Приложение 4

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Нижегородский государственный университет
им. Н.И. Лобачевского»

Институт информационных технологий, математики и механики

УТВЕРЖДЕНО

решением Ученого совета ННГУ

протокол № 10 от 02.12.2024 г.

Рабочая программа дисциплины

Оптимизация производительности программ

Уровень высшего образования
Бакалавриат

Направление подготовки / специальность
02.03.02 - Фундаментальная информатика и информационные технологии

Направленность образовательной программы
Сопряженная разработка программного и аппаратного обеспечения

Форма обучения
очная

г. Нижний Новгород

2025 год начала подготовки

1. Место дисциплины в структуре ОПОП

Дисциплина Б1.В.12.10 Оптимизация производительности программ относится к части, формируемой участниками образовательных отношений образовательной программы.

2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	**Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции**		**Наименование оценочного средства**	
	Индикатор достижения компетенции (код, содержание индикатора)	**Результаты обучения по дисциплине**	**Для текущего контроля успеваемости**	**Для промежуточной аттестации**
ПК-4: Способен проектировать программное обеспечение	*ПК-4.1: Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения* *ПК-4.2: Знает методы и средства проектирования программного обеспечения* *ПК-4.3: Знает методы и средства проектирования баз данных* *ПК-4.4: Умеет использовать существующие типовые решения и шаблоны проектирования программного обеспечения* *ПК-4.5: Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных*	*ПК-4.1:* *Знать: основные шаблоны проектирования (MVC, Singleton, Factory и др.), популярные библиотеки (STL, NumPy, Pandas) и принципы ООП.* *Уметь: выбирать и интегрировать типовые решения и библиотеки в проекты, адаптировать шаблоны под задачи.* *Владеть: инструментами управления зависимостями (Maven, pip, npm) и репозиториями (GitHub, PyPI).* *ПК-4.2:* *Знать: этапы разработки ПО (Agile, Scrum), методы спецификации требований (Use Case, User Story) и нотации моделирования (UML, BPMN).* *Уметь: проектировать архитектуру ПО, составлять спецификации и диаграммы UML, оценивать качество проектных решений.* *Владеть: инструментами моделирования (PlantUML, Draw.io) и коллективной работы (Confluence).* *ПК-4.3:* *Знать: модели данных*	*Задания* *Собеседование*	*Зачёт:* *Задачи*

		(реляционные, NoSQL), принципы нормализации (1NF–5NF), язык SQL и методы оптимизации запросов. *Уметь: проектировать схемы БД, нормализовать структуры, составлять SQL-запросы и выбирать СУБД под задачи проекта.* *Владеть: СУБД (MySQL, PostgreSQL, MongoDB), инструментами проектирования (MySQL Workbench) и методами оптимизации запросов.* *ПК-4.4:* *Знать: классические шаблоны проектирования (GoF), библиотеки компонентов (для аутентификации, логирования) и фреймворки.* *Уметь: внедрять и модифицировать типовые решения в коде, комбинировать шаблоны для сложных сценариев, документировать их использование.* *Владеть: навыками реализации шаблонов (Singleton, Adapter и др.), инструментами генерации кода и тестирования решений.* *ПК-4.5:* *Знать: методы анализа производительности, техники оптимизации (векторизация, параллелизм), инструменты профилирования (Intel VTune, Intel Advisor) и компиляторы (Intel C++ Compiler).* *Уметь: анализировать код на узкие места, оптимизировать алгоритмы и структуры данных, векторизовать циклы, балансировать нагрузку в вычислениях.* *Владеть: практическим применением Intel VTune и Intel Advisor, настройкой компиляции в Visual Studio,*		

		техниками оптимизации памяти и вычислений.		

3. Структура и содержание дисциплины

3.1 Трудоемкость дисциплины

	очная
Общая трудоемкость, з.е.	**2**
Часов по учебному плану	**72**
в том числе	
аудиторные занятия (контактная работа):	
- занятия лекционного типа	**32**
- занятия семинарского типа (практические занятия / лабораторные работы)	**0**
- КСР	**1**
самостоятельная работа	**39**
Промежуточная аттестация	**0** **Зачёт**

3.2. Содержание дисциплины

(структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий)

Наименование разделов и тем дисциплины	Всего (часы)	в том числе			
		Контактная работа (работа во взаимодействии с преподавателем), часы из них			Самостоятельная работа обучающегося, часы
		Занятия лекционного типа	Занятия семинарского типа (практические занятия/лабораторные работы), часы	Всего	
	очно	очно	очно	очно	очно
Раздел I. Введение в предмет.	20	10	0	10	10
Раздел II. Оптимизация программ: алгоритмы и структуры данных .	20	10	0	10	10
Раздел III. Программная оптимизация на примерах .	31	12	0	12	19
Аттестация	0				
КСР	1			1	
Итого	72	32	0	33	39

Содержание разделов и тем дисциплины

Раздел I. Введение в предмет.

1.1. Анализ производительности и оптимизация программ: введение в предмет (суть, цели, задачи, основные принципы, критерии, примеры, структура курса).

1.2. Архитектурные механизмы, влияющие на производительность. Уровни параллелизма.
1.3. Основные метрики, методика их сбора и анализа.
1.4. Практическое использование Intel C++ Compiler в среде разработки Microsoft Visual Studio. Компиляция и сборка из командной строки.
Раздел II. Оптимизация программ: алгоритмы и структуры данных.
2.1. Алгоритмическая оптимизация. Вычислительная сложность, практические аспекты.
2.2. Алгоритмическая оптимизация при реализации алгоритмов сортировки.
2.3. Оптимизация структур данных при работе с разреженными матрицами.
Раздел III. Программная оптимизация на примерах.
3.1. Векторизация циклов. Общие принципы и использование компилятора.
3.1.5. Инструменты для анализа производительности Intel VTune и Intel Advisor. Общее введение.
3.2. Векторизация циклов. Использование транслятора Intel Compiler и помощника Intel Advisor.
3.3. Рациональное использование иерархии памяти на примере задачи об умножении матриц. Использование профилировщика Intel VTune.
3.4. Оптимизация вычислений, аспекты параллелизма, балансировка нагрузки. Задача о вычислении простых чисел.
3.5. Пошаговая оптимизация программ в практических приложениях: вычисление формулы Блэка-Шоулса (векторизация, параллелизм, понижение точности, эквивалентные вычисления, NUMA).
3.6. Пошаговая оптимизация программ в практических приложениях: численное интегрирование (векторизация, параллелизм, работа с памятью + MKL).

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа обучающихся включает в себя подготовку к контрольным вопросам и заданиям для текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведенным в п. 5.

- Гергель В.П. Курс «Основы параллельных вычислений». http://www.intuit.ru/studies/courses/1091/293/info
- Гергель В.П. Курс «Теория и практика параллельных вычислений». https://intuit.ru/studies/courses/1156/190/info
- Немнюгин С. Курс «Основы параллельного программирования с использованием MPI». http://www.intuit.ru/studies/courses/1090/294/info
- Бахтин В. Курс «Параллельное программирование с OpenMP». http://www.intuit.ru/studies/courses/1111/295/info
- Баркалов К.А. и др. Курс «Intel Parallel Programming Professional» https://intuit.ru/studies/courses/4447/983/info
- Ануфриенко А., Идрисов Р. Курс «Введение в оптимизацию производительности приложений с использованием компиляторов Интел» https://intuit.ru/studies/courses/707/563/info

5. Фонд оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю)

5.1 Типовые задания, необходимые для оценки результатов обучения при проведении текущего контроля успеваемости с указанием критериев их оценивания:

5.1.1 Типовые задания (оценочное средство - Задания) для оценки сформированности компетенции ПК-4:

1) Лабораторная работа (проект) «Умножение плотных матриц. Блочная схема, алгоритм Фокса»

2) Лабораторная работа (проект) «Умножение плотных матриц. Блочная схема, алгоритм Фокса. Реализовать и использовать виртуальную топологию решетка»

3) Лабораторная работа (проект) «Умножение плотных матриц. DNS-алгоритм»

4) Лабораторная работа (проект) «Умножение плотных матриц. Алгоритм Штрассена»

5) Лабораторная работа (проект) «Умножение разреженных матриц. Формат хранения матрицы – координатный»

6) Лабораторная работа (проект) «Умножение разреженных матриц. Формат хранения матрицы – строковый (CRS)»

7) Лабораторная работа (проект) «Умножение разреженных матриц. Формат хранения матрицы – столбцовый (CCS)»

8) Лабораторная работа (проект) «Решение систем линейных уравнений метод сопряженных градиентов»

9) Лабораторная работа (проект) «Метод циклической редукции для трехдиагональных систем линейных уравнений»

10) Лабораторная работа (проект) «Поразрядная сортировка LSD Radix Sort для целых чисел»

11) Лабораторная работа (проект) «Поразрядная сортировка LSD Radix Sort для вещественных чисел (тип double)»

12) Лабораторная работа (проект) «Поразрядная сортировка MSD Radix Sort для целых чисел»

13) Лабораторная работа (проект) «Поразрядная сортировка MSD Radix Sort для вещественных чисел (тип double)»

14) Лабораторная работа (проект) «Поиск кратчайших путей из одной вершины (алгоритм Дейкстры)»

15) Лабораторная работа (проект) «Поиск кратчайших путей из одной вершины (алгоритм Мура)»

16) Лабораторная работа (проект) «Поиск кратчайших путей из одной вершины для разреженных графов (алгоритм Джонсона)»

17) Лабораторная работа (проект) «Алгоритм глобального поиска для одномерных задач оптимизации. Распараллеливание путем разделения области поиска»

18) Лабораторная работа (проект) «Алгоритм глобального поиска для одномерных задач оптимизации. Распараллеливание по характеристикам»

19) Лабораторная работа (проект) «Многошаговая схема решения двумерных задач глобальной оптимизации. Распараллеливание путем разделения области поиска»

20) Лабораторная работа (проект) «Многошаговая схема решения двумерных задач глобальной оптимизации. Распараллеливание по характеристикам»

Во всех лабораторных работах требуется разработать параллельную программу, решающую поставленную задачу указанным алгоритмом. При отсутствии явных указаний считается, что данные в

задаче имеют тип с плавающей запятой двойной точности. Оценивание результатов выполняется в смысле а) работоспособности программы – малое отклонение результата от эталона; б) эффективности программы – ускорение параллельной версии должно быть не менее указанного преподавателем значения.

Программная реализация выполняется на языках C или C++ с использованием технологий OpenMP и библиотеки TBB (весенний семестр).

Критерии оценивания (оценочное средство - Задания)

Оценка	Критерии оценивания
зачтено	Выполнена основная часть задания, возможно с незначительными недочетами.
не зачтено	Выполнено менее половины задания, есть существенные недочеты.

5.1.2 Типовые задания (оценочное средство - Собеседование) для оценки сформированности компетенции ПК-4:

1. Архитектурные механизмы, влияющие на производительность программ.
2. Цели и задачи оптимизации программ.
3. Критерии оптимизации программ
4. Алгоритмическая оптимизация и ее роль. Примеры
5. Оптимизация структур данных и ее роль. Примеры.
6. Программная оптимизация и ее роль. Примеры.
7. Векторизация и ее роль. Примеры.
8. Оптимизация параллельных программ. Примеры.
9. Оптимизация работы с памятью. Примеры.

Критерии оценивания (оценочное средство - Собеседование)

Оценка	Критерии оценивания
зачтено	Студент дал развернутый ответ на все вопросы без существенных ошибок.
не зачтено	При ответе студент допускает грубые ошибки в основном материале.

5.2. Описание шкал оценивания результатов обучения по дисциплине при промежуточной аттестации

Шкала оценивания сформированности компетенций

Уровень сформированности компетенций (индикатора достижения компетенций)	**плохо**	**неудовлетворительно**	**удовлетворительно**	**хорошо**	**очень хорошо**	**отлично**	**превосходно**
	не зачтено		**зачтено**				
Знания	Отсутствие знаний теоретического материала. Невозможность оценить полноту знаний вследствие отказа обучающегося от ответа	Уровень знаний ниже минимальных требований. Имели место грубые ошибки	Минимально допустимый уровень знаний. Допущено много негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько несущественных ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Ошибок нет.	Уровень знаний в объеме, превышающем программу подготовки.
Умения	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки	Продемонстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме	Продемонстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания в полном объеме, но некоторые с недочетами	Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания в полном объеме, но некоторые с недочетами.	Продемонстрированы все основные умения. Решены все основные задачи с отдельными несущественными недочетами, выполнены все задания в полном объеме	Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме без недочетов
Навыки	Отсутствие базовых навыков. Невозможность оценить наличие навыков вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки	Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами	Продемонстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продемонстрированы базовые навыки при решении стандартных задач без ошибок и недочетов	Продемонстрированы навыки при решении нестандартных задач без ошибок и недочетов	Продемонстрирован творческий подход к решению нестандартных задач

Шкала оценивания при промежуточной аттестации

Оценка		**Уровень подготовки**
зачтено	**превосходно**	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно», продемонстрированы знания, умения, владения по соответствующим компетенциям на уровне выше предусмотренного программой

	отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично».
	очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо»
	хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо».
	удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	**неудовлетворительно**	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно».
	плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

5.3 Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения на промежуточной аттестации с указанием критериев их оценивания:

5.3.1 Типовые задания (оценочное средство - Задачи) для оценки сформированности компетенции ПК-4

1. Взаимодействие и взаимоисключение потоков, синхронизация потоков. Проблемы взаимоблокировки потоков.
2. Принципы организации параллелизма с использованием технологии OpenMP.
3. Распределение вычислений с использованием технологии OpenMP.
4. Синхронизация с использованием технологии OpenMP.
5. Параллельное программирование для систем с общей памятью на основе технологии Intel Threading Building Blocks (TBB).
6. Библиотека TBB. Распараллеливание циклов.
7. Библиотека TBB. Распараллеливание циклов с редукцией.
8. Библиотека TBB. Механизм задач.
9. Библиотека TBB. Распараллеливание рекурсивных алгоритмов.

Критерии оценивания (оценочное средство - Задачи)

Оценка	Критерии оценивания
зачтено	Выполнены все или большая часть этапов решения задачи или задача решена с незначительными недочетами. Результаты работы представлены преподавателю в срок.
не зачтено	Выполнены не все практические задания или выполнены не в полном объеме (представлено не полное описание этапов выполнения заданий, получен неверный ответ, результаты работы не представлены преподавателю).

6. Учебно-методическое и информационное обеспечение дисциплины (модуля)

Основная литература:

1. Теория и практика параллельных вычислений / Гергель В.П. - Москва : ИНТУИТ, 2016., https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=663423&idb=0.
2. Славин О. А. Оптимизация быстродействия программного обеспечения на современных процессорных архитектурах : учебное пособие / Славин О. А. - Москва : РТУ МИРЭА, 2020. - 99 с. - Библиогр.: доступна в карточке книги, на сайте ЭБС Лань. - Книга из коллекции РТУ МИРЭА - Информатика. - Текст : электронный., https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=749988&idb=0.
3. Земляная Е. В. Введение в параллельное программирование на основе технологий MPI и OpenMP : учебное пособие / Земляная Е. В.,Башашин М. В. - Дубна : Государственный университет «Дубна», 2023. - 101 с. - Книга из коллекции Государственный университет «Дубна» - Информатика. - ISBN 978-5-89847-696-0. - Текст : электронный., https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=887262&idb=0.
4. Юрина Т. А. Параллельное программирование / Юрина Т. А. - Омск : СибАДИ, 2023. - 106 с. - Книга из коллекции СибАДИ - Информатика. - Текст : электронный., https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=897489&idb=0.

Дополнительная литература:

1. Параллельное программирование с использованием OpenMP / Левин М.П. - Москва : ИНТУИТ, 2016., https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=663520&idb=0.
2. Малявко Александр Антонович. Параллельное программирование на основе технологий openmp, cuda, opencl, mpi : учебник для вузов / А. А. Малявко. - 3-е изд. - Москва : Юрайт, 2026. - 135 с. - (Высшее образование). - URL: https://urait.ru/bcode/585284 (дата обращения: 24.01.2026). - ISBN 978-5-534-14116-0 : 549.00. - Текст : электронный., https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=993278&idb=0.
3. Уильямс Э. Параллельное программирование на C++ в действии. Практика разработки многопоточных программ : монография / Уильямс Э. - Москва : ДМК-пресс, 2023. - 674 с. - ISBN 978-5-89818-319-6. - Текст : электронный., https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=878880&idb=0.

Программное обеспечение и Интернет-ресурсы (в соответствии с содержанием дисциплины):

- https://intuit.ru/

7. Материально-техническое обеспечение дисциплины (модуля)

Учебные аудитории для проведения учебных занятий, предусмотренных образовательной программой, оснащены мультимедийным оборудованием (проектор, экран), техническими средствами обучения, компьютерами.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ по направлению подготовки/специальности 02.03.02 - Фундаментальная информатика и информационные технологии.

Автор(ы): Мееров Иосиф Борисович, кандидат технических наук, доцент.

Заведующий кафедрой: Мееров Иосиф Борисович, кандидат технических наук.

Программа одобрена на заседании методической комиссии от 02.12.2024, протокол № 5.