

УДК 004.021
МРНТИ 50.41.17

DOI: <https://doi.org/10.37788/2022-1/132-138>

Н.М. Зайцева

Инновационный Евразийский университет, Казахстан
(e-mail: zaitzevns@mail.ru)

Организация виртуального кластера для параллельных вычислений

Аннотация

Основная проблема: Организация многопроцессорных вычислений для освоения материала дисциплины «Архитектура систем параллельных вычислений» для студентов образовательных программ «Информационные технологии» в условиях дистанционного вида обучения, когда обучающиеся имеют в своем распоряжении всего один компьютер.

Цель: Разработать методику создания среды программирования, способной выполнять параллельные алгоритмы для многопроцессорных ЭВМ на одном персональном компьютере студента, то есть создать эмуляцию многопроцессорного вычисления, благодаря которой студент сможет на практике освоить работу параллельных алгоритмов и оценить скорость параллельных вычислений посредством их сравнений с последовательным вариантом решения этой же задачи.

Методы: Для реализации поставленной цели предложено использовать свободно распространяемую среду MPICH, с помощью которой можно создать кластер. Создание кластера должно выполняться путем установки MPICH на компьютер и настройки программного обеспечения, способного воспринимать процедуры передачи сообщений между работающими процессами. Для установки и настройки среды программирования предложена среда Visual Studio, в частности, язык C++. Для многопроцессорных вычислений необходима разработка консольного приложения, так как ввод и вывод данных при подобных вычислениях производится только на головном процессоре в независимости от возможных событий на всех остальных процессах, поэтому устанавливается режим разработки консольного приложения.

Вычислительные процессы в ходе своей работы на кластере должны обмениваться информацией в виде сообщений. Предлагаемые учебные примеры позволяют наглядно убедиться, каким образом происходит этот обмен. Кроме того, благодаря такому эмулятору, имеется возможность убедиться, как возрастает скорость расчета и точность вычислений при увеличении количества запускаемых на кластере процессов.

Результаты и их значимость: Предложен метод, позволяющий создать эмуляцию многопроцессорного вычисления на воображаемом кластере. В этом режиме имеется возможность проверки всех способов пересылки сообщений между выполняющимися процессами. Кроме того, имеется возможность вывести на экран затрачиваемое для вычислений время, что позволяет оценить разработанный алгоритм с точки зрения его эффективности. Таким образом, студенты, находясь на дистанционной форме обучения, могут в полном объеме освоить требуемый материал.

Ключевые слова: параллельные вычисления, кластер, программная среда, организация кластера, программирование, функционирование кластера.

Введение

Создание высокопроизводительных вычислительных систем в настоящий момент является одной из наиболее сложных научно-технических задач современности [1, 2, 3]. Это связано с жизненной необходимостью решения задач, которые ранее просто не решались. К таким задачам относятся прогнозирование развития ураганов, моделирование процессов ядерных реакций, создание моделей управления полетами космических ракет, самолетов, беспилотных автомобилей и многое другое. Одним из направлений повышения производительности вычислительных систем стало распараллеливание процесса вычисления. Параллельные вычислительные алгоритмы применялись уже с середины 70-х годов прошлого столетия в архитектуре больших вычислительных машин, называемых сейчас мэйнфреймами. С помощью параллельных архитектур увеличивалось быстродействие ЭВМ. Инженерами в области вычислительной техники разработаны различные архитектуры, которые реализуют принципы параллельной обработки данных в вычислительных машинах. Эти и векторно-конвейерные, и массивно-параллельные, и с широким командным словом, и гиперкубы, и спецпроцессоры, и мультипроцессоры, и кластеры и многие другие принципы параллельной обработки информации [1, 2].

С учётом требований времени в перечень дисциплин образовательных программ по информационным технологиям была введена дисциплина, связанная с освоением знаний в области параллельных вычислений как на одной ЭВМ, так и на нескольких, организованных с помощью информационных сетей в единый вычислительный узел, называемый кластером. Для задач дисциплины

параллельных вычислений нами было использовано создание кластера на основе однотипных ПЭВМ. Для этого в нашем вузе был организован такой кластер, состоящий из 8-ми ПЭВМ, на котором студенты имели возможность организовывать вычисления с помощью установленной в этом компьютерном классе специальной вычислительной среды, работающей по принципу обмена сообщений между процессорами. Эта среда получила название MPI по первым буквам фразы «Message Passing Interface», означающей «Интерфейс передачи сообщений» [1, 2].

Чтобы обеспечить получение студентами качественного объема знаний по указанной дисциплине в условиях дистанционного обучения, был найден метод организации виртуального кластера, который мог быть организован на домашнем компьютере каждого студента. Для этого нами найдено было решение этой проблемы, и в данной статье предлагается два варианта действий. Они предлагаются в связи с тем, что у студентов могут быть различные компьютеры, а также установленные на эти компьютеры операционные системы разного вида.

Материалы и методы

Первый вариант решения проблемы организации многопроцессорных вычислений на отдельном компьютере с 64-битым процессором: необходима установка на персональный компьютер, подключенный к сети Интернет специальной коммуникационной среде для параллельных вычислений MPI (MPICH). Эта среда представляется разработчиками на бесплатной основе и ее скачивание можно выполнить по ссылке: <https://www.microsoft.com/en-us/download/details.aspx?id=100593>.

На рисунке 1 показан интерфейс для установки на компьютер среды параллельных вычислений MPI.

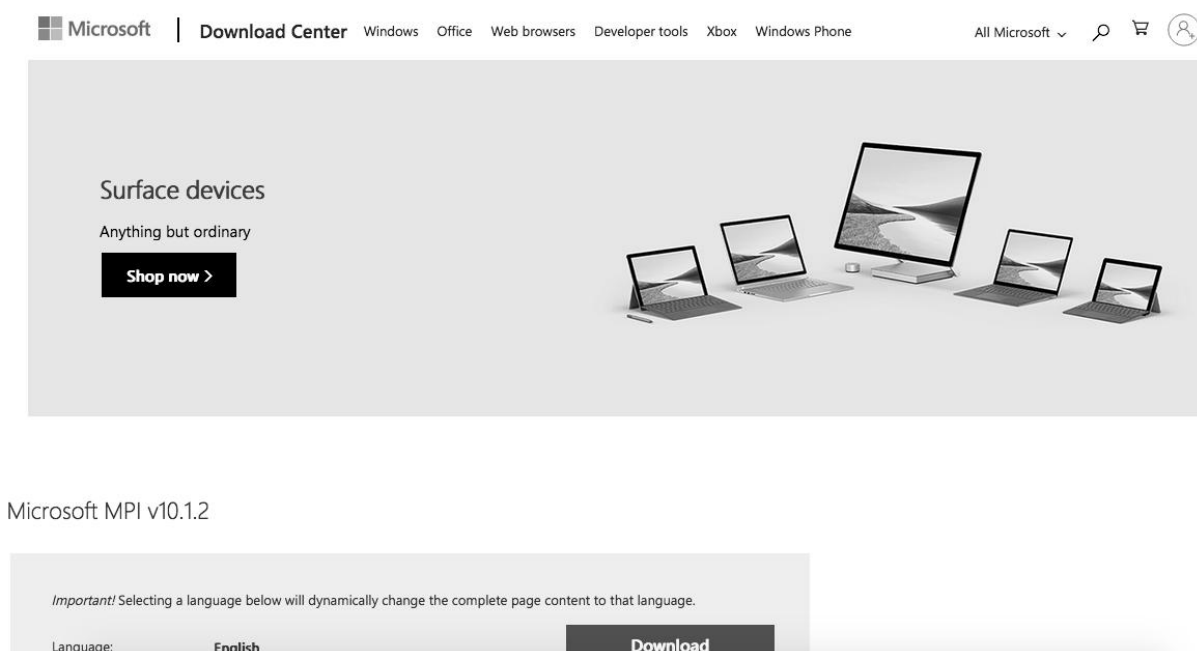


Рисунок 1 – Интерфейс официального сайта Microsoft для скачивания среды MPI для 64-х разрядных процессоров

Для разворачивания среды необходима установка файлов `msmpisetup.exe` и `msmpisdsk.msi`. При этом разворачиваемая среда подготавливается к многопроцессорной работе на кластере. По завершении этого процесса в каталоге Program Files устанавливаются программы, с помощью которых уже возможно включение данного компьютера в кластер.

Основной программой, организующей параллельные вычисления в этом пакете, является программа `mpiexec.exe` с параметром `-n` и количеством запускаемых процессов. Помимо этой, основной для параллельных вычислений программы, в пакете MPICH есть ряд тестовых программ, которые позволяют убедиться, что кластер настроен верно и работает как единая многопроцессорная система. Одной из таких тестовых программ является программа итерационного вычисления числа π . При запуске этой программы расчет π , например, на 5-ти процессорах выполняется точнее. Ошибка при расчетах составляет на 5-ти процессорах – 0,00333, а на двух процессорах – 0,02076. При расчете π на 10-ти процессорах ошибка еще меньше и составляет 0,000833. Сказанное иллюстрируется рисунком 2.

```

C:\Program Files (x86)\MPICH2\examples\cpi.exe
Enter the number of intervals: <@ quits> 2
pi is approximately 3.1623529411764704, Error is 0.0207602875866773
wall clock time = 0.000337
Enter the number of intervals: <@ quits> 5
pi is approximately 3.1449258640033277, Error is 0.0033332104135346
wall clock time = 0.000323
Enter the number of intervals: <@ quits> 10
pi is approximately 3.1424259850010987, Error is 0.0008333314113056
wall clock time = 0.000390
Enter the number of intervals: <@ quits>

```

Рисунок 2 – Результат работы тестового примера сначала на 2-х процессорах, затем на 5-ти и 10-ти процессорах

Рисунок прекрасно иллюстрирует увеличение точности вычислений числа π с увеличением количества процессоров: при двух процессорах ошибка вычислений – 0,02076, а при 10-ти процессорах – 0,00083

Следующий этап создания среды параллельных вычислений – этап установки и настройки специальным образом среды программирования Visual Studio, в частности, пакета разработки программ на языке C++. Дело в том, что для работы на кластере нужна разработка консольных приложений с подключением библиотек MPI. Ввод и вывод необходимых для вычислений данных при параллельных вычислениях на кластере производятся на головной ЭВМ. Поэтому необходимо соответствующим образом настроить среду программирования.

Это не очень простой процесс, поэтому настройку можно рекомендовать выполнить только для одного проекта, а потом этот проект использовать в качестве основы для разработки других приложений. Итак, нами была выполнена настройка по нижеприведенному алгоритму:

В окне «Свойства» проекта выбирается вкладка «Свойства конфигурации», затем – опция «Каталоги VC++», а затем - «Включаемые каталоги», а после этого вызван список «Изменить». Интерфейс предлагаемого алгоритма представлен на рисунке 3.

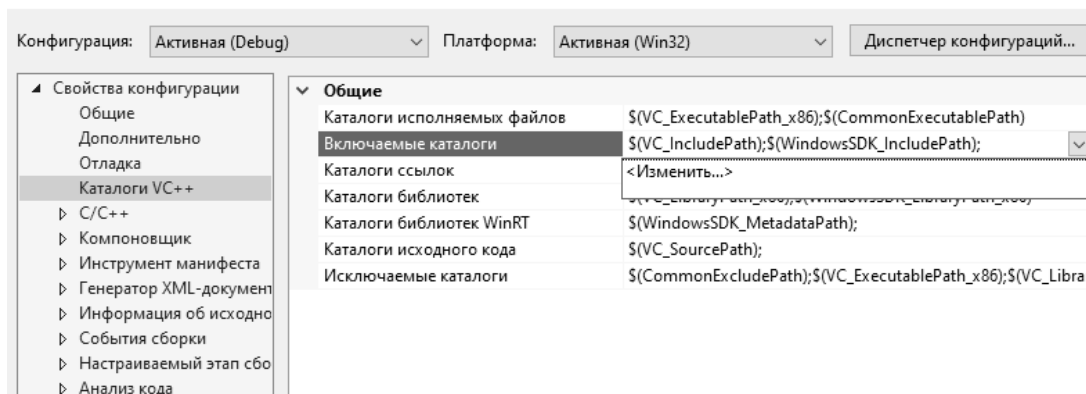


Рисунок 3 – Настройка «Включаемых каталогов» для организации консольного проекта многопроцессорных вычислений

В открывающемся окне была введена строка: $\$(MSMPI_INC);$(MSMPI_INC)\x86$

Следующим этапом были произведены настройки на вкладке «Компоновщик» -> «Дополнительные каталоги библиотек» -> вызвать список > «Изменить». Интерфейс этих настроек представлен на рисунке 4.

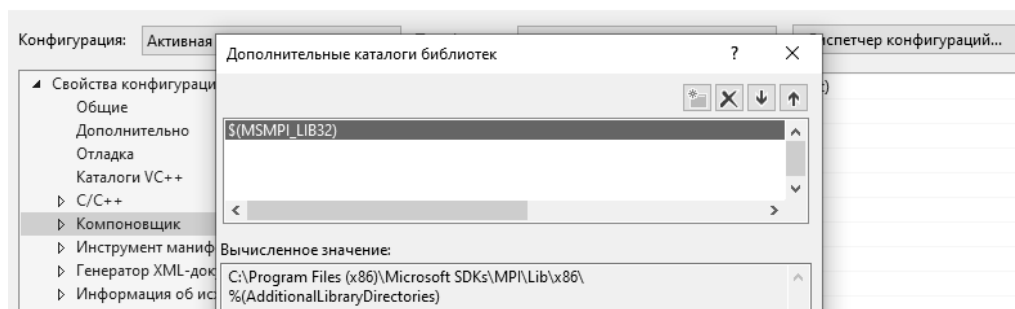


Рисунок 4 – Результат введенной строки на вкладке Компоновщик

В открывшейся строке была введена следующая информация: `$(MSMPI_LIB32)`.

На этом настройка среды программирования еще не заканчивается. На вкладке «Компоновщик» необходимо было выполнить следующие действия: выбрав опцию «Ввод», а затем в опции «Дополнительные зависимости» > в списке > «Изменить». Было установлено значение `msmpi.lib`.

Разрабатываемый после такой настройки проект устанавливается в x86, и затем может быть выполнена компиляция этого проекта. В процессе компиляции появляется уведомление о подключении, при этом необходимо выбрать «Allow access» (то есть разрешения доступа).

Второй вариант решения проблемы: организация многопроцессорных вычислений на отдельном компьютере с 32-битным процессором. Для разворачивания среды, способной работать на 32-х битном процессоре, необходима установка другой версии среды MPICH. При разворачивании кластера была использована ссылка, размещенная на официальном сайте:

<http://www-unix.mcs.anl.gov/mpi/mpich/downloads/mpich2-1.0.5p2-win32-ia32.msi>

Эта ссылка выводит на загрузку, интерфейс которой приведен на рисунке 5.

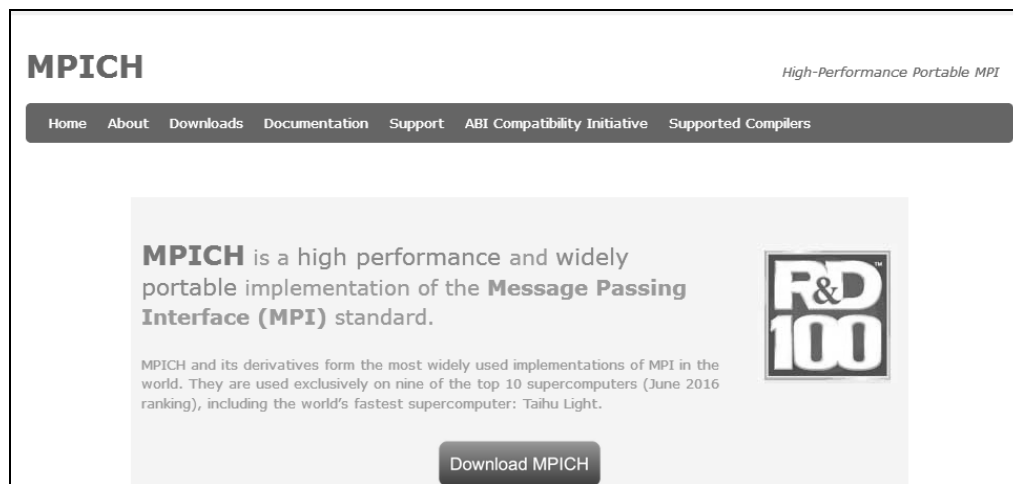


Рисунок 5 – Интерфейс загрузки среды параллельных вычислений на кластере MPICH.

После разворачивания MPICH выполняется настройка среды программирования C++. Она отличается от настройки для 64-х битных процессоров, ее алгоритм выглядит следующим образом. Создается консольное приложение. Во вкладке Проект и необходимо выбирать «Свойства», затем в открывшейся вкладке C/C++ и перейти на вкладку «Общее», а в пункте «Дополнительные каталоги включаемых файлов» вставить путь до файлов папки `mpich/include`. Сказанное иллюстрируется рисунком 6.

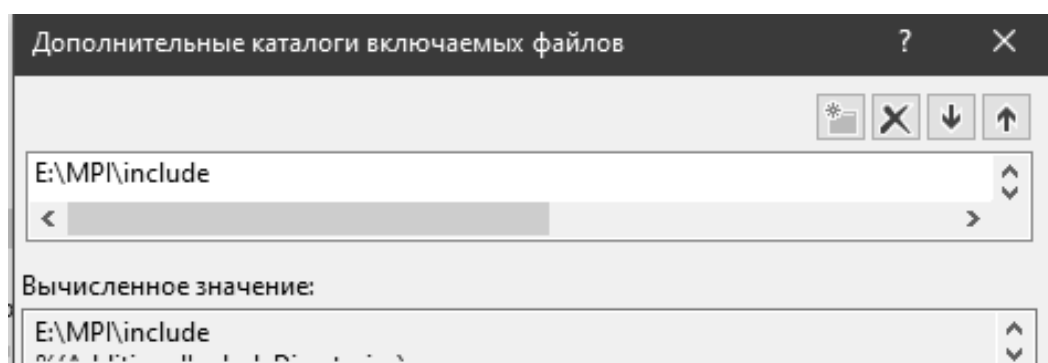


Рисунок 6 – Интерфейс настройки проекта для работы с MPI

В пункте «Предварительно откомпилированные заголовки» и в атрибуте «Предварительно откомпилированный заголовок» необходимо выбрать «Не использовать предварительно откомпилированные заголовки». Далее переходим в пункт «Дополнительно». Атрибуту «Компилировать как» необходимо задать значение «Компилировать как код C (/TC)». Во вкладке «Компоновщик» и требуется выбрать пункт «Общие», «Дополнительные каталоги библиотек», затем указать расположение каталога `Mpich2\lib`. Смотри рисунок 7.

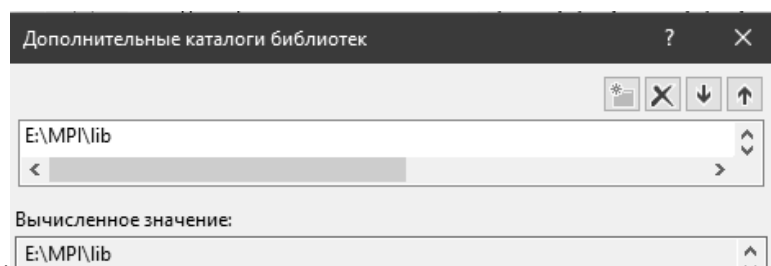


Рисунок 7 – Интерфейс следующей настройки проекта для работы с MPI

В пункте «Ввод» в атрибуте «Дополнительные зависимости» добавить файл `mpi.lib` из ранее добавленного каталога `Mpich2\lib`. На этом настройка среды программирования Visual Studio C++ заканчивается.

Результаты

Для проверки работоспособности параллельных вычислений на кластере была откомпилирована и запущена следующая тестовая программа:

```
#include <stdio.h>
#include "mpi.h"
int main(int argc, char* argv[])
{
    int rank, numprocs;
    // инициализация библиотеки MPI
    MPI_Init(&argc, &argv);
    // получение количества процессов в программе
    MPI_Comm_size(MPI_COMM_WORLD, &numprocs);
    // получение ранга процесса
    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    printf("\n Hello from process %3d", rank);
    // завершение работы библиотеки MPI
    MPI_Finalize();
    return 0;
}
```

На рисунке 8 изображено окно работы кластера, на котором запущено в работу 4 процесса, выполняющих приведенную выше программу. Каждый из процессов в этом примере, при запуске на своем процессоре выдает номер процесса на экран в виде сообщения «Hello from process N» (где N – номер процессора). Параллельный вычислительный процесс запускается из консоли с помощью команды:

`mpiexec -n 4 имя_программы`

Здесь «**имя_программы**» – полное имя консольного приложения.

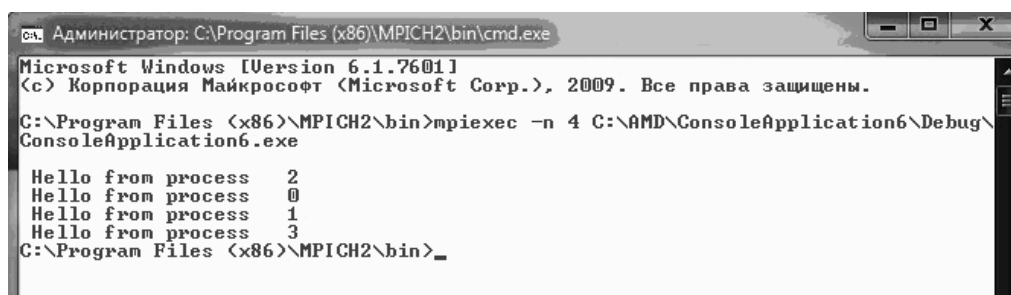


Рисунок 8 – Окно работы тестового примера на 4-х процессорах

Этот рисунок позволяет убедиться, что созданный кластер работает корректно. Каждый процесс выдает свое сообщение.

Обсуждение

Благодаря найденной методике студенты в период жесткого карантина смогли выполнить требуемые задания по дисциплине «Архитектура систем параллельных вычислений», а также выполнить исследования различных режимов работы кластера по пересылке информации между процессорами в ходе решения задач. Помимо этого, у них была возможность оценить скорость вычислительного алгоритма с применением распараллеливания и без этого механизма.

Заключение

1. Благодаря созданному таким образом кластеру каждому студенту IT образовательных программ нашего университета в условиях карантина удалось освоить параллельное программирование в полном объеме, несмотря на отсутствие возможности заниматься в лаборатории «Параллельных вычислений».

2. Используя предлагаемую технологию, любой программист может опробовать свой параллельный алгоритм на созданном таким образом эмуляторе кластере.

3. Разработанная в настроенной специальным образом среде программирования C++ программа может работать на любом реально существующем кластере без каких-либо изменений.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 Гергель В.П. Теория и практика параллельных вычислений: учеб. пос. – М: Интернет-университет информационных технологий; БИНОМ. Лаборатория знаний, 2017. – 424 с.

2 Хьюз К. Параллельное и распределенное программирование на C++. – М.: Издательский дом «Вильямс». – 2014. – 672с.

3 Антонов А.С. Технологии параллельного программирования MPI и OpenMP: учеб. пос. – М.: Изд-во МГУ, 2012. – 344с.

REFERENCES

1 Gergel, V.P. (2017). Teoriya i praktika parallel'nykh vychisleniy: uchebnoye posobiye [Theory and Practice of Parallel Computing: A Study Guide]. Moscow: Internet-universitet informacionnyh tekhnologii; BINOM. Laboratoriya znaniy [in Russian].

2 Hughes, K. (2014). Parallel'noye i raspredelennoye programmirovaniye na C++ [Parallel and Distributed Programming in C ++]. Moscow: Internet-universitet informacionnyh tekhnologii; BINOM. Laboratoriya znaniy [in Russian].

3 Antonov, A.S. (2012). Tekhnologii parallel'nogo programmirovaniya MPI i OpenMP [Examples from the textbook Parallel programming technologies MPI and OpenMP]. Moscow: Izd. MSU [in Russian].

Н.М. Зайцева

Инновациялық Еуразия университеті, Қазақстан

Параллель есептеулер үшін виртуалды кластерді ұйымдастыру

Студенттердің қолында бір ғана компьютер болған кезде қашықтықтан оқыту жағдайында «Ақпараттық технологиялар» білім беру бағдарламаларының студенттері үшін «Параллельді есептеу жүйелерінің архитектурасы» пәнінің материалын игеру үшін мультипроцессорлық есептеулерді ұйымдастыру.

Мақсаты – студенттің бір жеке компьютерінде көп процессорлы компьютерлер үшін параллель алгоритмдерді орындай алатын бағдарламалау ортасын құру әдістемесін жасау, яғни көп процессорлы есептеу эмуляциясын құру, соның арқасында студент параллель алгоритмдердің жұмысын іс жүзінде игере алады және параллель есептеулердің жылдамдығын сол мәселені шешудің дәйекті нұсқасымен салыстыру арқылы бағалай алады.

Қойылған мақсатты жүзеге асыру үшін кластер құруға болатын еркін таратылатын MPICH ортасын пайдалану ұсынылады. Кластер құру компьютерге MPICH орнату және жұмыс процестері арасында хабарлама жіберу процедураларын қабылдауға қабілетті бағдарламалық жасақтаманы орнату арқылы жүзеге асырылуы керек. Бағдарламалау ортасын орнату және конфигурациялау үшін Visual Studio ортасы, атап айтқанда c++тілі ұсынылады. Мультипроцессорлық есептеулер үшін консольдік қосымшаны әзірлеу қажет, өйткені мұндай есептеулерде деректерді енгізу және шығару барлық басқа процестердегі мүмкін оқиғаларға қарамастан тек бас процессорда жүзеге асырылады, сондықтан консольдік қосымшаны әзірлеу режимі орнатылады.

Кластердегі жұмыс барысында есептеу үрдістері хабарлама түрінде ақпарат алмасуы керек. Ұсынылған оқу мысалдары осы алмасудың қалай жүретініне көз жеткізуге мүмкіндік береді. Сонымен қатар, осы эмулятордың арқасында кластерде іске қосылған үрдістер санының артуымен есептеу жылдамдығы мен есептеу дәлдігі қалай өсетініне көз жеткізуге болады.

Қиял кластерінде мультипроцессорлық есептеуді эмуляциялау әдісі ұсынылады. Бұл режимде орындалатын процестер арасында хабарламаларды жіберудің барлық тәсілдерін тексеруге болады. Сонымен қатар, есептеулер үшін жұмсалған уақытты экранға шығаруға болады, бұл әзірленген алгоритмді оның тиімділігі тұрғысынан бағалауға мүмкіндік береді. Осылайша, студенттер қашықтықтан оқыту нысанында бола отырып, қажетті материалды толық көлемде меңгере алады.

Түйін сөздер: параллельді есептеу, кластер, бағдарламалық орта, кластерді ұйымдастыру, бағдарламалау, кластердің жұмысы.

N.M. Zaitseva

Innovative University of Eurasia, Kazakhstan

Organization of a virtual cluster for parallel computing

The organization of multiprocessor computing for mastering the material of the discipline «Architecture of parallel computing systems» for students of educational programs «Information Technology» in the conditions of distance learning, when students have only one computer at their disposal.

To develop a methodology for creating a programming environment capable of performing parallel algorithms for multiprocessor computers on one student's personal computer, that is, to actually create an emulation of multiprocessor computing and, thanks to such an environment, the student will be able to master the operation of parallel algorithms in practice and evaluate the speed of parallel computing by comparing them with a sequential solution of the same problem.

To achieve this goal, it is proposed to use the freely distributed MPICH environment, with which you can create a cluster. Cluster creation should be performed by installing MPICH on a computer and configuring software capable of perceiving message transmission procedures between running processes. To install and configure the programming environment, the Visual Studio environment is offered, in particular the C++ language. For multiprocessor computing, the development of a console application is necessary, since data input and output during such calculations is performed only on the head processor, regardless of possible events on all other processes, therefore, the console application development mode is set.

Computing processes in the course of their work on the cluster should exchange information in the form of messages. The proposed training examples allow you to clearly see how this exchange takes place. In addition, thanks to such an emulator, it is possible to verify how the calculation speed and accuracy of calculations increases with an increase in the number of processes running on the cluster.

A method is proposed that allows you to create an emulation of multiprocessor computing on an imaginary cluster. In this mode, it is possible to check all methods of forwarding messages between running processes. In addition, it is possible to display the time spent for calculations on the screen, which allows you to evaluate the developed algorithm in terms of its effectiveness. Thus, students, being on a distance learning form, can fully master the required material.

Keywords: parallel computing, cluster, software environment, cluster organization, programming, cluster operation.

Дата поступления рукописи в редакцию: 08.12.2022 г.