



Федеральное государственное бюджетное образовательное учреждение высшего образования
«ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ ДВАЖДЫ ГЕРОЯ
СОВЕТСКОГО СОЮЗА, ЛЕТЧИКА-КОСМОНАВТА А.А. ЛЕОНОВА»

УТВЕРЖДАЮ

И.о. проректора

А.В. Троицкий

«__» _____ 2023г.

***ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И ТЕХНОЛОГИЙ
КАФЕДРА МАТЕМАТИКИ И ЕСТЕСТВЕННОНАУЧНЫХ ДИСЦИПЛИН***

РАБОЧАЯ ПРОГРАММА

ДИСЦИПЛИНЫ

«Технологии параллельного программирования»

Направление подготовки: 01.03.02. Прикладная математика и информатика

Профиль: Программирование. Математическое моделирование

Уровень высшего образования: бакалавриат

Форма обучения: очная

Королев
2023

Рабочая программа является составной частью основной профессиональной образовательной программы и проходит рецензирование со стороны работодателей в составе профессиональной образовательной программы. Рабочая программа актуализируется и корректируется ежегодно.

Автор: Вилицов В.Я. Рабочая программа дисциплины: Технологии параллельного программирования. – Королев МО: «Технологический Университет», 2023г.

Рецензент: д.э.н. проф. Мищенко А.В.

Рабочая программа составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования по направлению подготовки бакалавров 01.03.02 «Прикладная математика и информатика» и Учебного плана, утвержденного Ученым советом Университета. Протокол № 9 от 11.04.2023 г.

Рабочая программа рассмотрена и одобрена на заседании кафедры:

Заведующий кафедрой (ФИО, ученая степень, звание, подпись)	Бугай И.В. к.т.н., доцент 			
Год утверждения (переподтверждения)	2023			
Номер и дата протокола заседания кафедры	№8 от 15.03.2023			

Рабочая программа согласована:

Руководитель ОПОП ВО _____  И.В. Бугай, к.т.н., доцент

Рабочая программа рекомендована на заседании УМС:

Год утверждения (переподтверждения)	2023			
Номер и дата протокола заседания УМС	№5 от 11.04.2023			

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОПОП ВО

Целью изучения дисциплины является:

1. формирование способности к восприятию, обобщению и анализу информации, необходимой для оценки вариантов, обеспечения и поддержки принятия эффективных конструкторских и управленческих решений;
2. освоение необходимого инструментария, применяемого при оценке вариантов, выборе эффективных конструкторских и управленческих решений;
3. формирование готовности применять методы математического анализа и моделирования сложных систем и процессов в профессиональной деятельности.

В процессе обучения студент приобретает и совершенствует следующие компетенции:

профессиональные компетенции (ПК):

- Способность использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ (ПК-3);
- Способность учитывать знания проблем и тенденций развития рынка ПО в профессиональной деятельности (ПК-4).

Основными **задачами** дисциплины являются:

1. Освоение студентами теоретических методов дисциплины, применяемых при оценке вариантов и выборе эффективных конструкторских и управленческих решений;
2. Получение студентами умений и навыков, применяемых для решения практических задач оценки вариантов и выбора эффективных конструкторских и управленческих решений.

Показатель освоения компетенции отражают следующие индикаторы:

Необходимые знания:

- Знать методы и приемы формализации задач
- Знать методы и средства проектирования программного обеспечения, программных интерфейсов
- Знать возможности существующей программно-технической архитектуры
- Знать возможности современных и перспективных средств разработки программных продуктов, технических средств
- Знать методологии разработки программного обеспечения и технологии программирования
- Знать методологии и технологии проектирования и использования баз данных

Необходимые умения:

- Уметь выбирать средства и вырабатывать реализации требований к программному обеспечению
- Уметь проводить оценку и обоснование рекомендуемых решений

- Уметь проводить анализ исполнения требований
- Уметь вырабатывать варианты реализации требований;

Трудовые действия:

- Владеть методами и средствами проектирования баз данных.
- Проводить оценку и обоснование рекомендуемых решений
- Осуществлять коммуникации с заинтересованными сторонами.

2. Место дисциплины в структуре ОПОП ВО

Дисциплина «Технологии параллельного программирования» относится к дисциплинам по выбору части, формируемой участниками образовательных отношений, основной профессиональной образовательной программы подготовки бакалавров по направлению подготовки 01.03.02. «Прикладная математика и информатика».

Дисциплина базируется на ранее изученных дисциплинах: «Информатика» (школьный курс) и «Математика» (школьный курс).

Знания и компетенции, полученные при освоении дисциплины, являются базовыми для изучения дисциплин: «Среды и методы имитационного моделирования и планирование вычислительного эксперимента», «Математическое моделирование технических систем и процессов», «Программирование микропроцессоров», «Математические методы и среды разработки экспертных систем», «Системы поддержки принятия решений», «Технологии и системы коллективной разработки программ» и др., и выполнения выпускной квалификационной работы бакалавра.

3. Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины для студентов очной формы составляет 3 зачетных единицы, 108 часов. Практическая подготовка обучающихся составляет 16 часов.

Таблица 1

Виды занятий	Всего часов	Семестр первый	Семестр
ОЧНАЯ ФОРМА ОБУЧЕНИЯ			
Общая трудоемкость	108	108	
Аудиторные занятия	48	48	
Лекции (Л)	16	16	
Практические занятия (ПЗ)	16	16	
Лабораторные работы (ЛР)	16	16	
Практическая подготовка	16	16	
Самостоятельная работа	60	60	
Курсовые работы (проекты)	-	-	
Расчетно-графические работы	-	-	
Контрольная работа	+	+	
Текущий контроль знаний	Тест	Тест	
Вид итогового контроля	экзамен	экзамен	
ЗАОЧНАЯ ФОРМА НЕ ПРЕДУСМОТРЕНА УЧЕБНЫМ ПЛАНОМ			

4. Содержание дисциплины

4.1. Темы дисциплины и виды занятий

Таблица 2

Наименование тем	Лекции, час.	Практические занятия, час	Лабораторные занятия, час	Занятия в интерактивной форме, час	Практическая подготовка	Код компетенций
Тема 1. Анализ параллельных алгоритмов. Принципы разработки параллельных методов.	4	4	4	2	4	ПК-3
Тема 2. Распараллеливание матрично-векторных операций. Решение систем линейных алгебраических уравнений	6	6	6	2	6	ПК-3
Тема 3. Параллельные сортировки и алгоритмы на графах	6	6	6	4	6	ПК-4
Итого:	16	16	16	8	16	

4.2. Содержание тем дисциплины

Тема 1. Анализ параллельных алгоритмов. Принципы разработки параллельных методов.

- 1.1. Принципы разработки параллельных методов.
- 1.2. Модели оценки времени выполнения коммуникационных операций.
- 1.3. Вычисление констант моделей.

Тема 2. Распараллеливание матрично-векторных операций. Решение систем линейных алгебраических уравнений.

- 2.1. Распараллеливание умножения матрицы на вектор.
- 2.2. Распараллеливание умножения матриц.
- 2.3. Параллельные реализации метода Гаусса.

Тема 3. Параллельные сортировки и алгоритмы на графах.

- 3.1. Параллельные сортировки.
- 3.2. Генетические алгоритмы.
- 3.3. Поиск всех кратчайших путей.
- 3.4. Задача оптимального разделения графов.

5. Перечень учебно-методического обеспечения для самостоятельной работы по дисциплине

Методические указания для обучающихся по освоению дисциплины

6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине «Технологии параллельного программирования» приведен в Приложении 1 к настоящей рабочей программе.

7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Арыков С.Б. Параллельное программирование над общей памятью: OpenMP: [16+] / С.Б. Арыков, М.А. Городничев, Г.А. Щукин; Новосибирский государственный технический университет. – Новосибирск: Новосибирский государственный технический университет, 2019. – 95 с.: ил., табл. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=576119>.
2. Лебедев, А. С. Технология параллельного программирования: учебно-методическое пособие / А. С. Лебедев, Ш. Г. Магомедов. — Москва: РТУ МИР-ЭА, 2021. — 98 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/176524>

Дополнительная литература:

1. Стуколов С. В. Параллельное программирование. Практикум: учебное пособие / С. В. Стуколов. — Кемерово: КемГУ, 2020. — 273 с. — ISBN 978-5-8353-2723-2. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/173547>
2. Елесина, С. И. Технология параллельного программирования OpenMP: учебное пособие / С. И. Елесина. — Рязань: РГРТУ, 2021. — 48 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/220409>
3. Богачёв К. Ю. Основы параллельного программирования: учебное пособие / К. Ю. Богачёв. — 4-е изд. — Москва: Лаборатория знаний, 2020. — 345 с. — ISBN 978-5-00101-758-5. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/135516>.

8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

Интернет-ресурсы:

1. <http://biblioclub.ru/> - Университетская библиотека ONLINE.
2. <http://www.rucont.ru/> - Электронная библиотека Руконт.
3. <http://www.znaniium.com/> - Электронная библиотека издательства Инфра-М.
4. <http://e.lanbook.com/> - Электронная библиотека Лань.

5. <http://www.olap.ru/> - Сайт OLAP-технологий.
6. <https://ru.atlassian.com/software/jira/> - Система планирования работ проекта ПО.
7. <https://www.jetbrains.com/youtrack/> - Система планирования работ проекта ПО.
8. <https://ru.atlassian.com/software/confluence> - Система хранения проектной документации.
9. <https://git-scm.com/> - Распределенная система контроля версий ПО.
10. <https://ru.wikipedia.org/wiki/Subversion> - Централизованная система контроля версий ПО.
11. <https://jenkins-ci.org/> - Система непрерывной интеграции проекта ПО.
12. <https://www.docker.com/> - Система виртуализации окружения.

9. Методические указания для обучающихся по освоению дисциплины

Методические указания для обучающихся по освоению дисциплины приведены в Приложении 2 к настоящей рабочей программе.

10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

Перечень программного обеспечения: MSOffice

Информационные справочные системы: Электронные ресурсы образовательной среды Университета

11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Лекционные занятия:

- аудитория, оснащенная презентационной техникой (проектор, экран);
- доской для письма мелом или маркерами;
- комплект электронных презентаций/слайдов;
- комплект записей лекций для дистанционного обучения.

Практические занятия:

- аудитория, оснащенная мультимедийными средствами (проектор, ноутбук),
- демонстрационными материалами (наглядными пособиями);
- доской для письма мелом или фломастерами;

Прочее:

- рабочее место преподавателя, оснащенное компьютером с доступом в Интернет;
- рабочие места обучающихся, оснащенные компьютером с доступом в Интернет.

***ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И ТЕХНОЛОГИЙ
КАФЕДРА МАТЕМАТИКИ И ЕСТЕСТВЕННОНАУЧНЫХ ДИСЦИПЛИН***

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ
ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО
ДИСЦИПЛИНЕ
«Технологии параллельного программирования»**

Направление подготовки: 01.03.02 Прикладная математика и информатика

Профиль: Программирование, математическое моделирование

Уровень высшего образования: бакалавриат

Форма обучения: очная

Королев
2023

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

№ п/п	Индекс компетенции	Содержание компетенции (или ее части)*	Раздел дисциплины, обеспечивающий формирование компетенции (или ее части)	В результате изучения раздела дисциплины, обеспечивающего формирование компетенции (или ее части), обучающийся приобретает:		
				Необходимые знания	Необходимые умения	Трудовые действия
1.	ПК-3	Способность использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ	Тема 1. Анализ параллельных алгоритмов. Принципы разработки параллельных методов Тема 2. Распараллеливание матрично-векторных операций. Решение систем линейных алгебраических уравнений	Знать методы и приемы формализации задач Знать методы и средства проектирования программного обеспечения, программных интерфейсов	Уметь выбирать средства и вырабатывать реализации требований к программному обеспечению Уметь проводить оценку и обоснование рекомендуемых решений	Владеть методами и средствами проектирования баз данных.
2.	ПК-4	Способность учитывать знания проблем и тенденций развития рынка ПО в профессиональной деятельности	Тема 3. Параллельные сортировки и алгоритмы на графах	Знать возможности существующей программно-технической архитектуры Знать возможности современных и перспективных средств разработки программных продуктов, технических средств Знать методологии разработки программного обеспечения и технологии программирования Знать методологии и технологии проектирования и использования баз данных	Уметь проводить анализ исполнения требований Уметь вырабатывать варианты реализации требований	Проводить оценку и обоснование рекомендуемых решений Осуществлять коммуникации с заинтересованными сторонами

2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Код компетенции	Инструменты, оценивающие сформированность компетенции	Этапы и показатель оценивания компетенции	Шкала и критерии оценки
ПК-3, ПК-4	Тест	<p>А) полностью сформирована (компетенция освоена на высоком уровне) – 90% правильных ответов</p> <p>Б) частично сформирована:</p> <ul style="list-style-type: none"> •компетенция освоена на продвинутом уровне – 70% правильных ответов; •компетенция освоена на базовом уровне – от 51% правильных ответов; <p>В) не сформирована (компетенция не освоена) – менее 50% правильных ответов</p>	<p>Проводится письменно</p> <p>Время, отведенное на процедуру –30 мин.</p> <p>Неявка 0 баллов.</p> <p>Критерии оценки определяются процентным соотношением.</p> <p>Неудовлетворительно – менее 50% правильных ответов.</p> <p>Удовлетворительно – от 51% правильных ответов.</p> <p>Хорошо – от 70%.</p> <p>Отлично – от 90%.</p> <p>Максимальная оценка – 5 баллов.</p>
	Выполнение контрольной работы	<p>А) полностью сформирована (компетенция освоена на высоком уровне) – 5 баллов</p> <p>Б) частично сформирована:</p> <ul style="list-style-type: none"> •компетенция освоена на продвинутом уровне – 4 балла; •компетенция освоена на базовом уровне – 3 балла; <p>В) не сформирована (компетенция не освоена) – 2 и менее баллов</p>	<p>При определении сформированности компетенций критериями оценивания выступают методические рекомендации, разработанные по дисциплине для данного вида.</p>

3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

3.1. Типовые вопросы, выносимые на тестирование

1. Какая характеристика кластера оказывает наибольшее влияние на его вычислительную производительность?

- (!) Пропускная способность коммуникаций
- (?) Объем оперативной памяти
- (?) Объем дискового пространства

2. Для чего применяется тест LINPACK?

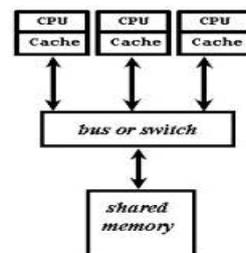
- (!) Для тестирования производительности одного вычислительного узла
- (?) Для тестирования производительности кластера
- (?) Для тестирования производительности пропускной способности сети
- (?) Для тестирования латентности сети

3. Для чего применяется тест SCALAPACK?

- (?) Для тестирования производительности одного вычислительного узла
- (!) Для тестирования производительности кластера
- (?) Для тестирования производительности пропускной способности сети
- (?) Для тестирования латентности сети

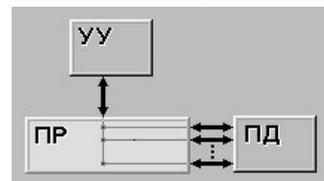
4. Какая архитектура изображена на рисунке?

- (?) Векторно-конвейерная
- (?) Массивно-параллельная
- (!) С общей памятью
- (?) Кластерная



5. К какой категории, согласно классификации Флинна, относится архитектура, изображенная на рисунке?

- (?) SISD
- (!) SIMD
- (?) MISD
- (?) MIMD



6. Что такое параллельный алгоритм?

- (?) Который выполняется одновременно на двух компьютерах.
- (!) Такой, который может быть реализован по частям на множестве различных вычислительных устройств с последующим объединением полученных результатов и получением корректного результата.

7. Может ли быть распараллелен метод Ньютона?

- (?) Да.
- (!) Нет.

8. Что такое мультипроцессорная система?

- (?) Это мультимодальная система.
- (!) Это вычислительная система, имеющая несколько процессоров.

9. Что такое вычислительная система с общей памятью?

- (?) В которой память конструктивно расположена в общем корпусе.
- (!) Имеющая несколько процессоров, пользующихся общим ресурсом памяти.

10. Что такое балансировка нагрузки в многопроцессорной системе?

- (?) Когда нагрузка балансирует на грани вычислительных возможностей.
- (!) Это диспетчеризация заданий таким образом, чтобы все процессоры были загружены примерно одинаково.

11. Что такое вычислительный кластер?

- (?) Это группа компьютеров, расположенных в локальной области.
- (!) Это группа компьютеров, объединенных высокоскоростными каналами связи, представляющая с точки зрения пользователя единый аппаратный ресурс.

12. Что такое вычислительные кластеры высокой доступности?

(?) Который практически всегда свободен.

(!) Они обеспечивают высокую доступность сервиса, предоставляемого кластером за счет наличия в кластере нескольких процессоров.

13. Что такое кластер с холодным резервом?

(?) У которого блок резервного питания всегда отключен.

(!) Это кластер, в котором один (например, из двух) процессор активен, а второй пассивен, но подключается при отказе первого.

14. Что такое кластер с горячим резервом?

(?) У которого блок резервного питания всегда включен.

(!) Это кластер, в котором все процессоры постоянно активны и при отказе одного в работу включается один из остальных свободных.

15. Что такое кластер с модульной избыточностью?

(?) В котором часть модулей функционально не востребованы.

(!) В них резервирование выполнено не на уровне процессоров, а на уровне модулей, которые включаются в работу при отказе других параллельных модулей.

16. Что такое программный вычислительный кластер?

(?) В котором выполняются программы.

(!) В отличие от аппаратных кластеров, программный является виртуальным и в нем процессоры образуют группу за счет применения специального интеграционного софта.

17. Что такое масштабируемость в параллельных алгоритмах?

(?) Когда размеры алгоритма можно без ущерба сократить.

(!) Это свойство требует гибкости программы по отношению к изменению числа процессоров (к их уменьшению или увеличению).

18. Что такое локальность в параллельных алгоритмах?

(?) Когда они выполняются в узкой группе задач.

(!) Это свойство характеризует необходимость того, чтобы доступ к локальным данным был более частым, чем к удаленным.

19. Что такое модульность в параллельных алгоритмах?

(?) Когда в каждом модуле имеются параллельные фрагменты.

(!) Это свойство отражает степень разложения сложных вычислительных объектов на более простые.

20. Что называют семафорами в параллельных вычислительных алгоритмах?

(?) Индикаторы того, что алгоритм используется.

(!) Это такое средство (переменная) управления процессами, которое обеспечивает их синхронизацию при обращении к разделяемым данным.

21. Что такое гипотеза Минского в параллельных вычислениях?

(?) Предположение о том, что распараллеливание возможно.

(!) Это утверждение: параллельные вычислительные системы, выполняющие последовательную программу под множеством исходных данных размера N , дают прирост производительности по крайней мере на показатель $1/\log(N)$.

22. В каких случаях следует прибегать к параллельным алгоритмам?

(?) Всегда.

(!) Когда все попытки создать быстрый последовательный алгоритм решения задачи оказались неудачными.

23. В чем заключается распараллеливание в модели параллелизма данных?

(?) Когда одни и те же данные используются разными подпрограммами.

(!) Когда одна и та же операция может быть применена к различным данным, а значит может быть выполнена в разных процессах (на разных процессорах).

24. В чем заключается распараллеливание в модели параллелизма задач?

(?) Когда параллельно выполняемые задачи могут быть объединены.

(!) Когда задача разбивается на множество подзадач, которые могут решаться разными процессорами независимо.

25. Что такое гонка данных?

(?) Это состязательная схема получения результата (данных) параллельными задачами.

(!) Это ситуация, когда несколько задач одновременно пытаются изменить общую область данных, конечное значение которой определяется самой быстрой задачей.

3.2 Тематика контрольных заданий, выполняемых с применением программных средств специального и общего назначения на компьютерах:

1. Методы анализа параллельных алгоритмов.
2. Принципы разработки параллельных методов.
3. Модели оценки времени выполнения коммуникационных операций.
4. Методы и подходы к вычислению констант моделей.
5. Методы распараллеливания умножения матрицы на вектор.
6. Анализ подходов к распределению элементов матрицы и вектора между процессорами вычислительной системы согласно выбранным базовым операциям при распараллеливании.
7. Варианты реализации алгоритмов для вычислительных систем с общей памятью.
8. Варианты реализации алгоритмов для вычислительных систем с распределенной памятью.
9. Варианты реализации алгоритмов для гибридных систем.
10. Вычисление ускорения и эффективности распараллеливания каждого алгоритмов.
11. Методы распараллеливания умножения матриц.
12. Анализ подходов к распределению элементов перемножаемых матриц между процессорами вычислительной системы согласно выбранным базовым операциям при распараллеливании.
13. Варианты реализации алгоритмов умножения матриц для вычислительных систем с общей памятью.
14. Варианты реализации алгоритмов умножения матриц для вычислительных систем с распределенной памятью.
15. Варианты реализации алгоритмов умножения матриц для гибридных систем.

16. Методы вычисления ускорения и эффективности распараллеливания алгоритмов умножения матриц.
17. Параллельные реализации метода Гаусса. Различные методы Гаусса решения систем линейных алгебраических уравнений для систем с общей и распределенной памятью.
18. Оценка ускорения и эффективности реализаций параллельных алгоритмов метода Гаусса.
19. Параллельные алгоритмы сортировки на базе последовательных алгоритмов: сортировки пузырьком и сортировка Шелла.
20. Реализации параллельных алгоритмов сортировки для различных технологий распараллеливания. Оценки эффективности распараллеливания построенных алгоритмов.
21. Генетические алгоритмы. Основные понятия и методы, лежащие в основе генетических алгоритмов поиска глобального оптимума.
22. Этапы генетического алгоритма поиска оптимума, подходы к распараллеливанию алгоритма, распараллеливание алгоритма для систем с общей и распределенной памятью.
23. Оценки эффективности распараллеливания построенных генетических алгоритмов.
24. Поиск всех кратчайших путей. Распараллеливание алгоритмов поиска кратчайших путей для систем с общей и распределенной памятью. Оценки эффективности распараллеливания построенных алгоритмов.
25. Задача оптимального разделения графов. Распараллеливание алгоритмов оптимального разделения графов для различных технологий распараллеливания. Оценки эффективности распараллеливания построенных алгоритмов.

4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Формой контроля знаний по дисциплине являются два текущих контроля знаний в форме тестирования и экзамен, проводимый по материалам лекций и выполненным практическим заданиям.

Неделя текущего контроля	Вид оценочного средства	Код компетенций, оценивающих знания, умения, навыки	Содержание оценочного средства	Требования к выполнению	Срок сдачи (неделя семестра)	Критерии оценки по содержанию и качеству с указанием баллов
Проводятся в сроки, установленные графиком образовательного процесса	Тестирование 1, 2	ПК-3 ПК-4	20 вопросов	Компьютерное тестирование; время, отведенное на процедуру - 30 минут	Результаты тестирования предоставляются в день проведения процедуры	Критерии оценки определяются процентным соотношением. Не явка - 0 Удовлетворительно - от 51% правильных ответов. Хорошо - от 70%. Отлично – от 90%.

						Максимальная оценка – 5 баллов
Проводится в сроки, установленные графиком образовательного процесса	Экзамен	ПК-3 ПК-4	2 вопроса и 1 задача	Экзамен проводится в устной форме, путем ответа на вопросы. Время, отведенное на процедуру – 45 минут.	Результаты предоставляются в день проведения экзамена	<p>Критерии оценки:</p> <p>«Отлично»: знание основных понятий предмета; умение использовать и применять полученные знания на практике; работа на практических занятиях; знание основных научных теорий, изучаемых предметов; ответ на вопросы билета.</p> <p>«Хорошо»: • знание основных понятий предмета; • умение использовать и применять полученные знания на практике; • работа на практических занятиях; • знание основных научных теорий, изучаемых предметов; • ответы на вопросы билета</p> <p>• неправильно решено практическое задание</p> <p>«Удовлетворительно»: демонстрирует частичные знания по темам дисциплин; незнание неумение использовать и применять полученные знания на практике; не работал на практических занятиях;</p> <p>Неудовлетворительно»: демонстрирует</p>

						частичные знания по темам дисциплин; незнание основных понятий предмета; неумение использовать и применять полученные знания на практике; не работал на практических занятиях; не отвечает на вопросы.
--	--	--	--	--	--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

4.1. Типовые вопросы, выносимые на экзамен

1. Основные направления развития высокопроизводительных компьютеров. Пути достижения параллелизма: независимость функционирования отдельных функциональных устройств, избыточность элементов вычислительной системы, дублирование устройств.
2. Векторная и конвейерная обработка данных.
3. Многопроцессорная и многомашинная, параллельная обработка данных.
4. Закон Мура, сдерживающие факторы наращивания количества транзисторов на кристалле и частоты процессоров. Сдерживающие факторы повсеместного внедрения параллельных вычислений.
5. Перечень критических задач, решение которых без использования параллельных вычислений затруднено или вовсе невозможно.
6. Однопроцессорная оптимизация. Архитектурно-зависимая оптимизация; отличия развертывания циклов для векторных и кэш-ориентированных архитектур. Конвейерная обработка данных. Зависимость производительности процессора от способа описания и хранения данных.
7. Стандартные методики измерения производительности MIPS, MFLOPS и т.д.
8. Классификация многопроцессорных вычислительных систем
9. Парадигмы, модели и технологии параллельного программирования
10. Параллельное программирование с использованием интерфейса передачи сообщений MPI
11. Параллельное программирование на системах с общей памятью (OpenMP)
12. Параллельное программирование на системах смешанного типа
13. Классификация ошибок параллельных программ (сильные, слабые ошибки).
14. Особенности отладки параллельных приложений. Трассировка.
15. Степень параллелизма численного алгоритма. Средняя степень параллелизма численного алгоритма. Зернистость алгоритма. Ускорение и эффективность. Закон Амдала.
16. Определение параллелизма: анализ задачи с целью выделить подзадачи, которые могут выполняться одновременно. Выявление параллелизма:

изменение структуры задачи таким образом, чтобы можно было эффективно выполнять подзадачи. Выражение параллелизма: реализация параллельного алгоритма в исходном коде с помощью системы обозначений параллельного программирования.

17. Параллельный алгоритм умножения матрицы на вектор и его ускорение по сравнению с последовательным алгоритмом.
18. Параллельный алгоритм умножения матрицы на матрицу и его ускорение по сравнению с последовательным алгоритмом.
19. Параллельный алгоритм решения СЛАУ прямым методом Гаусса и его ускорение по сравнению с последовательным алгоритмом.
20. Параллельный алгоритм решения СЛАУ итерационными методами Якоби, Гаусса - Зейделя и их ускорение по сравнению с последовательным алгоритмом.
21. Понятия вычислительная сложность и относительного времени выполнения алгоритма.
22. Модель алгоритма, определение и примеры графа «операции - операнды» и расписания параллельного алгоритма.
23. Время выполнения, ускорение и эффективность параллельного алгоритма.
24. Понятия степени параллелизма компьютера и алгоритма, понятие паракомпьютера.
25. Понятие сверхлинейного ускорения и причины его возникновения.
26. Понятие средней степени параллелизма алгоритма. Максимальное достижимое ускорение, закон Амдаля.
27. Этапы проектирования параллельного алгоритма.
28. Действия на этапах декомпозиции, проектирования коммуникаций, масштабирования подзадач, планирования вычислений.
29. Параллельные алгоритм умножения матрицы на вектор и матрицы на матрицу для систем с общей памятью.
30. Параллельный алгоритм умножения матрицы на вектор и матрицы на матрицу для систем с распределенной памятью.
31. Альтернативные параллельные алгоритмы матрично-векторного умножения: алгоритмы Виноградова и Фокса.
32. Теоретические оценки эффективности и ускорения алгоритмов умножения матриц.
33. Прямой и обратный проходы последовательного алгоритма метода Гаусса.
34. Идея параллельной реализации метода Гаусса для систем с общей памятью.
35. Способы декомпозиции матрицы СЛАУ при параллельной реализации метода Гаусса.
36. Параллельная реализация выбора ведущего элемента в методе Гаусса для систем с распределенной памятью.
37. Схема программной реализации параллельного варианта метода Гаусса с использованием MPI.
38. Теоретические оценки эффективности и ускорения параллельного алгоритма метода Гаусса.

39. Основные шаги последовательного алгоритма метода сопряженных градиентов. Предобуславливание матрицы системы линейных уравнений. Вычислительная трудоемкость алгоритма.
40. Формат CSIR хранения матрицы СЛАУ.
41. Распараллеливание метода сопряженных градиентов для систем с общей распределенной памятью. Ускорение и эффективность параллельного алгоритма метода сопряженных градиентов.
42. Последовательный алгоритм метода минимальных невязок, вычислительная трудоемкость метода минимальных невязок.
43. Распараллеливание метода минимальных невязок. Ускорение и эффективность параллельного алгоритма метода минимальных невязок.
44. Пузырьковая сортировка и ее чет-нечетная модификация. Вычислительная трудоемкость алгоритма чет-нечетной сортировки.
45. Распараллеливание алгоритма чет-нечетной перестановки для систем с общей распределенной памятью. Ускорение и эффективность параллельного алгоритма чет-нечетной сортировки.
46. Последовательная сортировка Шелла, вычислительная трудоемкость алгоритма сортировка Шелла.
47. Распараллеливание сортировки Шелла для систем с общей распределенной памятью. Ускорение и эффективность параллельного алгоритма чет-нечетной сортировки.
48. Последовательная быстрая сортировка, вычислительная трудоемкость алгоритма сортировка Шелла.
49. Распараллеливание быстрой сортировки для систем с общей распределенной памятью. Ускорение и эффективность параллельного алгоритма быстрой сортировки.
50. Принципы построения генетических алгоритмов на примере задачи о ферзях. Вычислительная трудоемкость генетических алгоритмов.
51. Подходы к распараллеливанию генетического алгоритма. Теоретические оценки ускорения и эффективности распараллеливания генетических алгоритмов.

Итоговое начисление баллов по дисциплине осуществляется в соответствии с разработанной и внедренной балльно-рейтинговой системой контроля и оценивания уровня знаний и внеучебной созидательной активности обучающихся.

*ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И ТЕХНОЛОГИЙ
КАФЕДРА МАТЕМАТИКИ И ЕСТЕСТВЕННОНАУЧНЫХ ДИСЦИПЛИН*

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО
ОСВОЕНИЮ ДИСЦИПЛИНЫ
«Технологии параллельного программирования»**

Направление подготовки: 01.03.02 Прикладная математика и информатика

Профиль: Программирование, математическое моделирование

Уровень высшего образования: бакалавриат

Форма обучения: очная

Королев
2023

1. Общие положения

Цель дисциплины:

1. формирование способности к восприятию, обобщению и анализу информации, необходимой для оценки вариантов, обеспечения и поддержки принятия эффективных конструкторских и управленческих решений;
2. освоение необходимого инструментария, применяемого при оценке вариантов, выборе эффективных конструкторских и управленческих решений;
3. формирование готовности применять методы математического анализа и моделирования сложных систем и процессов в профессиональной деятельности.

Задачи дисциплины:

1. Освоение студентами теоретических методов дисциплины, применяемых при оценке вариантов и выборе эффективных конструкторских и управленческих решений;
2. Получение студентами умений и навыков, применяемых для решения практических задач оценки вариантов и выбора эффективных конструкторских и управленческих решений.

2. Указания по проведению практических занятий

Практическое занятие 1.

Вид практического занятия: *компьютерное моделирование.*

Образовательные технологии: *технологии компьютерного обучения*

Тема и содержание занятия: *Разработка и анализ типовых параллельных алгоритмов в среде Visual Studio.*

Продолжительность занятия – 4 ч.

Практическое занятие 2.

Вид практического занятия: *компьютерное моделирование.*

Образовательные технологии: *технологии компьютерного обучения*

Тема и содержание занятия: *Разработка параллельного алгоритма решения системы линейных алгебраических уравнений в среде Visual Studio.*

Продолжительность занятия – 6 ч.

Практическое занятие 3.

Вид практического занятия: *компьютерное моделирование.*

Образовательные технологии: *технологии компьютерного обучения*

Тема и содержание занятия: *Разработка параллельного алгоритма сортировки в среде Visual Studio.*

Продолжительность занятия – 6 ч.

3. Указания по проведению лабораторного практикума

Лабораторная работа 1.

Вид занятия: *компьютерное моделирование.*

Образовательные технологии: *технологии компьютерного обучения*

Тема и содержание занятия: *Разработка и анализ типовых параллельных алгоритмов в среде Visual Studio.*

Продолжительность занятия – 4 ч.

Лабораторная работа 2.

Вид занятия: *компьютерное моделирование.*

Образовательные технологии: *технологии компьютерного обучения*

Тема и содержание занятия: *Разработка параллельного алгоритма решения системы линейных алгебраических уравнений в среде Visual Studio.*

Продолжительность занятия – 6 ч.

Лабораторная работа 3.

Вид занятия: *компьютерное моделирование.*

Образовательные технологии: *технологии компьютерного обучения*

Тема и содержание занятия: *Разработка параллельного алгоритма сортировки в среде Visual Studio.*

Продолжительность занятия – 6 ч.

4. Указания по проведению самостоятельной работы студентов

№ п/п	Наименование блока (раздела) дисциплины	Виды СРС
1.	Анализ параллельных алгоритмов. Принципы разработки параллельных методов	1. Подготовка к практическим занятиям по материалам лекций и учебной литературы. 2. Выполнение практических заданий 3. Самостоятельное изучение некоторых вопросов дисциплины (типовые параллельные алгоритмы).
2.	Распараллеливание матрично - векторных операций. Решение систем линейных алгебраических уравнений	1. Подготовка к практическим занятиям по материалам лекций и учебной литературы. 2. Выполнение практических заданий 3. Самостоятельное изучение некоторых вопросов дисциплины (матричные параллельные операции).
3.	Параллельные сортировки и алгоритмы на графах	1. Подготовка к практическим занятиям по материалам лекций и учебной литературы. 2. Выполнение практических заданий 3. Самостоятельное изучение некоторых вопросов дисциплины (параллельные алгоритмы на графах).

5. Указания по проведению контрольных работ для обучающихся очной формы обучения

Пояснения по содержанию, выполнению и оформлению работы:

Контрольная работа проводится с использованием ЭВМ. Задача оценивается в 5 баллов. Учитывается наличие всех указанных в задаче объектов метаданных, процедур обработки информации в модулях форм или документов, удобного интерфейса.

ФОРМА ОТЧЕТНОСТИ

После создания проекта обучающиеся предоставляют письменный отчет о проделанной работе (контрольную работу).

Контрольная работа должна содержать:

Титульный лист установленного образца

Содержание

Индивидуальное задание (указать индивидуальное задание, цели и задачи выполнения данного задания)

2. Техника безопасности

2. Выбор средств и реализация поставленных задач (описание работы)

3. Выводы (самоанализ проделанной работы: итог, трудности, новое, интересное и т. д.)

4. Список литературы

Приложения

Критерии оценок:

«отлично» - все объекты метаданных присутствуют, обмен информацией верен, пользовательские процедуры написаны на встроенном языке и выполняются без сбоев, интерфейс удобен, конфигурация успешно работает, созданы интерфейс и набор прав пользователей.

«хорошо» - все объекты метаданных присутствуют, обмен информацией может содержать незначительные ошибки, пользовательские процедуры написаны на встроенном языке и выполняются без сбоев, конфигурация успешно работает, созданы интерфейс и набор прав пользователей.

«удовлетворительно» - основные объекты метаданных присутствуют, обмен информацией может содержать незначительные ошибки, пользовательские процедуры на встроенном языке выполняются без сбоев, конфигурация успешно работает.

«неудовлетворительно» - основные объекты метаданных присутствуют, обмен информацией содержит ошибки, пользовательские процедуры на встроенном языке содержат ошибки либо вообще отсутствуют, конфигурация не работает.

Контрольная работа сдается на электронных носителях в виде каталога с названием фамилии студента, содержащего информационную базу и задачи на бумажном носителе в виде контрольной работы.

УКАЗАНИЯ ПО ОФОРМЛЕНИЮ ОТЧЕТА

Общие требования

Текстовые документы выполняются на листах белой бумаги стандартного формата А4 (Приложение В, Д). Для страницы документа устанавливаются поля: Верхнее - 1,6 мм; Левое - 2,4 мм; правое - 1,0 мм, нижнее - 3,1 мм.

Примечание: Параметры страницы устанавливаются в меню Файл->Параметры страницы:

Текстовые документы выполняются с применением персонального компьютера, в текстовом процессоре MS Word, шрифтом Times New Roman №16

(по усмотрению преподавателя может быть №14), междустрочный интервал 1,5, строчными буквами.

Поля устанавливаются для текста:

в начале строки – не менее 3 мм

в конце строки – не менее 3 мм.

Расстояние от верхней и нижней строки текста до линии рамки должно быть не менее 10 мм. (Приложение В, Г)

Поврежденные листы, помарки, следы прежнего, не полностью удаленного текста не допускаются.

Абзацный отступ – 1,25мм. Устанавливается в меню Формат->Абзац:

Построение документов

Текст документа при необходимости разделяют на разделы и подразделы.

Разделы должны иметь порядковые номера, обозначенные арабскими цифрами с точкой в пределах документа. Подразделы должны иметь нумерацию в пределах раздела. Номера подразделов состоят из номера раздела и подраздела, разделенной точкой.

В конце номера подраздела, также ставится точка.

Разделы, как и подразделы, могут состоять из одного или нескольких пунктов

Если раздел состоит из одного пункта, он нумеруется. Номер пункта состоит из номера раздела, номера подраздела и номера пункта, разделенных точками.

Подпункты в пределах пункта или перечисление требований, указаний, положений обозначаются арабскими цифрами со скобкой, например, 1), 2), 3) и т.д.

Каждый пункт, подпункт, перечисления записывают с нового абзаца, шрифтом Times New Roman № 18

Наименование разделов и подразделов записываются в виде заголовка, выровненного по центру, шрифтом Times New Roman №22, прописными буквами.

Наименование разделов и подразделов записывается в виде заголовка, шрифтом Times New Roman №22, строчными буквами, первая буква прописная.

В конце заголовка точку не ставят.

Расстояние между заголовками раздела и подразделов 10 мм (междустрочный интервал полуторный), между заголовками и текстом 15 мм (междустрочный интервал двойной), между последней строкой текста и последующим заголовком должно быть 15мм. (междустрочный интервал двойной).

Каждый раздел документа следует начинать с нового листа.

Каждый пункт текста записывается с абзаца (отступ красной строки 15 мм.), цифры, указывающие номер пунктов, не должны выступать за границу абзаца.

Оформление формул

В формулах в качестве символов следует применять обозначения, установленные соответствующими стандартами

Если в документе более одной формулы, их нумеруют арабскими цифрами в пределах раздела или всего документа. Номер указывают с правой стороны листа, арабскими цифрами на уровне формулы в круглых скобках.

Оформление иллюстраций и приложений

Количество иллюстраций должно быть достаточным для пояснения излагаемого текста.

Все иллюстрации нумеруются в пределах всего документа арабскими цифрами сквозной нумерации, за исключением иллюстраций приложений.

Иллюстрации каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения.

Пример оформления списка литературы: Андреев А.Ф. Применение грузозахватных устройств для строительно-монтажных работ: М.: Стройиздат, 1985. – 400 С.

6. Перечень основной и дополнительной учебной литературы

Основная литература:

1. Арыков С.Б. Параллельное программирование над общей памятью: OpenMP: [16+] / С.Б. Арыков, М.А. Городничев, Г.А. Щукин; Новосибирский государственный технический университет. – Новосибирск: Новосибирский государственный технический университет, 2019. – 95 с.: ил., табл. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=576119>.
2. Лебедев, А. С. Технология параллельного программирования: учебно-методическое пособие / А. С. Лебедев, Ш. Г. Магомедов. — Москва: РТУ МИР-ЭА, 2021. — 98 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/176524>

Дополнительная литература:

1. Стуколов С. В. Параллельное программирование. Практикум: учебное пособие / С. В. Стуколов. — Кемерово: КемГУ, 2020. — 273 с. — ISBN 978-5-8353-2723-2. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/173547>
2. Елесина, С. И. Технология параллельного программирования OpenMP: учебное пособие / С. И. Елесина. — Рязань: РГРТУ, 2021. — 48 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/220409>
3. Богачёв К. Ю. Основы параллельного программирования: учебное пособие / К. Ю. Богачёв. — 4-е изд. — Москва: Лаборатория знаний, 2020. — 345 с. — ISBN 978-5-00101-758-5. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/135516>.

7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

Интернет-ресурсы:

1. <http://biblioclub.ru/> - Университетская библиотека ONLINE.
2. <http://www.rucont.ru/> - Электронная библиотека Руконт.
3. <http://www.znaniium.com/> - Электронная библиотека издательства Инфра-М.
4. <http://e.lanbook.com/> - Электронная библиотека Лань.
5. <http://www.olap.ru/> - Сайт OLAP-технологий.

6. <https://ru.atlassian.com/software/jira/> - Система планирования работ проекта ПО.

7. <https://www.jetbrains.com/youtrack/> - Система планирования работ проекта ПО.

8. <https://ru.atlassian.com/software/confluence> - Система хранения проектной документации.

9. <https://git-scm.com/> - Распределенная система контроля версий ПО.

10. <https://ru.wikipedia.org/wiki/Subversion> - Централизованная система контроля версий ПО.

11. <https://jenkins-ci.org/> - Система непрерывной интеграции проекта ПО.

12. <https://www.docker.com/> - Система виртуализации окружения.

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

Перечень программного обеспечения: MSOffice

Информационные справочные системы: Электронные ресурсы образовательной среды Университета