ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ   
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ И ИНФОРМАТИКИ»

**КУРСОВОЙ ПРОЕКТ**

по дисциплине “Параллельные вычислительные технологии”

на тему

**Алгоритм биномиального дерева для корневой редукции**

|  |  |
| --- | --- |
| Выполнил студент |  |
|  | Ф.И.О. |

|  |  |
| --- | --- |
| Группы |  |
|  |  |

|  |  |  |
| --- | --- | --- |
| Работу принял |  | профессор д.т.н. М.Г. Курносов |
|  | подпись |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Защищена |  | Оценка |  |
|  |  |  |  |

Новосибирск – 2020

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 4](#__RefHeading___Toc337081083)

[1 Язык IBM X10 5](#__RefHeading___Toc337081084)

[1.1 Трансляционно-циклический обмен информацией 5](#__RefHeading___Toc337081085)

[ЗАКЛЮЧЕНИЕ 7](#__RefHeading___Toc337081086)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 8](#__RefHeading___Toc337081087)

[ПРИЛОЖЕНИЕ 9](#__RefHeading___Toc337081088)

[1 Исходный код примеров 10](#__RefHeading___Toc337081089)

ВВЕДЕНИЕ

Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст.

1 Язык IBM X10

* 1. Трансляционно-циклический обмен информацией

Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст. Текст.

Формулы следует набирать в Microsoft Equation, например так . Если на формулы есть ссылка в тексте, то формула набирается по центру.

|  |  |
| --- | --- |
|  | (1.1) |

Рисунки по центру.



Рисунок 1.1 – Пример иерархической организации  
коммуникационной среды кластерной ВС

На рис. 1.1 приведен пример иерархической организации коммуникационной среды кластера.



Рисунок 1.2 – Пример иерархической организции  
коммуникационной среды вычислительного кластера:  
три вычислительных узла на базе 2 x AMD Opteron 275;  
*N* = 12; *L* = 3; *n*23 = 2; *C*23 = {9, 10, 11, 12}; *с*23 = 4; *g*(3, 3, 4) = 2; *z*(1, 7) = 1

На рис. 1.2 приведен комплексный пример.

Таблицы оформлять так.

Таблица 5.1 – Результаты экспериментов

|  |  |  |
| --- | --- | --- |
| Граф | Время работы пакета Maple 11, с | Время работа созданного алгоритма, с |
| *C*1 | 2,8 | < 0,01 |
| *C*2 | 14,3 | < 0,01 |
| *C*3 | 32,4 | 0,01 |
| *G*1 | 3,0 | 0,02 |
| *G*2 | 1123,0 | 0,50 |
| *G*3 | > 24 часов | 507,2 |

ЗАКЛЮЧЕНИЕ

В результате выполнения работы разработан и исследован алгоритм …

Осуществлено моделирование разработанного алгоритма. Показано, что …

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Хорошевский В.Г. Архитектура вычислительных систем. – М.: МГТУ им. Н.Э. Баумана,  
   2008. – 520 с.
2. Евреинов Э.В., Хорошевский В.Г. Однородные вычислительные системы. – Новосибирск: Наука, 1978. – 320 с.
3. Rabenseifner R.. Automatic MPI Counter Profiling // Proceedings of the 42nd Cray User Group. – Noorwijk, The Netherlands, 2000. – 19 pp.
4. Han D., Jones T.. MPI Profiling // Technical Report UCRL-MI-209658 – Lawrence Livermore National Laboratory, USA, 2004. – 15 pp.
5. Thakur R., Rabenseifner R., and Gropp W. Optimization of collective communication operations in MPICH // Int. Journal of High Performance Computing Applications. – 2005. – Vol. 19, No. 1. – P. 49‑66.
6. Balaji P., Buntinas D., Goodell D., Gropp W., Kumar S., Lusk E., Thakur R. and Traff J. L. MPI on a Million Processors // Proc. of the PVM/MPI – Berlin: Springer-Verlag, 2009. – P. 20‑30.
7. Khoroshevsky V., Kurnosov M. Mapping Parallel Programs into Hierarchical Distributed Computer Systems // Proc. of “Software and Data Technologies”.  Sofia: INSTICC, 2009.  Vol. 2.  P. 123‑128.

ПРИЛОЖЕНИЕ

1 Исходный код примеров

/\*

\* pi\_mpi.c: MPI simple integration example.

\*

\*/

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include "mpi.h"

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

{

double PI25DT = 3.141592653589793238462643;

int i, rank, commsize;

double nsteps, step, local\_pi, pi, sum, x;

double time = 0.0;

int namelen;

char processor\_name[MPI\_MAX\_PROCESSOR\_NAME];

MPI\_Init(&argc,&argv);

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

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

MPI\_Get\_processor\_name(processor\_name, &namelen);

fprintf(stdout,"Process %d of %d is on %s\n",

rank, commsize, processor\_name);

fflush(stdout);

nsteps = (argc > 1) ? atoi(argv[1]) : 100000;

if (rank == 0)

time -= MPI\_Wtime();

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

step = 1.0 / (double)nsteps;

sum = 0.0;

for (i = rank + 1; i <= nsteps; i += commsize)

{

x = ((double)i - 0.5) \* step;

sum += 4.0 / (1.0 + x \* x);

}

local\_pi = nsteps \* sum;

MPI\_Reduce(&local\_pi, &pi, 1, MPI\_DOUBLE, MPI\_SUM, 0,

MPI\_COMM\_WORLD);

if (rank == 0) {

time += MPI\_Wtime();

printf("PI is approximately %.16f, Error is %.16f\n",

pi, fabs(pi - PI25DT));

printf("(nsteps = %d, step = %f)\n", nsteps, step);

printf("Elapsed time = %.4f sec.\n", time);

fflush(stdout);

}

MPI\_Finalize();

return 0;

}