

**ГОУ ВПО Российско-Армянский (Славянский)
университет**

Утверждено



**Директор Института
Математики и Информатики
Дарбинян А.А.
«18» июня 2024, протокол №15**

УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ДИСЦИПЛИНЫ

Наименование дисциплины: Параллельное программирование

Авторы: *канд. физ.-мат. наук Саргсян Севак Сеникович*

Направление подготовки: 01.04.02 Прикладная математика и информатика

1. АННОТАЦИЯ

1.1. Краткое описание содержания данной дисциплины;

Этот курс предоставляет студентам понимание основ и продвинутых концепций параллельного программирования. Курс охватывает теоретические аспекты, практические техники и инструменты для разработки параллельных приложений. Студенты научатся эффективно использовать многопоточность, распределенные системы и различные параллельные модели для решения вычислительных задач.

1.2. Трудоемкость в академических кредитах и часах, формы итогового контроля (экзамен/зачет);

2-ой семестр - 6 ЗЕТ- экзамен,

1.3. Взаимосвязь дисциплины с другими дисциплинами учебного плана специальности (направления)

Для прохождения этого курса изучение других дисциплин не требуются.

1.4. Результаты освоения программы дисциплины:

| Код компетенции | Наименование компетенции | Код индикатора достижения компетенций | Наименование индикатора достижений компетенций |
|-----------------|--|---------------------------------------|--|
| УК-3 | Способен организовывать и руководить работой команды, вырабатывая командную стратегию для достижения поставленной цели | УК-3.1 | Разрабатывает командную стратегию для достижения поставленной цели |
| | | УК-3.2 | Умеет организовывать и руководить работой команды |
| | | УК-3.3 | Демонстрирует понимание результатов работы команды и личных действий в ней |
| ПК-7 | способностью разрабатывать и оптимизировать бизнес-планы научно-прикладных проектов | ПК-7.1 | Знает основные критерии эффективности и качества функционирования системы |
| | | ПК-7.2 | Умеет руководить выполнением коллективной деятельностью |
| | | ПК-7.3 | Владеет методами постановки задачи, |

| | | | |
|--|--|--|--|
| | | | проведением эксперимента работоспособности системы |
|--|--|--|--|

2. УЧЕБНАЯ ПРОГРАММА

2.1. Цели и задачи дисциплины

- Изучить основы параллельного программирования и архитектуры.
- Понять принципы и модели параллельного выполнения.
- Овладеть методами разработки параллельных алгоритмов и программ.
- Научиться использовать современные инструменты и библиотеки для параллельного программирования.
- Развить навыки анализа производительности и оптимизации параллельных приложений.

2.2. Трудоемкость дисциплины и виды учебной работы (в академических часах и зачетных единицах)

| Виды учебной работы | Всего, в акад. часах | Распределение по семестрам | | | | | |
|---|----------------------------|----------------------------|-----------|------------|------------|----------|------------|
| | | I сем | II сем | III сем | IV сем. | V сем | VI сем. |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1. Общая трудоемкость изучения дисциплины по семестрам, в т. ч.: | 324 | | | | | | |
| 1.1. Аудиторные занятия, в т. ч.: | 154 | | | | | | |
| 1.1.1. Лекции | 58 | 34 | | | | | |
| 1.1.2. Практические занятия, в т. ч. | 96 | 72 | | | | | |
| 1.2. Самостоятельная работа, в т. ч.: | 107 | 74 | | | | | |
| 1.3. Другие методы и формы занятий | 63 | 36 | | | | | |
| Итоговый контроль (Экзамен, Зачет, диф. зачет - указать) | | Экзамен | Диф.зачет | | | | |

2.3. Содержание дисциплины

2.3.1. Тематический план и трудоемкость аудиторных занятий (модули, разделы дисциплины и виды занятий) по рабочему учебному плану

| Разделы и темы дисциплины | Всего (ак. часов) | Лекции (ак. часов) | Практ. занятия (ак. часов) | Семина- ры (ак. часов) | Лабор. (ак. часов) | Друг ие виды зая ний (ак. часо в) |
|---------------------------|----------------------|--------------------------|-------------------------------------|------------------------------|--------------------------|--|
| | | | | | | |

| 1 | 2=3+4+5+6+7 | 3 | 4 | 5 | 6 | 7 |
|---|-------------|-----------|-----------|---|---|---|
| Модуль 1. | | | | | | |
| Тема 1. Введение в параллельное программирование | 4 | 2 | 2 | | | |
| Тема 2. Архитектура параллельных систем | 4 | 2 | 2 | | | |
| Тема 3. Основы многопоточности | 4 | 2 | 2 | | | |
| Тема 4. Синхронизация и коммуникация | 4 | 2 | 2 | | | |
| Тема 5. Параллельные алгоритмы и структуры данных | 4 | 2 | 2 | | | |
| Тема 6. Инструменты и библиотеки для параллельного программирования | 4 | 2 | 2 | | | |
| Тема 7. Параллельное программирование на GPU | 4 | 2 | 2 | | | |
| Тема 8. Введение в CUDA и OpenCL | 4 | 2 | 2 | | | |
| Тема 9. Анализ и оптимизация производительности | 4 | 2 | 2 | | | |
| Тема 10. Параллельное программирование на Java | 4 | 2 | 2 | | | |
| Тема 11. Параллельное программирование на Python | 4 | 2 | 2 | | | |
| Тема 12. Распределенные системы | 4 | 2 | 2 | | | |
| Тема 13. Облачные вычисления | 4 | 2 | 2 | | | |
| Тема 14: Практическая работа и проекты | 4 | 2 | 2 | | | |
| ИТОГО | 64 | 32 | 32 | | | |

2.3.2. Краткое содержание разделов дисциплины в виде тематического плана

Тема 1: Введение в параллельное программирование

- Обзор параллельного программирования
- История и эволюция параллельных вычислений
- Основные концепции и термины

Тема 2: Архитектура параллельных систем

- Многоядерные и многопроцессорные системы
- Разделяемая память vs. распределенная память
- Параллельные модели: SIMD, MIMD, SPMD

Тема 3: Основы многопоточности

- Понятие потока и процесса
- Создание и управление потоками
- Проблемы многопоточности: гонки данных, взаимные блокировки

Тема 4: Синхронизация и коммуникация

- Механизмы синхронизации: мьютексы, семафоры, барьеры
- События и условные переменные
- Передача сообщений и коммуникационные модели

Тема 5: Параллельные алгоритмы и структуры данных

- Основы разработки параллельных алгоритмов
- Параллельные структуры данных: очереди, стек, деревья
- Примеры параллельных алгоритмов: сортировка, поиск

Тема 6: Инструменты и библиотеки для параллельного программирования

- Обзор инструментов и библиотек (OpenMP, MPI)
- Введение в OpenMP: директивы, клаузы, работа с циклами
- Введение в MPI: основы, передача сообщений, коллективные операции

Тема 7: Параллельное программирование на GPU

- Основы архитектуры GPU
- Введение в CUDA и OpenCL
- Программирование и оптимизация на GPU

Тема 8: Введение в CUDA и OpenCL

- Введение в CUDA и OpenCL
- Программирование и оптимизация на GPU

Тема 9: Анализ и оптимизация производительности

- Метрики производительности: скорость, масштабируемость, эффективность
- Профилирование и отладка параллельных программ
- Оптимизация и балансировка нагрузки

Тема 10: Параллельное программирование на Java

- Многопоточность в Java: Thread, Executor, ForkJoinPool

- Синхронизация и конкурирующие коллекции в Java
- Работа с библиотекой `java.util.concurrent`

Тема 11: Параллельное программирование на Python

- Многопоточность в Python: `threading`, `multiprocessing`
- Асинхронное программирование: `asyncio`
- Использование библиотек для параллельных вычислений (например, `concurrent.futures`)

Тема 12: Распределенные системы и облачные вычисления

- Основы распределенных систем

Тема 13: Облачные вычисления

- Введение в облачные платформы и сервисы
- Разработка распределенных приложений

Тема 14: Практическая работа и проекты

- Практическое применение изученных концепций
- Разработка собственного параллельного проекта
- Презентация и обсуждение проектов

2.3.3. Краткое содержание семинарских/практических занятий/лабораторного практикума

Решение задач согласно пройденной на лекционном занятии темы.

2.3.4. Материально-техническое обеспечение дисциплины

Компьютеры с интернет-браузером.

2.4. Модульная структура дисциплины с распределением весов по формам контролей

| | | | | | | | | |
|--|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| | $\Sigma = 1$ | $\Sigma = 1$ | $\Sigma = 1$ | $\Sigma = 1$ | $\Sigma = 1$ | $\Sigma = 1$ | $\Sigma = 1$ | $\Sigma = 1$ |
|--|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|

3. Теоретический блок

3.1. Материалы по теоретической части курса

3.1.1. Брайан Керниган, Деннис Ритчи, Язык программирования Си, 2-е издание

Бьерн Страуструп, Язык программирования C++, 2-е издание

Герберт Шилдт, C++. Базовый курс

3.1.2. Учебное пособие:

«Ալգորիթմներ և ալգորիթմական լեզուներ» Գործնական պարապմունքների մեթոդական ձեռնարկ

[https://rau.am/uploads/post/editor_image/C_C%20practice%20\(1\)_1717589821.pdf](https://rau.am/uploads/post/editor_image/C_C%20practice%20(1)_1717589821.pdf)

4. Фонды оценочных средств.

4.1. Материалы по практической части курса

4.1.1. Учебно-методические пособия;

Մարգարիտի Ա., Մարգարիտի Մ.

4.2. Вопросы и задания для самостоятельной работы студентов

В системе ejudge

4.3. Перечень экзаменационных вопросов

1. Представление целых и вещественных чисел в двоичной форме. Прямой, обратный и дополнительный код. IEEE-754 формат. Порядок представления байтов (big and little endian). Размеры типов в языке С. Знаковые и беззнаковые числа. Приведение типов.

Литература:

- Книга Computer Systems Programmers perspective, вторая глава.
- Книга Зубков С. В, первая глава
- Книга В.И. Юров: глава 4, глава 17

2. Процессор x86, регистры. Формат данных. Пересылка данных (mov). Регистр eflags. Арифметические и логические операции. Сдвиги. Переполнение. Флаги OF, CF, ZF. Отображение из С в ассемблер и наоборот.

Литература:

- Книга Computer Systems Programmers perspective: 3.1 - 3.6

- Книга В.И. Юров: глава 7, 8, 9

3. Флаги OF, CF, ZF, SF. Команды передачи управления. Инструкция jmp и условный переход. Представление операторов условного перехода языка Си в ассемблере.

Оператор switch.

Литература:

- Книга Computer Systems Programmers perspective: 3.1 - 3.6

- Intel reference manual

- Книга В.И. Юров: глава 10

4. Организация циклов. Инструкция loop. Представление операторов цикла языка Си в ассемблере: do-while, while, for. Инструкция условной пересылки (Conditional Move Instructions)

Литература:

- Книга Computer Systems Programmers perspective: 3.1 - 3.6

- Книга В.И. Юров: глава 10

5. Представление программы в памяти. Стек и локальные переменные.

Расположение глобальных и статических переменных. Организация вызова функций. Передача аргументов, адрес возврата. Фрейм стека. Различные соглашения о вызовах.

Литература:

- Книга Computer Systems Programmers perspective: 3.7

- Книга Зубков С. В, 5.2, 5.3

- Книга В.И. Юров: глава 15

6. Массивы и указатели. Представление одномерных и двумерных массивов. Структуры данных в ассемблере.

Литература:

- Книга Computer Systems Programmers perspective: 3.1 - 3.6

- Книга В.И. Юров: глава 13

7. Процессор x87. Организация процессора и его регистры. Основные команды процессора x87.

Литература:

- Книга В.И. Юров: Глава 17

8. RISC-V

Литература:

- <https://riscv.org/wp-content/uploads/2017/05/riscv-spec-v2.2.pdf>
- <https://github.com/riscv/riscv-bitmanip/releases/download/1.0.0/bitmanip-1.0.0.pdf>