**Санкт-Петербургский государственный университет**

**Р А Б О Ч А Я П Р О Г Р А М М А**

**УЧЕБНОЙ ДИСЦИПЛИНЫ**

Распараллеливание и синхронизация

Paralleling and Synchronization

**Язык(и) обучения**

русский

Трудоемкость в зачетных единицах: 3

Регистрационный номер рабочей программы: 042873

Санкт-Петербург

2020

**Раздел 1. Характеристики учебных занятий**

**1.1. Цели и задачи учебных занятий**

Данная дисциплина предназначена для обучения распараллеливанию вычислений и изучению проблем, связанных с синхронизацией.

Предполагается теоретическое и практическое освоение обучающимися основ распараллеливания вычислений в системах с распределенной и общей памятью, формирование у обучающихся общего представления об архитектурах современных суперкомпьютерных вычислительных систем; содержании, задачах и методах параллельного программирования как современного направления, самостоятельной научной и инженерной дисциплины, диапазона и разнообразия ее типичных приложений.

В процессе изучения дисциплины обучающиеся получают представление об особенностях структуры и функционирования параллельных программ, методах их разработки, получают практические навыки компиляции, отладки, тестирования параллельных программ и анализа полученных результатов в сравнении с последовательным вариантом на различном числе процессоров и с различной топологией.

**1.2. Требования подготовленности обучающегося к освоению содержания учебных занятий (пререквизиты)**

умение программировать в СИ,

знание основ вычислительной математики.

**1.3. Перечень результатов обучения (learning outcomes)**

В результате освоения дисциплины обучающиеся должны

* иметь общее представлениеоб архитектуре параллельных многоядерных процессоров**;**
* владеть принципами разработки параллельных программ;
* иметь представление об архитектурах суперкомьютерных вычислительных систем, технологиях параллельного программирования, а также некоторых специальных приемах и навыках в области современного параллельного программирования;
* иметь представление о возможностях применения знаний, излагаемых в разделах курса в различных прикладных областях науки и техники.

уметь

* строить параллельные формы алгоритмов;
* формализовывать поставленные задачи и реализовывать сложные программные комплексы как с точки зрения грамотной профессиональной разработки различного рода проектов, так и с точки зрения управления психологическим климатом в процессе работы в коллективе разработчиков для достижения эффективного результата.

ПКП-1 Способность проводить научные исследования на основе существующих методов в конкретной области профессиональной деятельности

ПКП-2 Способен проводить патентные исследования, определять формы и методы правовой охраны и защиты прав на результаты интеллектуальной деятельности, распоряжаться правами на них для решения задач в области развития науки, техники и технологии

ПКП-8 Способен использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков

программирования и пакетов прикладных программ моделирования

ПКП-9 Способен участвовать в разработке технической документации программных продуктов и программных комплексов

**1.4. Перечень активных и интерактивных форм учебных занятий**

Выступление на семинарах и участие в обсуждениях возникающих проблем при распараллеливании вычислений. Решение практических задач по распараллеливанию вычислений.

**Раздел 2. Организация, структура и содержание учебных занятий**

**2.1. Организация учебных занятий**

**2.1.1**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Трудоёмкость, объёмы учебной работы и наполняемость групп обучающихся | | | | | | | | | | | | | | | | | | |
| Код модуля в составе дисциплины,  практики и т.п. | Контактная работа обучающихся с преподавателем | | | | | | | | | | | | Самостоятельная работа | | | | Объём активных и интерактивных  форм учебных занятий | Трудоёмкость |
| лекции | семинары | консультации | практические  занятия | лабораторные работы | контрольные работы | коллоквиумы | текущий контроль | промежуточная  аттестация | итоговая аттестация | под руководством преподавателя | в присутствии  преподавателя | сам. раб. с использованием  методических материалов | текущий контроль (сам.раб.) | промежуточная аттестация (сам.раб.) | итоговая аттестация  (сам.раб.) |
| ОСНОВНАЯ ТРАЕКТОРИЯ | | | | | | | | | | | | | | | | | | |
| Форма обучения: очная | | | | | | | | | | | | | | | | | | |
| Семестр 4 | 18 | 18 | 2 |  |  |  |  |  | 2 |  |  |  | 38 |  | 30 |  | 20 | 3 |
|  | 2-15 | 2-15 | 2-15 |  |  |  |  |  | 2-150 |  |  |  | 1-1 |  | 1-1 |  |  |  |
| ИТОГО | 18 | 18 | 2 |  |  |  |  |  | 2 |  |  |  | 38 |  | 30 |  |  | 3 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Виды, формы и сроки текущего контроля успеваемости и промежуточной аттестации | | | | | | |
| Код модуля в составе дисциплины, практики и т.п. | Формы текущего контроля успеваемости | | Виды промежуточной аттестации | | Виды итоговой аттестации  (только для программ итоговой аттестации и дополнительных образовательных программ) | |
| Формы | Сроки | Виды | Сроки | Виды | Сроки |
| ОСНОВНАЯ ТРАЕКТОРИЯ | | | | | | |
| Форма обучения: очная | | | | | | |
| Семестр 4 |  |  | экзамен, устно, традиционная форма | по графику промежуточной аттестации |  |  |

**2.2. Структура и содержание учебных занятий**

1. Распределенное программирование. Краткий обзор системы распределенного параллельного программирования– MPI

Виртуальные топологии. Парные взаимодействия. Коллективные взаимодействия.

MPI - Типы данных.

1. Параллельное программирование с разделяемыми переменными.

Система параллельного программирования ОpenMP. Краткий обзор функций OpenMP.

1. Процесс. Модель процесса. Создание процесса. Завершение процесса. Моделирование режима многозадачности.

Взаимодействие процессов. Критические области. Семафоры. Передача сообщений. Синхронизация процессов через доступ к общим ресурсам Задача взаимного исключения. Монитор Хоара. Задача производитель-потребитель с ограниченным буфером. Задача читатели-писатели. Взаимная блокировка (deadlok). Задача о пяти обедающих философах.

1. Синхронизация процессов через обмен данными

Передача сообщений в системах параллельного программирования. Параллельная программа разделения множеств (Дейкстра). Модели с параллельно-последовательными структурами управления.

1. Модели вычислений. Модель вычислений методом параллельных операторов (ярусно-параллельные формы). Модель вычислений методом параллельных ветвей. Модель параллельно-последовательного программирования ( fork-join).

Модели вычислений MPMD, SPMD.

1. Ускорение и эффективность. Законы Амдала.
2. Модели асинхронных вычислений. Событийное управление. Потоковые вычисления. Динамическое управление. Синхронные вычисления.
3. Языки параллельного программирования.
4. Разработка параллельных программ алгоритма умножения матрицы на вектор и матрицы на матрицу. Разработка параллельных программ для решения систем линейных алгебраических уравнений методом простой итерации.

Разработка параллельных программ для решения систем линейных алгебраических уравнений методом Гаусса.

Разработка параллельных программ сортировки множеств.

Разработка параллельных программ для решения уравнений Пуассона в двух и трехмерных областях.

Примерные темы докладов

Ускорение и эффективность. Законы Амдала.

Модели асинхронных вычислений. Событийное управление. Потоковые вычисления. Динамическое управление. Синхронные вычисления.

Языки параллельного программирования.

Разработка параллельных программ алгоритма умножения матрицы на вектор и матрицы на матрицу. Разработка параллельных программ для решения систем линейных алгебраических уравнений методом простой итерации.

Разработка параллельных программ для решения систем линейных алгебраических уравнений методом Гаусса.

Разработка параллельных программ сортировки множеств.

Разработка параллельных программ для решения уравнений Пуассона в двух и трехмерных областях.

**Раздел 3. Обеспечение учебных занятий**

**3.1. Методическое обеспечение**

**3.1.1 Методические указания по освоению дисциплины**

Предмет предполагает выступления и обсуждение тем.

Решение практических задач по распараллеливанию.

**3.1.2 Методическое обеспечение самостоятельной работы**

Методические материалы для самостоятельной работы в списке литературы.

Список задач для решения формируется преподавателем

**3.1.3 Методика проведения текущего контроля успеваемости и промежуточной аттестации и критерии оценивания**

Обучающийся отвечает на вопросы билета. В билете два вопроса и выступление с двумя докладами.

Оценка «отлично» «A» на экзамене ставится за правильные ответы на вопросы билета и активное владение материалом курса в целом и выступление с двумя докладами.

Оценка «хорошо» «B» на экзамене ставится за правильные в целом ответы на вопросы билета и знание основных тем курса и выступление с двумя докладами.

Оценка «хорошо» «C» ставится за неполные ответы на вопросы билета при наличии не слишком грубых ошибок и выступление с двумя докладами.

Оценка «удовлетворительно» «D» ставится за неполный ответ на один из вопросов билета при наличии не слишком грубых ошибок и выступление с одним докладом.  
Оценка «удовлетворительно» «E» ставится за неполный ответs на все вопросы билета при наличии не слишком грубых ошибок и выступление с одним докладом.

Оценка «F» ставится за наличие грубых ошибок

**3.1.4 Методические материалы для проведения текущего контроля успеваемости и промежуточной аттестации (контрольно-измерительные материалы, оценочные средства)**

Образцы вопросов для подготовки к экзамену.

1. Модели параллельно-последовательного программирования. MPMD и SPMD модели программирования. Необходимость явного распараллеливания.

2. Определение процесса. Два главных типа взаимодействия параллельных процессов.

Задача взаимного исключения (определение).

Умение разработать параллельную программу на ОPENMP, MPI.

**3.1.5 Методические материалы для оценки обучающимися содержания и качества учебного процесса**

**3.2. Кадровое обеспечение.**

**3.2.1 Образование и (или) квалификация штатных преподавателей и иных лиц, допущенных к проведению учебных занятий.**

Доцент, профессор или ст. преподаватель

**3.2.2 Обеспечение учебно-вспомогательным и (или) иным персоналом**

Учебно-вспомогательный и инженерно-технический персонал должен иметь соответствующее высшее или среднее специальное образование, и обладать навыками организации работы с пользовательскими программными продуктами в локальной сети компьютерного класса и в Интернете.

**3.3. Материально-техническое обеспечение**

**3.3.1 Характеристики аудиторий (помещений, мест) для проведения занятий.**

Компьютерный класс с многопроцессорными компьютерами, с проектором, белой доской

**3.3.2 Характеристики аудиторного оборудования, в том числе неспециализированного компьютерного оборудования и программного обеспечения общего пользования**

Компьютерный класс, c многоядерными компьютерами, с установленными MPI и/или OPENMP.

**3.3.3 Характеристики специализированного оборудования.**

MPI и/или OPENMP.

**3.3.4 Характеристики специализированного программного обеспечения.**

Microsoft Visual Studio/2005,2008,2010. Linux.

**3.3.5 Перечень и объёмы требуемых расходных материалов**

Фломастеры для белой доски

**3.4. Информационное обеспечение**

**3.4.1 Список обязательной литературы.**

1. В. В. Воеводин, Вл. В. Воеводин. Параллельные вычисления. Изд-во: [БХВ-Петербург](http://www.ozon.ru/context/detail/id/1098685/), 2004 г., 608 с.

2. Ю.К.Демьянович. И.Г.Бурова и др. Параллельные алгоритмы. Разработка и реализация. Интуит: Бином. 2012 344 с.

**3.4.2 Список дополнительной литературы**

Нет.

**3.4.3 Перечень иных информационных источников**

Нет.

**Раздел 4. Разработчики программы**

Бурова И.Г, профессор кафедры вычислительной математики.

Мирошниченко И.Д, старший преподаватель кафедры параллельных алгоритмов.