

Министерство науки и высшего образования РФ
Федеральное государственное автономное образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Б1.В.04 Параллельное программирование

наименование дисциплины (модуля) в соответствии с учебным планом

Направление подготовки / специальность

01.03.02 Прикладная математика и информатика

Направленность (профиль)

01.03.02.31 Математическое моделирование и вычислительная
математика

Форма обучения

очная

Год набора

2021

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Программу составили _____

к.ф.-м.н., доцент, Кареева Е.Д.

должность, инициалы, фамилия

1 Цели и задачи изучения дисциплины

1.1 Цель преподавания дисциплины

Дисциплина «Параллельное программирование» предназначена для изучения средств и методов создания приложений для различных архитектур вычислительных систем (ВС). При изучении дисциплины уделяется особое внимание получению практических навыков написания параллельных программ в терминах конкретных библиотек и/или языковых реализаций для ВС как с общей, так и распределенной памятью (в том числе, многоядерных и кластерных архитектур).

В рамках изучения дисциплины рассматриваются базовые сведения о ряде систем программирования, позволяющие начать разработку параллельных программ для параллельных ВС с общей и распределенной памятью. Знания и навыки, полученные при изучении дисциплины, позволяют в дальнейшем перейти к более детальному освоению инструментальных средств разработки параллельных программ.

Дисциплина «Параллельное программирование» согласно учебному плану является дисциплиной по выбору вариативной части профессионального цикла (Б1.В.ДВ.6) по направлению 02.03.01 «Математика и компьютерные науки».

Дисциплина изучается в последнем (8-ом) семестре бакалавриата и завершает формирование основных общенаучных, общепрофессиональных и профессиональных компетенций студента.

Изучение дисциплины формирует базу для углубленного изучения параллельного программирования для высокопроизводительных ВС в магистратуре.

1.2 Задачи изучения дисциплины

При изучении дисциплины учащиеся должны решить следующие основные задачи.

1. Получить представление об особенностях архитектуры различных классов ПВС, задачах, которые невозможно решить без помощи высокопроизводительных вычислений и параллельного программирования.

2. Изучить основные проблемы, возникающие при программировании для ВС с общей памятью и пути их разрешения, понять особенности синхронизации и взаимодействия потоков.

3. Рассмотреть основные концепции и средства, предлагаемые в области параллельного программирования.

4. Изучить основные функции WinAPI, используемые для создания и управления потоками в ОС Windows.

5. Провести сравнительный анализ теоретических основ и различных языковых реализаций механизма синхронизации и взаимодействия потоков для ВС с общей памятью.

6. Изучить основные проблемы, возникающие при программировании для ВС с распределенной памятью и пути их разрешения, понять особенности синхронизации и взаимодействия процессов.

7. Рассмотреть SPMD-модель организации параллельных вычислений на основе одновременного выполнения одной и той же программы на нескольких процессорах с организацией взаимодействия процессов при помощи передачи сообщений.

8. Получить практические навыки программирования с использованием библиотеки передачи сообщений MPI для ВС с распределенной памятью.

1.3 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

Код и наименование индикатора достижения компетенции	Запланированные результаты обучения по дисциплине
ПК-4: Способен использовать современные методы разработки и реализации алгоритмов на базе языков программирования и пакетов прикладных программ моделирования	
ПК-4.1: Применяет знания современных методов разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования при решении конкретных задач	<p>Современные методы реализации базовых алгоритмов программирования</p> <p>Современные методы реализации нескольких алгоритмов для решения базовых задач программирования</p> <p>Современные методы реализации нескольких алгоритмов для решения базовых задач программирования, а также алгоритмы, реализующиеся в пакетах прикладных программ</p> <p>Применять методы реализации базовых алгоритмов программирования</p> <p>Применять современные методы реализации нескольких алгоритмов для решения базовых задач программирования</p> <p>Применять современные методы реализации нескольких алгоритмов для решения базовых задач программирования, а также алгоритмы, реализующиеся в пакетах прикладных программ</p> <p>Навыком использования методов реализации базовых алгоритмов программирования</p> <p>Навыком использования методов реализации нескольких алгоритмов для решения базовых задач программирования</p> <p>Навыком использования современных методов реализации нескольких алгоритмов для решения базовых задач программирования, а также алгоритмов, реализующиеся в пакетах прикладных программ</p>

ПК-4.2: Разрабатывает и реализовывает алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования	Современные методы реализации базовых алгоритмов программирования Современные методы реализации нескольких алгоритмов для решения базовых задач программирования Современные методы реализации нескольких
	алгоритмов для решения базовых задач программирования, а также алгоритмы, реализующиеся в пакетах прикладных программ Реализовывать базовые алгоритмы программирования Разрабатывать и реализовывать алгоритмы решения задач программирования Разрабатывать и реализовывать алгоритмы решения задач программирования, в том числе с помощью пакетов прикладных программ Навыком реализации базовых алгоритмов программирования Навыком разработки и реализации алгоритмов решения задач программирования Навыком разработки и реализации алгоритмов решения задач программирования, в том числе с помощью пакетов прикладных программ

1.4 Особенности реализации дисциплины

Язык реализации дисциплины: Русский.

Дисциплина (модуль) реализуется с применением ЭО и ДОТ

URL-адрес и название электронного обучающего курса: <http://study.sfu-kras.ru/course/category.php?id=13> .

2. Объем дисциплины (модуля)

Вид учебной работы	Всего, зачетных единиц (акад.час)	е
		1
Контактная работа с преподавателем:	1,67 (60)	
занятия лекционного типа	0,67 (24)	
практические занятия	1 (36)	
Самостоятельная работа обучающихся:	1,33 (48)	
курсовое проектирование (КП)	Нет	
курсовая работа (КР)	Нет	
Промежуточная аттестация (Экзамен)	1 (36)	

3 Содержание дисциплины (модуля)

3.1 Разделы дисциплины и виды занятий (тематический план занятий)

		Контактная работа, ак. час.							
№ п/п	Модули, темы (разделы) дисциплины	Занятия лекционного типа		Занятия семинарского типа				Самостоятельная работа, ак. час.	
				Семинары и/или Практические занятия		Лабораторные работы и/или Практикумы			
		Всего	В том числе в ЭИОС	Всего	В том числе в ЭИОС	Всего	В том числе в ЭИОС	Всего	В том числе в ЭИОС
1. Обзор области параллельного программирования. Технологии, парадигмы, программные средства									
	1. Введение. Параллельные компьютеры и суперЭВМ.	2							
	2. Введение. Особенности параллельного программирования.	2							
	3. Анатомия простого многопоточного приложения. Компиляция, компоновка и запуск готового многопоточного приложения с помощью WIN API.			4					
	4. Введение. Параллельные компьютеры и суперЭВМ.							2	2
	5. Введение. Особенности параллельного программирования.							4	4
2. Особенности разработки параллельных приложений для систем с общей памятью									
	1. Процессы и потоки. Реализации многопоточности.	2							
	2. Механизм синхронизации процессов. Решения с активным ожиданием.	2							
	3. Семафоры и мониторы. Реализации.	6							

4. Практические приемы построения многопоточных приложений. Алгоритмы с активным ожиданием. Решение задачи с помощью WIN API.			4					
5. Механизмы синхронизации в пользовательском режиме и с помощью объектов ядра.			4					
6. Создание многопоточных приложений с синхронизацией потоков с помощью WIN API.			6					
7. Создание многопроцессных приложений, работающих с разделяемым ресурсом с синхронизацией процессов с помощью WIN API.			4					
8. Процессы и потоки. Реализации многопоточности.							4	4
9. Механизм синхронизации процессов. Решения с активным ожиданием.							4	4
10. Семафоры и мониторы. Реализации.							4	4
3. Особенности разработки параллельных приложений для систем с распределенной памятью								
1. Примитивы передачи сообщений.	1							
2. Работа на кластере СФУ.	1							
3. Программирование с помощью библиотеки MPI.	2							
4. Двухточечные обмены в библиотеке MPI.	5							
5. Коллективные обмены в библиотеке MPI.	1							
6. Работа на кластере рабочих станций, работа в среде MVS-1000. Компиляция и запуск готового приложения, использующего библиотеку MPI.			4					
7. Создание приложения с помощью библиотеки MPI. Различные способы двухточечного обмена в MPI.			6					
8. Коллективные обмены в MPI. Ввод и вывод в MPI.			4					
9. Примитивы передачи сообщений.							8	8

10. Работа на кластере СФУ.							8	8
11. Программирование с помощью библиотеки MPI.							6	6
12. Двухточечные обмены в библиотеке MPI.							4	4
13. Коллективные обмены в библиотеке MPI.							4	4
Всего	24		36				48	48

4 Учебно-методическое обеспечение дисциплины

4.1 Печатные и электронные издания:

1. Карепова Е. Д., Кузьмин Д. А., Легалов А. И., Редькин А. В., Удалова Ю. В., Федоров Г. А. Средства разработки параллельных программ: электрон. учеб.-метод. комплекс дисциплины(Красноярск: ИПК СФУ).
2. Гергель В. П. Высокопроизводительные вычисления для многопроцессорных многоядерных систем: учебник для студентов вузов (Москва: Изд-во МГУ).
3. Карепова Е. Д. Основы многопоточного и параллельного программирования: учебное пособие для студентов вузов, обучающихся по направлениям "Прикладная математика и информатика" и "Фундаментальная информатика и информационные технологии"(Красноярск: СФУ).
4. Немнюгин С.А., Стесик О.Л. Параллельное программирование для многопроцессорных вычислительных систем: Пособие(Санкт-Петербург: Издательство "БХВ-Петербург").
5. Ортега Д. М., Икрамов Х. Д. Введение в параллельные и векторные методы решения линейных систем: перевод с английского(Москва: Мир).
6. Таненбаум Э. Современные операционные системы(Москва: Питер).
7. Корняков К. В., Кустикова В. Д., Мееров И. Б., Сиднев А. А., Сысоев А. В., Шишков А. В., Гергель В. П. Инструменты параллельного программирования в системах с общей памятью: учебник для студентов вузов(Москва: Изд-во МГУ).
8. Старченко А. В., Данилкин Е. А., Лаева В. И., Проханов С. А., Старченко А. В. Практикум по методам параллельных вычислений: учебник для студентов вузов(Москва: Изд-во МГУ).
9. Антонов А. С. Технологии параллельного программирования MPI и OpenMP: учебное пособие для студентов вузов, обучающихся по напр. 010400 "Прикладная математика и информатика", 010300 "Фундаментальная информатика и информационные технологии"(Москва: Изд-во Московского университета).
10. Гергель В. П. Современные языки и технологии параллельного программирования: учебник для студентов вузов, обуч. по направлениям 010400 "Прикладная математика и информатика" и 010300 "Фундаментальная информатика и информационные технологии"(Москва: Издательство Московского университета).
11. Малышкин В. Э., Корнеев В. Д. Параллельное программирование мультимедийных компьютеров: [учебник для вузов](Новосибирск: Издательство НГТУ).
12. Таненбаум Э., Гребеньков А. Компьютерные сети(Москва: Питер).
13. Таненбаум Э., Бос Х. Современные операционные системы(Москва: Питер).

14. Таненбаум Э. Компьютерные сети(СПб.: Питер).
15. Таненбаум Э. Современные операционные системы(СПб.: Питер).
16. Рихтер Дж Windows для профессионалов: создание эффективных Win32 приложений с учетом специфики 64-разрядной версии Windows(СПб.: Питер).
17. Шлее М. Qt4.5. Профессиональное программирование на C++: Практическое руководство(Санкт-Петербург: Издательство "БХВ-Петербург").
18. Воеводин В. В., Воеводин В. В. Параллельные вычисления: Пособие (Санкт-Петербург: Издательство "БХВ-Петербург").
19. Сиротинина Н. Ю., Миркес Е. М., Кареева Е. Д. Параллельные вычислительные системы: учебно-методический комплекс дисциплины (Красноярск: ИПК СФУ).

4.2 Лицензионное и свободно распространяемое программное обеспечение, в том числе отечественного производства (программное обеспечение, на которое университет имеет лицензию, а также свободно распространяемое программное обеспечение):

1. Лекционные занятия должны проводиться в аудитории, оснащенной проекционным оборудованием, обеспечивающим показ компьютерных презентаций. Используемый при этом компьютер должен быть оснащен программным обеспечением, позволяющим выполнять необходимые примеры программного кода.
2. Лекционные занятия должны проводиться в аудитории, оснащенной проекционным оборудованием, обеспечивающим показ компьютерных презентаций. Используемый при этом компьютер должен быть оснащен программным обеспечением, позволяющим выполнять необходимые примеры программного кода.
3. Компьютер может быть оснащен любой из операционных систем, обеспечивающих поддержку инструментальных средств, необходимых для проведения занятий. В настоящее время основной упор сделан на использование операционной системы Windows. Однако возможно и использование ОС Linux, которая содержит программные средства, позволяющие использовать как уже разработанное методическое обеспечение, так и применяемые программы.
4. Для успешного проведения лекционных занятий необходимо обеспечить показ презентаций в формате MS Power Point и Adobe Acrobat Reader. Внимание! Презентации содержат специфические объекты и плохо отображаются в формате Open Office.

5. Также необходимо иметь установленные средства для отображения исходных текстов программ на используемых в примерах языках программирования (C++, C). В простейшем случае это могут быть обычные текстовые редакторы. Однако целесообразнее использовать текстовые редакторы с подсветкой синтаксиса или среды разработки, обеспечивающие отображение исходных текстов с подсветкой. Желательна установка ПО со средой разработки, позволяющей собрать проект, откомпилировать и запустить его как в обычном режиме, так и в режиме отладки. Для ОС Windows – это может быть как продукт Visual C, входящий в пакет Microsoft Visual Studio версии не ниже 6.0. Возможно использование более простой среды, например, Borland C или свободно распространяемого компилятора MinGW C++.
6. Помимо этого целесообразно иметь выход в сеть Сибирского федерального университета для подключения к суперкомпьютеру, имеющимся кластерным системам и вычислительным системам с многоядерной архитектурой.
- 7.
8. Лабораторные занятия должны проводиться в компьютерном классе или с использованием удаленного доступа к имеющимся кластерным системам и суперкомпьютеру. При изучении многопоточного программирования с применением Win 32 API желательно наличие компьютеров с многоядерной архитектурой или многопроцессорных систем с общей памятью. Это позволит не только попробовать изучаемые технологии, но и увидеть, как они функционируют на реальной многопроцессорной системе, оценить выигрыш, получаемый от использования параллельных вычислительных технологий.
- 9.
10. При изучении технологии MPI возможно использование кластера, построенного на базе компьютерного класса. Он позволяет отработать учебные программы в коллективном режиме и снять предварительные оценки. Для окончательного выполнения программ необходимо иметь сетевой доступ к одной из кластерных систем или суперкомпьютеру. Это кроме всего прочего позволит изучить методы и дисциплины обслуживания удаленных клиентов, применяемых при решении реальных вычислительных задач.

4.3 Интернет-ресурсы, включая профессиональные базы данных и информационные справочные системы:

1. Не требуется.

5 Фонд оценочных средств

Оценочные средства находятся в приложении к рабочим программам дисциплин.

6 Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине (модулю)

Лекционные занятия должны проводиться в аудитории, оснащенной проекционным оборудованием, обеспечивающим показ компьютерных презентаций.

Лабораторные работы должны проводиться в компьютерном классе. Желательно, чтобы используемые компьютеры были многоядерными (многопроцессорными системами на общей памяти). Для проведения лабораторных работ № 6-8 необходим доступ к одному из высокопроизводительных кластеров по протоколам ssh и sftp.