

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего
образования «Ростовский государственный экономический университет (РИНХ)»

УТВЕРЖДАЮ
Директор Таганрогского института
имени А.П. Чехова (филиала)
РГЭУ (РИНХ)
_____ Голобородько А.Ю.
«_____» _____ 20__ г.

Рабочая программа дисциплины
Параллельные алгоритмы

направление 09.03.03 Прикладная информатика
направленность (профиль) 09.03.03.01 Прикладная информатика в менеджменте

Для набора _____ года

Квалификация
Бакалавр

КАФЕДРА информатики**Распределение часов дисциплины по семестрам**

Семестр (<Курс>.<Семестр на курсе>)	7 (4.1)		Итого	
	УП	РП	УП	РП
Неделя	17 4/6			
Вид занятий	УП	РП	УП	РП
Лекции	16	16	16	16
Лабораторные	16	16	16	16
Итого ауд.	32	32	32	32
Контактная работа	32	32	32	32
Сам. работа	40	40	40	40
Итого	72	72	72	72

ОСНОВАНИЕ

Учебный план утвержден учёным советом вуза от 29.08.2023 протокол № 1.

Программу составил(и): канд. техн. наук, Доц., Буланов Сергей Георгиевич _____

Зав. кафедрой: Тюшнякова И. А. _____

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1.1	формирование системы знаний, умений и навыков, связанных с особенностями параллельных вычислений и современных технологий программирования как базы для развития универсальных компетенций и основы для развития профессиональных компетенций
-----	---

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

ОПК-3.1:	Знает принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.
ОПК-3.2:	Умеет решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.
ОПК-3.3:	Владеет навыками подготовки обзоров, аннотаций, составления рефератов, научных докладов, публикаций, и библиографии по научно-исследовательской работе с учетом требований информационной безопасности.
ОПК-7.1:	Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий.
ОПК-7.2:	Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.
ОПК-7.3:	Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач.

В результате освоения дисциплины обучающийся должен:

Знать:	основы информационной и библиографической культуры с применением информационно-коммуникационных технологий; алгоритмы и программы, пригодные для практического применения;
Уметь:	решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий; разрабатывать алгоритмы и программы, пригодные для практического применения;
Владеть:	решать стандартные задачи профессиональной деятельности с учетом основных требований информационной безопасности; по разработке алгоритмов и программ для практического применения.

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Компетенции	Литература
	Раздел 1. Параллельные вычисления и приложения				
1.1	Архитектура ВС. Классификация вычислительных систем /Лек/	7	2	ОПК-3.1 ОПК-7.3	Л1.3Л2.1 Л2.2
1.2	Классификация вычислительных систем /Ср/	7	2	ОПК-3.2 ОПК-7.1	Л1.2 Л1.3Л2.1 Л2.2
1.3	Знакомство с многопоточной обработкой /Лаб/	7	2	ОПК-3.3 ОПК-7.3	Л1.1 Л1.3Л2.1 Л2.3
1.4	Мультикомпьютеры с распределенной памятью /Ср/	7	2	ОПК-3.1 ОПК-7.3	Л1.2 Л1.3Л2.2 Л2.3
1.5	Способы достижения параллелизма /Лек/	7	2	ОПК-3.1 ОПК-7.1	Л1.1 Л1.2Л2.2 Л2.3

1.6	Уровни параллелизма в многоядерных архитектурах /Ср/	7	2	ОПК-3.2 ОПК-7.2	Л1.1 Л1.3Л2.1 Л2.2
1.7	Поиск простых чисел /Лаб/	7	2	ОПК-3.3 ОПК-7.3	Л1.2 Л1.3Л2.1 Л2.3
1.8	Анализ эффективности параллельных вычислений /Ср/	7	2	ОПК-3.1 ОПК-7.3	Л1.1 Л1.3Л2.1 Л2.3
1.9	Основные этапы разработки параллельных приложений /Лек/	7	2	ОПК-3.3 ОПК-7.3	Л1.1 Л1.2Л2.1 Л2.2
1.10	Проблемы разработки параллельных приложений. Декомпозиция /Ср/	7	2	ОПК-7.1 ОПК-7.2	Л1.2 Л1.3Л2.1 Л2.2
Раздел 2. Параллельное программирование на C#					
2.1	Работа с потоками. Многопоточная обработка /Лек/	7	2	ОПК-3.1 ОПК-7.2	Л1.1 Л1.2Л2.2 Л2.3
2.2	Разработка параллельных приложений /Ср/	7	2	ОПК-3.1 ОПК-7.3	Л1.2 Л1.3Л2.1 Л2.3
2.3	Синхронизация доступа к одноэлементному буферу /Лаб/	7	2	ОПК-3.3 ОПК-7.2	Л1.1 Л1.3Л2.2 Л2.3
2.4	Модели параллельных приложений /Ср/	7	2	ОПК-3.3 ОПК-7.2	Л1.2 Л1.3Л2.1 Л2.3
2.5	Средства синхронизации /Лек/	7	2	ОПК-7.1 ОПК-7.3	Л1.1 Л1.3Л2.1 Л2.2
2.6	Работа с потоками. Структура потока. Состояния потоков /Ср/	7	4	ОПК-3.1 ОПК-3.3	Л1.2 Л1.3Л2.1 Л2.2
2.7	Конкурентные коллекции. Работа с задачами /Лек/	7	2	ОПК-3.2 ОПК-7.2	Л1.1 Л1.3Л2.2 Л2.3
2.8	Средства синхронизации. Сигнальные сообщения. Семафоры. /Ср/	7	4	ОПК-3.1 ОПК-7.2	Л1.1 Л1.2Л2.1 Л2.3
2.9	Синхронизация приоритетного доступа к многоэлементному буферу /Лаб/	7	2	ОПК-3.2 ОПК-7.2	Л1.1 Л1.3Л2.1 Л2.3
2.10	Конкурентные коллекции. Эффективность конкурентных коллекций /Ср/	7	4	ОПК-3.1 ОПК-7.3	Л1.2 Л1.3Л2.2 Л2.3
2.11	Шаблоны параллелизма Parallel /Лек/	7	2	ОПК-3.2 ОПК-7.3	Л1.1 Л1.3Л2.2 Л2.3
2.12	Работа с задачами. Статусы задачи. Работа с данными в задаче /Ср/	7	4	ОПК-3.3 ОПК-7.1	Л1.2 Л1.3Л2.1 Л2.3
2.13	Технология PLINQ /Лек/	7	2	ОПК-3.3 ОПК-7.2	Л1.1 Л1.2Л2.2 Л2.3
2.14	Шаблоны параллелизма Parallel. Параллельные циклы. Параллельный запуск /Ср/	7	4	ОПК-3.1 ОПК-7.2	Л1.1 Л1.2Л2.1 Л2.3

2.15	Клеточная модель «Игра Жизнь» Дж.Конвея /Лаб/	7	4	ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.2 Л1.3Л2.1 Л2.3
2.16	Технология PLINQ. Эффективность распараллеливания /Ср/	7	4	ОПК-3.3 ОПК-7.1 ОПК-7.3	Л1.2 Л1.3Л2.2 Л2.3
2.17	Планировщик задач. Опция PreferFairness. Стратегии /Ср/	7	2	ОПК-3.3 ОПК-7.2	Л1.1 Л1.2Л2.1 Л2.3
2.18	Знакомство с «Визуализатором параллелизма» в Visual Studio /Лаб/	7	4	ОПК-3.3 ОПК-7.3	Л1.1 Л1.3Л2.2 Л2.3
2.19	/Зачёт/	7	0	ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3

4. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Структура и содержание фонда оценочных средств для проведения текущей и промежуточной аттестации представлены в Приложении 1 к рабочей программе дисциплины.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Основная литература

	Авторы, составители	Заглавие	Издательство, год	Колич-во
Л1.1	Антонов А. С.	Параллельное программирование с использованием технологии MPI: курс: учебное пособие	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), 2008	http://biblioclub.ru/index.php?page=book&id=233577 неограниченный доступ для зарегистрированных пользователей
Л1.2	Барский А. Б.	Параллельное программирование: монография	Москва: Национальный Открытый Университет «ИНТУИТ», 2016	http://biblioclub.ru/index.php?page=book&id=578026 неограниченный доступ для зарегистрированных пользователей
Л1.3	Туральчук, К. А.	Параллельное программирование с помощью языка C#	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019	http://www.iprbookshop.ru/79714.html неограниченный доступ для зарегистрированных пользователей

5.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год	Колич-во
Л2.1	Левин М. П.	Параллельное программирование с использованием OpenMP: учебное пособие	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ) Бином. Лаборатория знаний, 2008	http://biblioclub.ru/index.php?page=book&id=233111 неограниченный доступ для зарегистрированных пользователей
Л2.2	Арыков С. Б., Городничев М. А., Щукин Г. А.	Параллельное программирование над общей памятью: POSIX Thread: учебное пособие	Новосибирск: Новосибирский государственный технический университет, 2018	http://biblioclub.ru/index.php?page=book&id=576621 неограниченный доступ для зарегистрированных пользователей

	Авторы, составители	Заглавие	Издательство, год	Колич-во
Л2.3	Федотов, И. Е.	Параллельное программирование. Модели и приемы	Москва: СОЛОН-ПРЕСС, 2018	http://www.iprbookshop.ru/90420.html неограниченный доступ для зарегистрированных пользователей

5.3 Профессиональные базы данных и информационные справочные системы

rsl.ru – Российская государственная библиотека

elibrary.ru – Научная электронная библиотека

biblioclub.ru – Университетская библиотека онлайн

intuit.ru – Национальный Открытый Университет «ИНТУИТ»

5.4. Перечень программного обеспечения

DelphiStudio

Microsoft Office

5.5. Учебно-методические материалы для студентов с ограниченными возможностями здоровья

При необходимости по заявлению обучающегося с ограниченными возможностями здоровья учебно-методические материалы предоставляются в формах, адаптированных к ограничениям здоровья и восприятия информации. Для лиц с нарушениями зрения: в форме аудиофайла; в печатной форме увеличенным шрифтом. Для лиц с нарушениями слуха: в форме электронного документа; в печатной форме. Для лиц с нарушениями опорно-двигательного аппарата: в форме электронного документа; в печатной форме.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Помещения для проведения всех видов работ, предусмотренных учебным планом, укомплектованы необходимой специализированной учебной мебелью и техническими средствами обучения. Для проведения лекционных занятий используется демонстрационное оборудование. Лабораторные занятия проводятся в компьютерных классах, рабочие места в которых оборудованы необходимыми лицензионными программными средствами и выходом в интернет.

7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

Методические указания по освоению дисциплины представлены в Приложении 2 к рабочей программе дисциплины.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

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

1.1 Показатели и критерии оценивания компетенций:

ЗУН, составляющие компетенцию	Показатели оценивания	Критерии оценивания	Средства оценивания
ОПК-3 – способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности			
<i>Знать:</i> основы информационной и библиографической культуры с применением информационно-коммуникационных технологий;	Выполняет лабораторные задания. Выполняет поиск и сбор необходимой информации. Изучает лекционный материал, основную и дополнительную литературу. Выполняет поиск и сбор необходимой информации.	Полнота и правильность выполнения лабораторных заданий, наличие выводов. Полное, развернутое, грамотное и логическое изложение вопроса. Полнота и содержательность ответа.	ЛЗ – лабораторные задания (1-2) О – опрос З – вопросы к зачету (1-17)
<i>Уметь:</i> решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий;	Выполняет лабораторные задания. Выполняет поиск и сбор необходимой информации. Изучает лекционный материал, основную и дополнительную литературу. Выполняет поиск и сбор необходимой информации.	Полнота и правильность выполнения лабораторных заданий, наличие выводов. Полное, развернутое, грамотное и логическое изложение вопроса. Полнота и содержательность ответа.	ЛЗ – лабораторные задания (5-6) О – опрос З – вопросы к зачету (1-17)
<i>Иметь навыки:</i> решать стандартные задачи профессиональной деятельности с учетом основных требований информационной безопасности;	Выполняет лабораторные задания. Выполняет поиск и сбор необходимой информации. Изучает лекционный материал, основную и дополнительную литературу. Выполняет поиск и сбор	Полнота и правильность выполнения лабораторных заданий, наличие выводов. Полное, развернутое, грамотное и логическое изложение вопроса. Полнота и	ЛЗ – лабораторные задания (3-4) О – опрос З – вопросы к зачету (1-17)

	необходимой информации.	содержательность ответа.	
ОПК-7 – способен разрабатывать алгоритмы и программы, пригодные для практического применения			
<i>Знать:</i> алгоритмы и программы, пригодные для практического применения;	Выполняет лабораторные задания. Выполняет поиск и сбор необходимой информации. Изучает лекционный материал, основную и дополнительную литературу. Выполняет поиск и сбор необходимой информации.	Полнота и правильность выполнения лабораторных заданий, наличие выводов. Полное, развернутое, грамотное и логическое изложение вопроса. Полнота и содержательность ответа.	ЛЗ – лабораторные задания (5-6) О – опрос З – вопросы к зачету (1-17)
<i>Уметь:</i> разрабатывать алгоритмы и программы, пригодные для практического применения;	Выполняет лабораторные задания. Выполняет поиск и сбор необходимой информации. Изучает лекционный материал, основную и дополнительную литературу. Выполняет поиск и сбор необходимой информации.	Полнота и правильность выполнения лабораторных заданий, наличие выводов. Полное, развернутое, грамотное и логическое изложение вопроса. Полнота и содержательность ответа.	ЛЗ – лабораторные задания (1-2) О – опрос З – вопросы к зачету (1-17)
<i>Иметь навыки:</i> по разработке алгоритмов и программ для практического применения.	Выполняет лабораторные задания. Выполняет поиск и сбор необходимой информации. Изучает лекционный материал, основную и дополнительную литературу. Выполняет поиск и сбор необходимой информации.	Полнота и правильность выполнения лабораторных заданий, наличие выводов. Полное, развернутое, грамотное и логическое изложение вопроса. Полнота и содержательность ответа.	ЛЗ – лабораторные задания (3-4) О – опрос З – вопросы к зачету (1-17)

1.2 Шкалы оценивания:

Текущий контроль успеваемости и промежуточная аттестация осуществляется в рамках накопительной балльно-рейтинговой системы в 100-балльной шкале:

7 семестр

50-100 баллов (зачет);

0-49 баллов (незачет).

2 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

Вопросы к зачету

1. Параллельные вычисления. Мотивы параллелизма. Классификация вычислительных систем.
2. Архитектура однопроцессорной машины. Мультикомпьютеры с распределенной памятью. Мультипроцессор с разделяемой памятью.
3. Режимы выполнения независимых частей программы. Уровни параллелизма в многоядерных архитектурах.
4. Анализ эффективности параллельных вычислений. Пределы параллелизма.
5. Проблемы разработки параллельных приложений. Декомпозиция. Декомпозиция по данным. Масштабирование подзадач.
6. Проблемы разработки параллельных приложений. Проблема гонки данных. Проблемы синхронизации.
7. Проблемы разработки параллельных приложений. Проблемы кэшируемой памяти. Модели параллельных приложений
8. Работа с потоками. Структура потока. Состояния потоков. Переключение контекста. Работа с потоками в C#
9. Средства синхронизации. Средства для взаимного исключения. Monitor. Mutex.
10. Средства синхронизации. Сигнальные сообщения. Семафоры. Атомарные операторы.
11. Конкурентные коллекции. Эффективность конкурентных коллекций
12. Работа с задачами. Статусы задачи. Работа с данными в задаче.
13. Шаблоны параллелизма Parallel. Параллельные циклы. Параллельный запуск.
14. Технология PLINQ. Эффективность распараллеливания.
15. Планировщик задач. Опция PreferFairness. Стратегии.
16. Типовые модели параллельных приложений. Модель делегирования.
17. Типовые модели параллельных приложений. Модель с равноправными узлами. Модель конвейера

Критерии оценки:

- оценка «зачтено (50-100 баллов)» выставляется студенту, если он в ходе ответа показал наличие твердых знаний по вопросу, изложение ответов с отдельными ошибками, уверенно исправленными после дополнительных вопросов; правильные в целом действия по применению знаний на практике;

- оценка «не зачтено» (менее 50 баллов) выставляется студенту, если ответы не связаны с вопросами, наличие грубых ошибок в ответе, непонимание сущности излагаемого вопроса, неумение применять знания на практике, неуверенность и неточность ответов на дополнительные и наводящие вопросы.

Опрос

Вопросы

1. Почему эффект от распараллеливания наблюдается только при большем числе элементов?
2. Почему увеличение сложности обработки повышает эффективность многопоточной обработки?
3. Какое число потоков является оптимальным для конкретной вычислительной системы?

4. Почему неравномерность загрузки потоков приводит к снижению эффективности многопоточной обработки?
5. Какие другие варианты декомпозиции позволяют увеличить равномерность загрузки потоков?
6. В какой ситуации круговая декомпозиция не обеспечивает равномерную загрузку потоков?

. Поиск простых чисел

Вопросы и упражнения

1. Какими достоинствами и недостатками обладает каждый вариант распараллеливания?
2. Какие средства синхронизации можно использовать вместо конструкции lock? Какой вариант будет более эффективным?
3. Какой вариант ожидания завершения работ, запущенных пулом потоков, более эффективный и почему?
4. Реализуйте один или несколько вариантов распараллеливания с помощью объектов Task и с помощью метода Parallel.For. Выполните эффективность алгоритмов.
5. Реализуйте алгоритм поиска простых чисел как LINQ-запрос к массиву чисел.

Синхронизация доступа к одноэлементному буферу

Вопросы и упражнения

1. Почему проблема гонки данных проявляется не при каждом прогоне?
2. Какие факторы увеличивают вероятность проявления проблемы гонки данных?
3. Возможно ли в данной задаче при отсутствии средств синхронизации возникновение исключения и аварийное завершение программы?
4. Можно ли в данной задаче использовать атомарные операторы для обеспечения согласованности доступа? Необходимы ли при этом дополнительные средства синхронизации?
5. Можно ли в данной задаче использовать потокобезопасные коллекции для обеспечения согласованного доступа?
6. Какие средства синхронизации обеспечивают наилучшее быстродействие в данной задаче? Объясните с чем это связано.

Синхронизация приоритетного доступа к многоэлементному буферу

Вопросы и упражнения

1. Можно ли вместо объектов ManualResetEventSlim использовать другие типы сигнальных сообщений: AutoResetEvent или ManualResetEvent?
2. Какие особенности задачи не позволяют использовать объект ReaderWriterSlim?
3. Почему структура кольцевого буфера не требует синхронизации при работе одного читателя и одного писателя?
4. Почему в предложенной реализации не используются критические секции?
5. Реализуйте учет времени обращения рабочих потоков к буферу.
6. Реализуйте решение задачи с использованием конкурентных коллекций в качестве буфера.

Клеточная модель "Игра Жизнь" Дж.Конвея

Вопросы и упражнения

1. Имеет ли смысл распараллеливание внутреннего цикла расчета? Почему?

```

for(int i=0; i < Height; i++)
{
    Parallel.For(0, Width, j =>
    {
        p = CalcPotential(table[i,j]);
        tableNew[i, j] = lifeRules(p, table[i, j]);
    });
}

```

2. Как вариант расчета – по строкам или по столбцам – более эффективен и с чем это связано?
3. Продумайте вариант блочной декомпозиции, где блок выступает матрицей размера $N \times N$. В чем достоинства и недостатки блочной декомпозиции для этой задачи? Какое значение параметра N следует выбирать?

Вопросы

В каких случаях не возникает проблема гонки данных

(Ответ считается верным, если отмечены все правильные варианты ответов.)

Вариант 1 один поток осуществляет запись в разделяемую переменную типа double, другой поток читает значения

Вариант 2 потоки осуществляют запись в разделяемую переменную типа int

Вариант 3 один поток добавляет элементы в коллекцию типа FIFO (очередь), другой поток извлекает элементы из коллекции

Вариант 4 потоки инкрементируют значения, которые хранятся в разделяемом массиве; каждый поток работает с элементом по уникальному индексу

Выберете правильные утверждения, соответствующие системам с общей памятью

(Ответ считается верным, если отмечены все правильные варианты ответов.)

Вариант 1 обеспечивается параллелизм только на уровне приложений как разделение процессорного времени

Вариант 2 для взаимодействия параллельных частей требуется применять средства синхронизации

Вариант 3 для взаимодействия параллельных потоков не требуется специальных интерфейсов передачи сообщений

Вариант 4 существует возможность параллелизма как на уровне потоков одного процесса, так и на уровне процессов

Выберете правильные утверждения, соответствующие системам с общей памятью

(Ответ считается верным, если отмечены все правильные варианты ответов.)

Вариант 1 при выполнении параллельной программы может возникнуть ситуация гонки данных

Вариант 2 применение средств синхронизации позволяет решить проблему гонки данных

Вариант 3 синхронизация необходима только при выполнении программы в режиме разделения времени

Вариант 4 для реализации параллельного выполнения программы обязательно применение средств синхронизации

Выберете правильные утверждения, соответствующие свойству масштабируемости параллельного приложения

(Отметьте один правильный вариант ответа.)

Вариант 1 приложение может обрабатывать разные объемы данных

Вариант 2 нет правильных вариантов

Вариант 3 приложение обеспечивает ускорение, равное предельному, согласно закону Амдала

Вариант 4 приложение максимально задействует все имеющиеся вычислительные ресурсы

Параллельный алгоритм состоит из четырех независимых подзадач, которые выполняются за одинаковое время. Ускорение алгоритма на двудерной системе составляет 1.5. Чему равно ускорение алгоритма на четырех ядерной системе?

(Отметьте один правильный вариант ответа.)

Вариант 1 3

Вариант 2 2

Вариант 3 1.5

Вариант 4 6

Какое предельное ускорение может быть получено согласно закону Амдала для параллельной программы, содержащей 40% параллельного кода

(Отметьте один правильный вариант ответа.)

Вариант 1 10

Вариант 2 нет правильных ответов

Вариант 3 4

Вариант 4 5

Вопросы

1. Какие достоинства у непосредственной работы с потоками?
2. В чем заключаются основные проблемы организации многопоточной обработки?
3. В каких случаях работа с пулом потоков является более эффективной, чем непосредственная работа с потоками?

Упражнения

Реализуйте пользовательский пул потоков, который повторяет функциональность объекта ThreadPool. Добавьте новые методы пула, которые упрощают работу и делают общение с пулом более информативным. Например, методы ожидания запущенных задач; получение сводной информации о выполнении задач (число потоков, участвовавших в обработке; время обработки).

Вопросы

1. Можно ли организовать работу нескольких потоков без средств синхронизации?
2. Может ли многопоточное приложение, использующее только конструкции lock, войти в состояние взаимоблокировки?
3. Какие средства синхронизации позволяют реализовать функциональность критической секции? В каких случаях следует отдавать предпочтение тому или иному объекту?

Упражнения

1. Исследуйте эффективность легковесных средств синхронизации по сравнению с аналогичными объектами ядра операционной системы: SemaphoreSlim – Semaphore, ManualResetEvent – ManualResetEventSlim.

2. Для анализа можно использовать задачу обращения к разделяемому счетчику:

```
void ThreadFunc() {  
    // Вход в критическую секцию с помощью  
    // того или иного средства синхронизации  
    totalCount++;  
    // Выход из критической секции  
}
```

3. Исследуйте эффективность потокобезопасных коллекций по сравнению с синхронизированным доступом к обычным коллекциям.

4. Исследуйте эффективность атомарных операторов по сравнению со средствами организации критической секции (lock, Monitor, Mutex).
5. Самостоятельно освоите работу с объектами, реализующими типовые схемы синхронизации, CountdownEvent и Barrier. Реализуйте функциональность этих объектов с помощью средств синхронизации, рассмотренных в лекции. Исследуйте эффективность и удобство работы объектов CountdownEvent и Barrier.

Вопросы

1. В чем основные достоинства работы с задачами по сравнению с непосредственной работой с потоками?
2. В каком случае задача завершается со статусом RanToCompletion при возникновении исключения?
3. Каким образом можно реализовать критическую секцию, используя задачи и не используя средств синхронизации, рассмотренных в предыдущей главе?
4. Для каких целей можно использовать вложенные недочерные задачи?

Упражнения

1. Исследуйте эффективность работы большого числа задач по сравнению с работой большого числа потоков.

Вопросы

1. Почему для досрочного выхода из параллельного цикла используются два метода Break и Stop?
2. В чем преимущество пакетной обработки?

Упражнения

1. Исследуйте эффективность цикла Parallel.For в задаче матричного умножения.
2. Исследуйте эффективность цикла Parallel.For с пакетной обработкой и без пакетной обработки.

Вопросы

1. Почему LINQ-запросы не распараллеливаются автоматически?
2. С какой целью используется оператор возвращения к последовательному выполнению запроса AsSequential?
3. Почему при выполнении параллельного запроса порядок элементов может сохраниться?

Упражнения

1. Составьте запросы, которые демонстрируют неэффективность статической декомпозиции.
2. Исследуйте эффективность выполнения PLINQ-запросов и шаблона Parallel.For.
3. Составьте запрос, с помощью которого можно убедиться в параллельности или последовательности его выполнения.
4. Исследуйте эффективность выполнения запроса с разными режимами буферизации.

Вопросы

1. В каком потоке будут выполняться вложенные задачи, если родительская задача запущена с опцией LongRunning?
2. Возможна ли ситуация блокировки потока при ожидании завершения задачи, находящейся в глобальной очереди?
3. В каких случаях возможно параллельное выполнение вложенных задач?
4. В каких случаях нарушается порядок выполнения вложенных задач?

Упражнения

1. Напишите код, который позволяет оценить достоинства и/или недостатки организации локальной очереди потока по принципу LIFO.

Напишите код с вычислительно-емкими задачами и оцените эффективность выполнения задач в пуле потоков и в сторонних потоках.

Критерии оценки:

Каждый ответ оценивается максимум в 2 балл:

- 2 балла – дан полный, развернутый ответ на поставленный вопрос, изложение материала при ответе грамотное, логически стройное;
- 1 балл – в ответе на поставленный вопрос были неточности;
- 0 баллов – студент не владеет материалом по заданному вопросу.

Максимальное количество баллов – 28 (за 14 ответов).

Лабораторные задания

Предполагается 6 лабораторных заданий.

Критерии оценки:

- 10-12 баллов – работа выполнена полностью. Студент владеет теоретическим материалом, отсутствуют ошибки при описании теории, формулирует собственные, самостоятельные, обоснованные, аргументированные суждения, представляет полные и развернутые ответы на дополнительные вопросы.
- 7-9 баллов – работа выполнена полностью. Студент владеет теоретическим материалом, отсутствуют ошибки при описании теории, формулирует собственные, самостоятельные, обоснованные, аргументированные суждения, допуская незначительные ошибки на дополнительные вопросы.
- 4-6 балла – работа выполнена полностью. Студент владеет теоретическим материалом на минимально допустимом уровне, отсутствуют ошибки при описании теории, испытывает затруднения в формулировке собственных обоснованных и аргументированных суждений, допуская ошибки на дополнительные вопросы.
- 1-3 балла – работа выполнена не полностью. Студент не владеет теоретическим материалом, допускает грубые ошибки, испытывает затруднения в формулировке собственных суждений, неспособен ответить на дополнительные вопросы.

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

Процедуры оценивания включают в себя текущий контроль и промежуточную аттестацию.

Текущий контроль успеваемости проводится с использованием оценочных средств, представленных в п. 2 данного приложения. Результаты текущего контроля доводятся до сведения студентов до промежуточной аттестации.

Промежуточная аттестация проводится в форме зачета – 7 семестр.

Зачет проводится по окончании теоретического обучения до начала экзаменационной сессии. Количество вопросов в задании – 2. Проверка ответов и объявление результатов производится в день зачет. Результаты аттестации заносятся в зачетную ведомость и зачетную книжку студента. Студенты, не прошедшие промежуточную аттестацию по графику сессии, должны ликвидировать задолженность в установленном порядке.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Методические указания по освоению дисциплины адресованы студентам всех форм обучения.

Учебным планом предусмотрены следующие виды занятий:

- лекции;
- лабораторные работы.

Важным условием успешного освоения дисциплины «Параллельные алгоритмы» является создание системы правильной организации труда, позволяющей распределить учебную нагрузку равномерно в соответствии с графиком образовательного процесса. Большую помощь в этом может оказать составление плана работы на семестр, месяц, неделю, день. Его наличие позволит подчинить свободное время целям учебы, трудиться более успешно и эффективно. С вечера всегда надо распределять работу на завтрашний день. В конце каждого дня целесообразно подвести итог работы: тщательно проверить, все ли выполнено по намеченному плану, не было ли каких-либо отступлений, а если были, по какой причине они произошли. Нужно осуществлять самоконтроль, который является необходимым условием успешной учебы. Если что-то осталось невыполненным, необходимо изыскать время для завершения этой части работы, не уменьшая объема недельного плана. Все задания к лабораторным работам, а также задания, вынесенные на самостоятельную работу, рекомендуется выполнять непосредственно после соответствующей темы лекционного курса, что способствует лучшему усвоению материала, позволяет своевременно выявить и устранить «пробелы» в знаниях, систематизировать ранее пройденный материал, на его основе приступить к овладению новыми знаниями и навыками.

Знакомство с дисциплиной происходит уже на первой лекции, где от студента требуется не просто внимание, но и самостоятельное оформление конспекта. При работе с конспектом лекций необходимо учитывать тот фактор, что одни лекции дают ответы на конкретные вопросы темы, другие – лишь выявляют взаимосвязи между явлениями, помогая студенту понять глубинные процессы развития изучаемого предмета как в истории, так и в настоящее время.

Конспектирование лекций – сложный вид вузовской аудиторной работы, предполагающий интенсивную умственную деятельность студента. Конспект является полезным тогда, когда записано самое существенное и сделано это самим обучающимся. Не надо стремиться записать дословно всю лекцию. Такое «конспектирование» приносит больше вреда, чем пользы. Целесообразно вначале понять основную мысль, излагаемую лектором, а затем записать ее. Желательно запись осуществлять на одной странице листа или оставляя поля, на которых позднее, при самостоятельной работе с конспектом, можно сделать дополнительные записи, отметить непонятные места.

Конспект лекции лучше подразделять на пункты, соблюдая красную строку. Этому в большой степени будут способствовать вопросы плана лекции, предложенные преподавателям. Следует обращать внимание на акценты, выводы, которые делает лектор, отмечая наиболее важные моменты в лекционном материале замечаниями «важно», «хорошо запомнить» и т.п. Можно делать это и с помощью разноцветных маркеров или ручек, подчеркивая термины и определения.

Целесообразно разработать собственную систему сокращений, аббревиатур и символов. Однако при дальнейшей работе с конспектом символы лучше заменить обычными словами для быстрого зрительного восприятия текста.

Работая над конспектом лекций, всегда необходимо использовать не только учебник, но и ту литературу, которую дополнительно рекомендовал лектор. Именно такая

серьезная, кропотливая работа с лекционным материалом позволит глубоко овладеть теоретическим материалом.

В процессе подготовки к лабораторным занятиям, студентам необходимо обратить особое внимание на самостоятельное изучение рекомендованной литературы. При всей полноте конспектирования лекции в ней невозможно изложить весь материал из-за лимита аудиторных часов. Поэтому самостоятельная работа с учебниками, учебными пособиями, научной, справочной литературой, материалами периодических изданий и Интернета является наиболее эффективным методом получения дополнительных знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у студентов свое отношение к конкретной проблеме.

Изучение дисциплины проходит с акцентом на лабораторные работы. По темам лабораторных работ разработаны учебно-методические материалы, в которых изложены подробные методические рекомендации по изучению каждой темы и выполнению заданий. Наличие таких учебно-методических и дидактических материалов позволяет каждому студенту работать в своем индивидуальном темпе, а также дополнительно прорабатывать изучаемый материал во время самостоятельных занятий.

Для успешного овладения предлагаемым курсом студент должен обладать определённой информационной культурой: навыками работы с литературой, умением определять и находить информационные ресурсы, соответствующие целям и задачам образовательного процесса, получать к ним доступ и использовать в целях повышения эффективности своей профессиональной деятельности. При изучении данного курса необходимо максимально использовать компьютер, изучать дополнительные информационные ресурсы.

Подготовка к промежуточной аттестации.

При подготовке к промежуточной аттестации целесообразно:

- внимательно изучить перечень вопросов и определить, в каких источниках находятся сведения, необходимые для ответа на них;
- внимательно прочитать рекомендованную литературу;
- составить краткие конспекты ответов (планы ответов).