMikula 100% 13KB 639.7KB/s 00:00

5\_z1\_KusyMikula 0% 0 0.0KB/s --:-- ETA

5\_z1\_KusyMikula 100% 13KB 613.1KB/s 00:00

5\_z1\_KusyMikula 0% 0 0.0KB/s --:-- ETA5\_z1\_KusyMikula 100% 13KB 834.3KB/s 00:00

* 1. **Zadanie nr 3**
     1. Cel czynności – sprawdzenie poprawności działania programu

[PR1g2@p200 ~]$ mpirun -np 2 5\_z1\_KusyMikula 1 1000

RAPORT: cykl 0, czas trwania 251913.4432 ns.

Sredni czas komunikatow: 251913.443208 ns,

Sredni czas przeslania w jedna strone: 125956.721604 ns

[PR1g2@p200 ~]$ mpirun -np 2 5\_z1\_KusyMikula 1 1000

RAPORT: cykl 0, czas trwania 184103.8465 ns.

Sredni czas komunikatow: 184103.846550 ns,

Sredni czas przeslania w jedna strone: 92051.923275 ns

* 1. **Zadanie nr 4**
     1. Cel czynności – utworzenie skryptu programu PBS pro i próba jego uruchomienia

#!/bin/bash

#PBS -N 5\_z1\_KusyMikula

#PBS -j oe

#PBS -l nodes=2:ppn=1

mpirun -np 2 --host p200:2, p201:2,p203:2,p204:2,p205:2 5\_z1\_KusyMikula 1000

Niestety podczas próby uruchomienia skryptu poleceniem qsub maszyna odmówiła współpracy, zwracając nieznany błąd. W związku z tym, program został uruchomiony bez niego.

[PR1g2@p200 ~]$ qsub 5\_KusyMikula.pbs

qsub: Bad UID for job execution

* 1. **Zadanie nr 5**
     1. Cel czynności – wykonanie programu dla liczby cykli równej 1000, 1 000 000 i 1 000 000 000

[PR1g2@p200 ~]$ mpirun -np 2 --hostfile hostfile 5\_z1\_KusyMikula 1 1000

Sredni czas komunikatow: 190464.779735 ns,

Sredni czas przeslania w jedna strone: 95232.389867 ns

[PR1g2@p200 ~]$ mpirun -np 2 --hostfile hostfile 5\_z1\_KusyMikula 1 1000000

Sredni czas komunikatow: 124728.307128 ns,

Sredni czas przeslania w jedna strone: 62364.153564 ns

[PR1g2@p200 ~]$ mpirun -np 2 --hostfile hostfile 5\_z1\_KusyMikula 1 1000000000

Sredni czas komunikatow: 185428.187251 ns,

Sredni czas przeslania w jedna strone: 92714.093626 ns

Ponowna próba w celu sprawdzenia poprawności wyników:

[PR1g2@p200 ~]$ mpirun -np 2 --hostfile hostfile 5\_z1\_KusyMikula 1 1000

Sredni czas komunikatow: 240761.786699 ns,

Sredni czas przeslania w jedna strone: 120380.893350 ns

[PR1g2@p200 ~]$ mpirun -np 2 --hostfile hostfile 5\_z1\_KusyMikula 1 1000000

Sredni czas komunikatow: 186787.918210 ns,

Sredni czas przeslania w jedna strone: 93393.959105 ns

[PR1g2@p200 ~]$ mpirun -np 2 --hostfile hostfile 5\_z1\_KusyMikula 1 1000000000

Sredni czas komunikatow: 257275.998592 ns,

Sredni czas przeslania w jedna strone: 128637.999296 ns

* 1. **Zadanie nr 6**
     1. Cel czynności – sporządzenie tabeli średnich czasów obiegu komunikatów dla obydwu trybów uruchomienia i różnych wartości liczby cykli oraz ich interpretacja

|  |  |  |  |
| --- | --- | --- | --- |
| Liczba komunikatów | 1 000 | 1 000 000 | 1 000 000 000 |
| Średni czas komunikatów [ns] | 19 0464.779735 | 12 4728.307128 | 18 5428.187251 |
| Średni czas przesłania w jedną stronę [ns] | 9 5232.389867 | 6 2364.153564 | 9 2714.093626 |

Czas przesyłania komunikatów był najniższy dla wartości 1 000 000, dla wartości 1 000 000 000 czas przesyłania był niż niższy niż dla wartości 1 000.

1. Wnioski z przeprowadzonych prac

Pomimo licznych prób uruchomienia programu z jednakowymi parametrami, wyniki nigdy nie były identyczne. Wynika to z faktu zastosowania czasu pomiaru liczonego w nanosekundach. Największe rozbieżności czasu można zauważyć przy niskiej liczbie komunikatów, ze względu na duże prawdopodobieństwo największej losowości dla pierwszego komunikatu.

Czas komunikatów jest wyraźnie krótszy dla 1 000 000 komunikatów, jednak znów zaczyna się wydłużać przy wzroście liczby komunikatów aż do 1 000 000 000, jednak wciąż jest on jednak krótszy od niskiej liczby wymiany komunikatów.

1. Inne uwagi

Brak.