

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
(факультет / институт / филиал)

УТВЕРЖДЕНО

решением Ученого совета ННГУ

протокол №13 от 30.11.2022 г.

Рабочая программа дисциплины

Оптимизация производительности программ
(наименование дисциплины (модуля))

Уровень высшего образования
Бакалавриат

(бакалавриат / магистратура / специалитет)

Направление подготовки / специальность
02.03.02 Фундаментальная информатика и информационные технологии
(указывается код и наименование направления подготовки / специальности)

Направленность образовательной программы
Инженерия программного обеспечения
(указывается профиль / магистерская программа / специализация)

Форма обучения

Очная

(очная / очно-заочная / заочная)

Нижегород

2023 год

1. Место дисциплины в структуре ООП

Дисциплина Б1.В.ДВ.02.05 Оптимизация производительности программ относится к части ООП направления подготовки 02.03.02 «Фундаментальная информатика и информационные технологии», формируемой участниками образовательных отношений.

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	
ПК-4.: Способен проектировать программное обеспечение	ПК-4.1.: Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения	Знать методику анализа производительности и оптимизации программ	<i>Собеседование</i>
	ПК-4.2.: Знает методы и средства проектирования программного обеспечения	Знать методы и программные средства анализа производительности и оптимизации программ	<i>Собеседование</i>
	ПК-4.3.: Знает методы и средства проектирования баз данных	Знать методы и средства проектирования баз данных	<i>Задание</i>
	ПК-4.4.: Умеет использовать существующие типовые решения и шаблоны проектирования программного обеспечения	Уметь применять на практике методы и программный инструментарий для анализа производительности и оптимизации программ	<i>Задание</i>
	ПК-4.5.: Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных	Уметь применять методы и средства проектирования программного обеспечения, структур данных, баз данных для оптимизации производительности программ	<i>Задание</i>

3. Структура и содержание дисциплины

3.1. Трудоемкость дисциплины

	Очная форма обучения
Общая трудоемкость	3 ЗЕТ
Часов по учебному плану	108
в том числе	
аудиторные занятия (контактная работа):	33
- занятия лекционного типа	16
- занятия семинарского типа	0
- занятия лабораторного типа	16
- текущий контроль (КСР)	1
самостоятельная работа	75
Промежуточная аттестация – зачет	

3.2. Содержание дисциплины

Наименование и краткое содержание разделов и тем дисциплины	Всего (часы)	В том числе				Самостоятельная работа обучающегося, часы
		Контактная работа (работа во взаимодействии с преподавателем), часы. Из них				
		Занятия лекционного типа	Занятия семинарского типа	Занятия лабораторного типа	Всего	
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.	30	5		5	10	20
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	30	5		5	10	20
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	47	6		6	12	35

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)						
Текущий контроль (КСР)	1				1	
Промежуточная аттестация – зачет						
Итого	108	16	0	16	33	75

Текущий контроль успеваемости реализуется в формах опросов на занятиях лабораторного типа.

Промежуточная аттестация проходит в традиционной форме (зачет).

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Виды самостоятельной работы студентов

Самостоятельная работа студентов подразделяется на следующие категории:

- Изучение учебной литературы (см. перечень образовательных материалов).
- Выполнение практических работ на следующие темы: настройка среды разработки и проекта в Microsoft Visual Studio для сборки программ, применение современных программных инструментов для анализа производительности и оптимизации программ (оптимизирующая система, профилировщики, математические библиотеки), разработка, отладка, анализ производительности и оптимизация программ, решающих типовые задачи информатики и вычислительной математики (сортировки и поиски, линейная алгебра – умножение плотных и разреженных матриц, решение систем линейных уравнений, метод Монте-Карло, численное интегрирование и др.).

Образовательные материалы для самостоятельной работы студентов

- 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».
<http://www.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>

Контрольные вопросы и задания для проведения текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведены в п. 5.2.

5. Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю), включающий:

5.1.Описание шкал оценивания результатов обучения по дисциплине

Уровень сформированности компетенций (индикатора достижения компетенций)	Шкала оценивания сформированности компетенций						
	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно
	Не зачтено		Зачтено				
<u>Знания</u>	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 незначительных ошибок	Уровень знаний в объеме, соответствующем программе подготовки, без ошибок.	Уровень знаний в объеме, превышающем программу подготовки.
<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки.	Продemonстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания,	Продemonстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания, в	Продemonстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме, но	Продemonстрированы все основные умения, решены все основные задачи с отдельными незначительными недочетами,	Продemonстрированы все основные умения, решены все основные задачи. Выполнены все задания, в полном

	обучающего-ся от ответа		но не в полном объеме.	полном объеме, но некоторые с недочетами.	некоторые с недочетами.	выполнены все задания в полном объеме.	объеме без недочетов
<u>Навыки</u>	Отсутствие владения материалом. 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 оценки при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	Превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно»
	Отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично», при этом хотя бы одна компетенция сформирована на уровне «отлично»
	Очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо», при этом хотя бы одна компетенция сформирована на уровне «очень хорошо»
	Хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо», при этом хотя бы одна компетенция сформирована на уровне «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», ни одна из компетенций не сформирована на уровне «плохо»
	Плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

5.2. Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения

5.2.1 Контрольные вопросы для собеседования

Вопрос	Код компетенции (согласно РПД)
1. Архитектурные механизмы, влияющие на производительность программ.	ПК-4
2. Цели и задачи оптимизации программ.	ПК-4
3. Критерии оптимизации программ.	ПК-4
4. Алгоритмическая оптимизация и ее роль. Примеры.	ПК-4

5.	Оптимизация структур данных и ее роль. Примеры.	ПК-4
6.	Программная оптимизация и ее роль. Примеры.	ПК-4
7.	Векторизация и ее роль. Примеры.	ПК-4
8.	Оптимизация параллельных программ. Примеры.	ПК-4
9.	Оптимизация работы с памятью. Примеры.	ПК-4

5.2.2. Типовые задания для оценки сформированности компетенции ПК-4

- 1) Умножение плотных матриц. Блочный алгоритм.
- 2) Алгоритмы умножения разреженных матриц.
- 3) Алгоритм быстрой сортировки и его оптимизация по скорости.

5.2.3. Типовые задачи, выносимые на зачет, для оценки сформированности компетенции ПК-4

- 1) Разработка примера программы, в которой наблюдается проблема производительности «отсутствие векторизации кода» с последующим устранением описанной проблемы.
- 2) Разработка примера программы, в которой наблюдается проблема производительности «неэффективное распараллеливание» (дисбаланс вычислительной нагрузки) с последующим устранением описанной проблемы.

6. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

- 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».
<http://www.intuit.ru/studies/courses/1156/190/info>

б) дополнительная литература:

- 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>

7. Материально-техническое обеспечение дисциплины

Помещения представляют собой учебные аудитории для проведения учебных занятий, предусмотренных программой, оснащенные оборудованием и техническими средствами обучения.

Помещения для самостоятельной работы обучающихся оснащены компьютерной

техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду ННГУ.

Программа составлена в соответствии с требованиями ОС ННГУ 02.03.02 Фундаментальная информатика и информационные технологии.

Автор: к.т.н., доцент кафедры МОСТ ИТММ, доцент Мееров И.Б.

Рецензент: д.т.н., профессор НГТУ им. Р.Е. Алексеева Ломакина Л.С.

Заведующий кафедрой МОСТ: д.ф.-м.н. Стронгин Р.Г.

Программа одобрена на заседании методической комиссии института информационных технологий, математики и механики 30.11.2022 года, протокол № 3