ИСПОЛЬЗОВАНИЕ ИНТЕРНЕТА ДЛЯ ОБУЧЕНИЯ ПАРАЛЛЕЛЬНОМУ ПРОГРАММИРОВАНИЮ

В.А. Бахтин, О.Ф. Жукова, Н.А. Катаев, А.С. Колганов, В.А. Крюков, Н.В. Поддерюгина, М.Н. Притула, О.А. Савицкая, А.А. Смирнов

Абрау-Дюрсо, 23 сентября 2015 г.

Доля интернет-пользователей среди молодежи (18-24 летних) составляет 96%, среди обучающихся в ВУЗах – 81 %.

Опрос проведен ВЦИОМ 21-22 марта 2015 года, опрошено 1600 человек в 130 населенных пунктах в 46 областях, краях и республиках. (Пресс-выпуск №2836 — URL: http://wciom.ru/index.php?id=236&uid=115255).

Проблемы и возможности

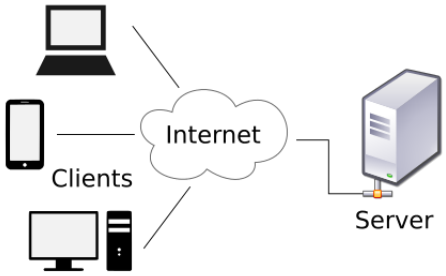
- ссылки на электронные ресурсы с проверенной информацией;
- ресурсы, описывающие задания и контролирующие их выполнение;
- укрепление связи студент-преподаватель вне семинарских занятий: электронная переписка, списки рассылки, связь в режиме реального времени.

Для обучения параллельному программированию необходим доступ к вычислительным ресурсам

суперкомпьютерные системы

Суперкомпьютерный комплекс «Ломоносов»
http://parallel.ru/cluster/superinfo

небольшие рабочие станции



Кафедра Системного программирования факультета Вычислительной математики и кибернетики Московского государственного университета им М.В. Ломоносова.

Институт Прикладной Математики им. М.В. Келдыша РАН.

Подготовка в области суперкомпьютерных технологий

- исследования в направлении разработки высокоуровневых языков параллельного программирования (система DVM и DVMH-языки) и создания систем автоматизированного распараллеливания (САПФОР);
- теоретическая подготовка и опыт разработки параллельных программ для современных вычислительных кластеров.

Полезные особенности

- управление несколькими проектами из одного приложения
- гибкая настройка ролей пользователей в различных проектах
- контроль за выполнением поставленных задач
- общий доступ к файлам
- создание Wiki-страниц и форума для каждого проекта
- интеграция проектов с системами контроля версий
- рассылка уведомлений по электронной почте
- поддержка разных языков

Разработка систем DVM и САПФОР

- распределение задач между разработчиками
- расстановка приоритетов задач
- отслеживание статуса задач и найденных ошибок

Обучение параллельному программированию

Создание учебного проекта → Регистрация участника спецсеминара в системе → Назначение роли Student в учебном проекте → Индивидуальная постановка задачи → Выполнение задачи под контролем преподавателя

Обучение параллельному программированию

Проект состоит из описания задания и набора индивидуальных задач.
Студенты знакомятся с различными технологиями параллельного программирования (OpenMP, OpenACC, DVMH).
Выполнение задания происходит в несколько этапов, с каждым этапом сложность постепенно возрастает.

Обучение параллельному программированию

Создание учебного проекта

Регистрация участника спецсеминара в системе

Назначение роли Student в учебном проекте

Индивидуальная постановка задачи

Выполнение задачи под контролем преподавателя

Home My page Projects Administration Help

Logged in as N_Kataev My account Sign out

Practical training 2014

Search: Practical training 2014

Overview Activity Issues New issue Gantt Calendar Wiki Files Settings

Задание 2014 г.

Edit Watch Lock Rename Delete History

В 2014 году каждому студенту выдаются в качестве исходных две Фортран-программы – последовательная версия одного из тестов NAS NPB 3.3.1 и ее параллельная версия в модели OpenMP. Требуется распараллелить последовательную программу в моделях OpenACC и DVMH. Это задание разбивается на следующие этапы, каждому из которых в итоговом отчете соответствует раздел.

Задание 2014 г.
1. Получение на целевой ЭВМ времен выполнения
2. Профилирование выполнения последовательной программы
3. Профилирование выполнения OpenMP-версии на 1, 2, 4, 8, 12 нитях
4. Распараллеливание программы на OpenACC и получение на целевой ЭВМ времен выполнения на одном ГПУ
5. Распараллеливание программы на DVM и получение на целевой ЭВМ времен выполнения
6. Распараллеливание программы на DVMH и получение на целевой ЭВМ времен выполнения

Доступ к целевым ЭВМ осуществляется по SSH

1. Получение на целевой ЭВМ времен выполнения

Получение на целевой ЭВМ (например, К-100, Ломоносов или др.) для нескольких классов начальных данных (классы A, C) следующих времен выполнения:

- время выполнения последовательной версии;
- время последовательного выполнения OpenMP-версии;
- время выполнения OpenMP-версии на 1, 2, 4, 8, 12 нитях одного узла.

2. Профилирование выполнения последовательной программы

- получение информации с помощью инструментов Интела;
- получение информации о временах выполнения всех циклов с помощью инструментов DVM (задание опции –e4, расширение fdv, инструкция PA);
- определение самых времяемких циклов;
- оценка накладных расходов профилирования.

Профилирование последовательной и OpenMP-программы может осуществляться при помощи:

- Intel VTune Amplifier XE 2013
- Intel Advisor XE

Данные инструменты входят в состав Intel Cluster Studio XE 2013, Intel Parallel Studio XE или Intel Fortran Studio XE.
Пробные версии для Windows (на 30 дней) или версию для некоммерческого использования для Linux можно скачать с сайта
http://software.intel.com/ru-ru/

Для профилирования OpenMP-программы на К-100 можно использовать опцию -openmp-profile.
В результате запуска программы, скомпилированной с опцией -openmp-profile, на выполнение создается текстовый файл guide.gvs - программа разбивается на последовательные и параллельные участки, и для каждого такого участка собираются характеристики эффективности.

Также для профилирования может быть полезен AMD CodeXL:
http://developer.amd.com/tools-and-sdks/heterogeneous-computing/codexl/

3. Профилирование выполнения OpenMP-версии на 1, 2, 4, 8, 12 нитях

- получение информации с помощью инструментов Интела;
- определение времяемких циклов, которые не распараллеливаются или мало ускоряются.

4. Распараллеливание программы на OpenACC и получение на целевой ЭВМ времен выполнения на одном ГПУ

5. Распараллеливание программы на DVM и получение на целевой ЭВМ времен выполнения

Распараллеливание программы на DVM и получение на целевой ЭВМ времен выполнения на 1, 2, 4, 8, 12 процессах одного узла, а также времени выполнения этой DVM-программы в последовательном режиме.

6. Распараллеливание программы на DVMH и получение на целевой ЭВМ времен выполнения

Распараллеливание программы на DVMH и получение на целевой ЭВМ времен выполнения на 1,2,3 ГПУ.

Wiki

Start page
Index by title
Index by date

Проект состоит из описания задания и набора индивидуальных задач.
Студенты знакомятся с различными технологиями параллельного программирования (OpenMP, OpenACC, DVMH).
Выполнение задания происходит в несколько этапов, с каждым этапом сложность постепенно возрастает.

Обучение параллельному программированию

Создание учебного проекта

Проект состоит из описания задания и набора индивидуальных задач.
Студенты знакомятся с различными технологиями параллельного программирования (OpenMP, OpenACC, DVMH).
Выполнение задания происходит в несколько этапов, с каждым этапом сложность постепенно возрастает.

Регистрация участника спецсеминара в системе

Создается аккаунт, связанный с почтовым адресом студента.
На указанный адрес приходит информация о статусе выполняемых студентом задач, отправляются оповещения о появлении новых задач, о том, считается ли задача сданной.

Назначение роли Student в учебном проекте

Индивидуальная постановка задачи

Выполнение задачи под контролем преподавателя

Обучение параллельному программированию

Создание учебного проекта

Проект состоит из описания задания и набора индивидуальных задач.
Студенты знакомятся с различными технологиями параллельного программирования (OpenMP, OpenACC, DVMH).
Выполнение задания происходит в несколько этапов, с каждым этапом сложность постепенно возрастает.

Регистрация участника спецсеминара в системе

Создается аккаунт, связанный с почтовым адресом студента.
На указанный адрес приходит информация о статусе выполняемых студентом задач, отправляются оповещения о появлении новых задач, о том, считается ли задача сданной.

Назначение роли Student в учебном проекте

В разных проектах допускается исполнение разных ролей (Manager, Developer, Student, Reporter).
Для каждой роли определены разрешения на уровне отдельного проекта (можно ли создавать и редактировать задачи, оставлять к ним комментарии, просматривать Wiki-страницу проекта и т.д.).

Индивидуальная постановка задачи

Выполнение задачи под контролем преподавателя

Обучение параллельному программированию

Создание учебного проекта

Проект состоит из описания задания и набора индивидуальных задач.
Студенты знакомятся с различными технологиями параллельного программирования (OpenMP, OpenACC, DVMH).
Выполнение задания происходит в несколько этапов, с каждым этапом сложность постепенно возрастает.

Регистрация участника спецсеминара в системе

Создается аккаунт, связанный с почтовым адресом студента.
На указанный адрес приходит информация о статусе выполняемых студентом задач, отправляются оповещения о появлении новых задач, о том, считается ли задача сданной.

Назначение роли Student в учебном проекте

В разных проектах допускается исполнение разных ролей (Manager, Developer, Student, Reporter).
Для каждой роли определены разрешения на уровне отдельного проекта (можно ли создавать и редактировать задачи, оставлять к ним комментарии, просматривать Wiki-страницу проекта и т.д.).

Индивидуальная постановка задачи

Основная задача состоит из подзадач, соответствующих определенным этапам обучения.

Выполнение задачи под контролем преподавателя

Обучение параллельному программированию

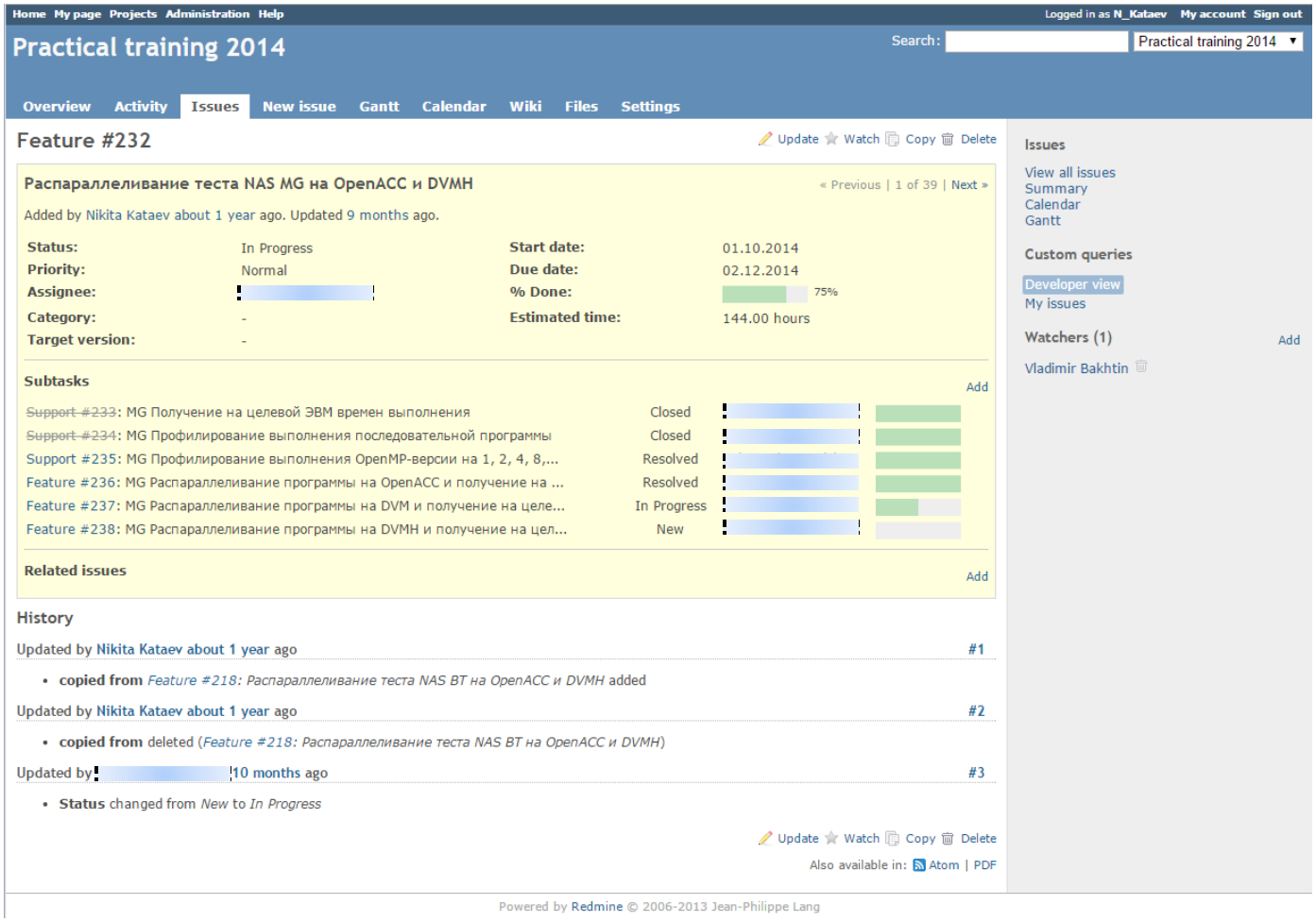
Создание учебного проекта

Регистрация участника спецсеминара в системе

Назначение роли Student в учебном проекте

Индивидуальная постановка задачи

Выполнение задачи под контролем преподавателя



Основная задача состоит из подзадач, соответствующих определенным этапам обучения.

Обучение параллельному программированию

Создание учебного проекта

Проект состоит из описания задания и набора индивидуальных задач.
Студенты знакомятся с различными технологиями параллельного программирования (OpenMP, OpenACC, DVMH).
Выполнение задания происходит в несколько этапов, с каждым этапом сложность постепенно возрастает.

Регистрация участника спецсеминара в системе

Создается аккаунт, связанный с почтовым адресом студента.
На указанный адрес приходит информация о статусе выполняемых студентом задач, отправляются оповещения о появлении новых задач, о том, считается ли задача сданной.

Назначение роли Student в учебном проекте

В разных проектах допускается исполнение разных ролей (Manager, Developer, Student, Reporter).
Для каждой роли определены разрешения на уровне отдельного проекта (можно ли создавать и редактировать задачи, оставлять к ним комментарии, просматривать Wiki-страницу проекта и т.д.).

Индивидуальная постановка задачи

Основная задача состоит из подзадач, соответствующих определенным этапам обучения.

Выполнение задачи под контролем преподавателя

С каждой задачей связан аккаунт одного или нескольких преподавателей, отслеживающих ее выполнение.
Допускается четыре стадии выполнения задачи (New, In Progress , Resolved , Closed).
В процессе выполнения студент указывает время, которое он затратил на каждый этап задания.

Обучение параллельному программированию

Создание учебного проекта

Регистрация участника спецсеминара в системе

Назначение роли Student в учебном проекте

Индивидуальная постановка задачи

Выполнение задачи под контролем преподавателя

#	Tracker	Status	Priority	Subject	Updated	% Done
7						
232	Feature	In Progress	Normal	Распараллеливание теста NAS MG на OpenACC и DVMH	22.12.2014 22:10	
233	Support	Closed	Normal	MG Получение на целевой ЭВМ времен выполнения	25.05.2015 23:22	
234	Support	Closed	Normal	MG Профилирование выполнения последовательной программы	25.05.2015 23:23	
235	Support	Resolved	Normal	MG Профилирование выполнения OpenMP-версии на 1, 2, 4, 8, 12 нитях	28.10.2014 09:07	
236	Feature	Resolved	Normal	MG Распараллеливание программы на OpenACC и получение на целевой ЭВМ времен выполнения на одном ГПУ	02.12.2014 16:52	
237	Feature	In Progress	Normal	MG Распараллеливание программы на DVM и получение на целевой ЭВМ времен выполнения	22.12.2014 22:10	
238	Feature	New	Normal	MG Распараллеливание программы на DVMH и получение на целевой ЭВМ времен выполнения	10.09.2014 12:11	

Распараллеливание программы на DVMH и получение на целевой ЭВМ времен выполнения на 1,2,3 ГПУ.

- NPB3.3-DVMH.zip - Исходный код для запуска на K100 (78.233 KB) 19.11.2013 06:34
- MG.FDV.zip - Вариации кода программы (71.146 KB) 19.11.2013 06:34
- results.zip - Результаты измерений (4.837 MB) 19.11.2013 06:34
- remote.xlsx - Таблица с данными по REMOTE_ACCESS (11.277 KB) 19.11.2013 06:35
- times.xlsx - Сводная таблица измерений (24.448 KB) 19.11.2013 06:36
- Отчет.docx - Отчет (15.926 KB) 19.11.2013 06:36
- Отчет.docx - Новый отчет (16.762 KB) 24.11.2013 15:29
- report.xlsx - Временя (26.056 KB) 24.11.2013 15:29
- sources.zip - Исходники (29.746 KB) 24.11.2013 15:32
- results.7z - RAW results (6.605 MB) 24.11.2013 15:34
- optimized.zip - Финальный отчет и результаты по этапу (7.534 MB) 21.12.2013 20:43

Допускается четыре стадии выполнения задачи (New, In Progress , Resolved , Closed).
Когда задача считается законченной, к ее описанию добавляется отчет о выполнении.

Цель создания сайта DVM-системы

предоставить пользователям разного уровня подготовки, в том числе студентам, изучающим параллельное программирование, актуальную информацию о возможностях системы, заинтересовать и привлечь новых пользователей, обеспечить удобный доступ к системе и оперативную обратную связь с пользователями.

Пять основных разделов

- главная страница – общая информация, последние новости;
- «Изучите» - примеры программ , описания DVMH-модели и языков Fortran-DVMH и C-DVMH, данные о производительности DVMH-программ;
- «Воспользуйтесь» - получение доступа к системе;
- «Прочтите и посмотрите» - опубликованные статьи, презентации и видео с выступлений;
- «Свяжитесь с нами» – форма обратной связи.

Три возможности использования DVM-системы

- установить систему на своем компьютере (Linux, Windows, Mac OS X);
- получить удаленный доступ к рабочей станции;
- воспользоваться суперкомпьютерами, на которых система установлена («Ломоносов» (НИВЦ МГУ), К100 (ИПМ им. М.В. Келдыша), МВС-10П (МСЦ РАН), МВС-100k (МСЦ РАН).

Сайт DVM-системы
http://dvm-system.org/

DVM SYSTEM Study Download Read and View Contact us Search Рус Eng

Work where you feel comfortable

DVM-system can be used on personal computers and workstations and is already installed on different supercomputers.

LOMONOSOV
SRCC MSU

K100
KIAM RAS

MVS-10P
JSCC RAS

MVS-100k
JSCC RAS

Try DVM-system on our server

You can get a login to the server with installed DVM-system.

Study getting started guide

What is DVMH-model, DVMH-language and DVMH-program?

How to start using DVM-system?

See examples of programs

How to simply and efficiently parallelize methods for solving systems of linear equations?

NEWS

PCT 04/2015

March 30th – April 3rd, 2015

Automation of serial Fortran-program transformation, nessesary for its efficient parallelization by SAPFOR system

Presented by N.A. Kataev.

Read (PDF, Russian) View (PDF, Russian)

Parallelization of NAS NPB3.3.1 tests on Fortran-DVMH for Intel Xeon Phi coproccessor

Presented by V.A. Bakhtin.

Read (PDF, Russian)

ABRAU 09/2014

Semptember 22nd – 27th, 2014

Automatic mapping of Fortran programs on clusters with accelerators

Presented by V.A. Bakhtin.

Read (PDF, Russian)

Execution performance of NAS NBP 3.3

May 2015

Performance benchmarks from NAS NPB 3.3 had been parallelized using Fortran-DVMH language and were executed on graphics accelerators NVIDIA, on 6-cores processor Intel and on 60-cores Intel Xeon Phi.

NSCF 11/2014

November 25th – 27th, 2014

Study the performance of the DVMH-programs on clusters with multicore processors and accelerators

Presented by V.A. Bakhtin.

View (PDF, Russian)

DVM-system was developed in Keldysh Institute of Applied Mathematics, Russian Academy of Sciences, with the active participation of graduate students and students of Faculty of Computational Mathematics and Cybernetics of Lomonosov Moscow State University. It is designed to create parallel programs of scientific-technical calculations in C-DVMH and Fortran-DVMH languages. These languages use the same model of parallel programming (DVMH-model) and are the extensions of standard C and Fortran languages by parallelism specifications, implemented as compiler directives. The directives are invisible to standard compilers, so a programmer can have one program for sequential and for parallel execution on computers of different architectures.

DVMH-model allows to create efficient parallel programs (DVMH-programs) for heterogeneous computational clusters, which nodes use as computing devices not only universal multi-core processors but also can use attached accelerators (GPUs or Intel Xeon Phi coprocessors). In last case computations mapped to the node can be automatically distributed between the computing devices of the node taking into account their performance.

C-DVMH and Fortran-DVMH compilers convert the source program into a parallel program using standard programming technologies MPI, OpenMP and CUDA.

DVM-system includes the tools of functional debugging and performance debugging of DVMH-programs.

DVM System, Copyright© 2015

Спасибо за внимание ☺

?

http://dvm-system.org/