

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ  
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ЛУГАНСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ЛГПУ»)**

**Структурное подразделение** Институт физико-математического  
образования, информационных и обслуживающих технологий  
**Кафедра** информационных образовательных технологий и систем

**УТВЕРЖДАЮ**

Директор ИФМОИОТ

Е.Е. Горбенко

2023 г.



Приложение к рабочей программе учебной дисциплины

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ  
для проведения текущего контроля и промежуточной аттестации  
обучающихся по дисциплине  
«Основы параллельных вычислений»**

**По направлению подготовки** 09.03.04 Программная инженерия  
**Профиль подготовки** Программное обеспечение систем и комплексов  
**Квалификация выпускника** – бакалавр  
**Форма обучения** очная, заочная  
**Курс** ОФО – 3 курс, ЗФО – 4 курс

Разработчик  
Капустин Д.А.  
доцент каф. информационных  
образовательных технологий и  
систем  
Заведующий кафедрой  
Д.А. Капустин  
Протокол от «24» ноября 2023 г. №8

Луганск, 2023

# 1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

## 1.1. Область применения

Фонд оценочных средств (ФОС) – неотъемлемая часть рабочей программы дисциплины (модуля) Основы параллельных вычислений и предназначен для контроля и оценки образовательных достижений студентов, освоивших программу дисциплины (модуля).

## 1.2. Цели и задачи фонда оценочных средств

Цель ФОС – установить соответствие уровня подготовки обучающегося требованиям ФГОС ВО бакалавриат / специалитет / магистратура по направлению подготовки 09.03.04 Программная инженерия, утвержденного приказом Министерства образования и науки Российской Федерации от 19 сентября 2017 г. № 920 (с изменениями и дополнениями).

## 1.3. Перечень компетенций, формируемых в процессе освоения основной образовательной программы

Процесс освоения дисциплины направлен на формирование следующих компетенций и индикаторов их достижения:

Код по ФГОС ВО	Индикатор достижения
Профессиональные	
ПК-3. Владеет навыками использования различных технологий разработки программного обеспечения	ПК-3.1. Знать методы и технологии разработки программного обеспечения компьютерных вычислительных систем ПК-3.2. Уметь определять оптимальные методы и технологии разработки программного обеспечения компьютерных вычислительных систем и комплексов ПК-3.3. Владеть навыками применения технологий разработки программного обеспечения систем и комплексов

## 1.4. Этапы формирования компетенций и средства оценивания уровня их сформированности

Этапы формирования компетенций	Компетенции	Контрольно-оценочные средства / способ оценивания
Тема 1. Теоретические основы параллельных вычислений	ПК-3	Выполнение лабораторных работ
Тема 2. Использование многопоточности в параллельных вычислениях	ПК-3	Выполнение лабораторных работ
Тема 3. Параллельные вычисления с использованием технологий TPL	ПК-3	Выполнение лабораторных работ

Тема 4. Параллельное программирование с использованием MPI	ПК-3	Выполнение лабораторных работ
Тема 5. Параллельное программирование с использованием OpenMP	ПК-3	Выполнение лабораторных работ
Текущая аттестация	ПК-3	Контрольная работа
Промежуточная аттестация	ПК-3	Экзамен

### 1.5. Описание показателей формирования компетенций

Код компетенции	Результаты сформированности
ПК-3.	Знает основные направления в области организации параллельных вычислений на многопроцессорных вычислительных системах; технологии параллельного программирования; Умеет разрабатывать параллельные алгоритмы и программы для решения разного класса задач на компьютерах с распределенной памятью и общей памятью; Владеет основами разработки параллельных программ, теоретическими знаниями в области организации взаимодействующих процессов, средствами параллельного программирования MPI, OpenMP, Threads, способами проверки правильности параллельных программ.

### 1.6. Критерии оценивания компетенций на разных этапах их формирования

Вид учебной работы	Количество баллов		
6 семестр / 10-11 триместр			
	ОФО	О-ЗФО	ЗФО
Выполнение и защита практических работ	50		50
Самостоятельная работа	40		40
Экзамен	10		10
Итого за семестр:	100 баллов		100 баллов
Всего	100 баллов		

### Накопительная система оценивания по 100-балльной шкале

Четырехбал- льная система оценивания экзамена	100- балльная шкала	Буквенная шкала, соответствующая 100- балльной шкале	Система оценивания зачета
Отлично	90–100	А – отлично – теоретическое содержание курса освоено полностью, без пробелов; необходимые практические навыки работы с освоенным материалом сформированы; все предусмотренные программой обучения учебные задания выполнены, качество их	

		выполнения оценено числом баллов, близким к максимальному	Зачтено
Хорошо	<b>83–89</b>	<b>В</b> – очень хорошо – теоретическое содержание курса освоено полностью, без пробелов; необходимые практические навыки работы с освоенным материалом в основном сформированы; все предусмотренные программой обучения учебные задания выполнены, качество выполнения большинства из них оценено числом баллов, близким к максимальному	
Хорошо	<b>75–82</b>	<b>С</b> – хорошо – теоретическое содержание курса освоено полностью; некоторые практические навыки работы с освоенным материалом сформированы недостаточно; все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	
Удовлетворительно	<b>63–74</b>	<b>Д</b> – удовлетворительно – теоретическое содержание дисциплины освоено частично, но пробелы не носят существенного характера; необходимые практические навыки работы с освоенным материалом в основном сформированы; большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, содержат ошибки	
Удовлетворительно	<b>50–62</b>	<b>Е</b> – посредственно – теоретическое содержание курса освоено частично; некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения учебные задания не выполнены либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному	Не зачтено
Неудовлетворительно	<b>21–49</b>	<b>FX</b> – неудовлетворительно – теоретическое содержание курса освоено частично; необходимые практические навыки работы не сформированы; большинство предусмотренных программой обучения учебных заданий не выполнено либо качество их выполнения оценено числом баллов, близким к минимальному; при дополнительной самостоятельной работе над материалом курса возможно повышение качества выполнения учебных заданий	
Неудовлетворительно	<b>0–20</b>	<b>F</b> – неудовлетворительно – теоретическое содержание курса не освоено; необходимые практические навыки работы не сформированы; все выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над материалом курса не приведет к какому-либо	

		значимому повышению качества выполнения учебных заданий	
--	--	---	--

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФГБОУ ВО «ЛГПУ»  
ИНСТИТУТ ФИЗИКО-МАТЕМАТИЧЕСКОГО ОБРАЗОВАНИЯ, ИНФОРМАЦИОННЫХ  
И ОБСЛУЖИВАЮЩИХ ТЕХНОЛОГИЙ

2024 – 2025 учебный год

Направление подготовки (специальность): 09.03.04 Программная инженерия

Профиль подготовки Программное обеспечение систем и комплексов

курс / форма обучения: 3 курс, ОФО, 4 курс ЗФО

Семестр / триместр 6 семестр / 10-11 триместр

Учебная дисциплина: Основы параллельных вычислений

**ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1**

1. Идея параллельной реализации метода Гаусса для систем с общей памятью.
2. Понятия степени параллелизма компьютера и алгоритма, понятие паракомпьютера.
3. Способ распараллеливания алгоритма Флойда

Утверждено на заседании кафедры информационных образовательных технологий и систем

Протокол № \_\_\_\_\_ от \_\_\_\_\_ г.

Заведующий кафедрой ИОТС \_\_\_\_\_ Капустин Д.А.

(подпись)

Экзаменатор

\_\_\_\_\_

(подпись)

доцент, Капустин Д.А.

(должность, ФИО преподавателя)

## **2. КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА**

### **2.1. Оценочные средства текущего контроля (типовые)**

Вопросы для текущего контроля:

1. Примеры использования параллельных вычислений в науке и технике.
2. Проблемы, связанные с использованием параллельных вычислений.
3. Понятия вычислительная сложность и относительного времени выполнения алгоритма.
4. Модель алгоритма, определение и примеры графа «операции - операнды» и расписания параллельного алгоритма.
5. Время выполнения, ускорение и эффективность параллельного алгоритма.
6. Понятия степени параллелизма компьютера и алгоритма, понятие паракомпьютера.
7. Понятие сверхлинейного ускорения и причины его возникновения.
8. Понятие средняя степени параллелизма алгоритма. Максимальное достижимое ускорение, закон Амдаля.
9. Этапы проектирования параллельного алгоритма.
10. Действия на этапах декомпозиции, проектирования коммуникаций, масштабирования подзадач, планирования вычислений.
11. Параллельные алгоритм умножения матрицы на вектор и матрицы на матрицу для систем с общей памятью.
12. Параллельный алгоритм умножения матрицы на вектор и матрицы на матрицу для систем с распределенной памятью.
13. Альтернативные параллельные алгоритмы матрично-векторного умножения: алгоритмы Виноградова и Фокса.
14. Теоретические оценки эффективности и ускорения алгоритмов умножения матриц.
15. Прямой и обратный проходы последовательного алгоритма метода Гаусса.
16. Идея параллельной реализации метода Гаусса для систем с общей памятью.
17. Способы декомпозиции матрицы СЛАУ при параллельной реализации метода Гаусса.
18. Параллельная реализация выбора ведущего элемента в методе Гаусса для систем с распределенной памятью.
19. Схема программной реализации параллельного варианта метода Гаусса с использованием MPI.
20. Теоретические оценки эффективности и ускорения параллельного алгоритма метода Гаусса.

### **2.2. Оценочные средства для промежуточной аттестации**

Вопросы для проведения аттестации

1. Основные шаги последовательного алгоритма метода сопряженных градиентов. Предобуславливание матрицы системы линейных уравнений. Вычислительная трудоемкость алгоритма.
2. Формат CSIR хранения матрицы СЛАУ.
3. Распараллеливание метода сопряженных градиентов для систем с общей распределенной памятью. Ускорение и эффективность параллельного алгоритма метода сопряженных градиентов.
4. Последовательный алгоритм метода минимальных невязок, вычислительная трудоемкость метода минимальных невязок.
5. Распараллеливание метода минимальных невязок. Ускорение и эффективность параллельного алгоритма метода минимальных невязок.
6. Понятие графа и минимального остовного дерева. Способы задания графов.
7. Задача поиска всех кратчайших путей. Алгоритм Флойда. Вычислительная трудоемкость алгоритма Флойда.
8. Способ распараллеливания алгоритма Флойда. Ускорение и эффективность параллельного алгоритма Флойда.
9. Задача нахождения минимального остовного дерева. Алгоритм Прима. Вычислительная трудоемкость алгоритма Прима.
10. Способ распараллеливания алгоритма Прима. Ускорение и эффективность параллельного алгоритма Флойда.
11. Пузырьковая сортировка и ее чет-нечетная модификация. Вычислительная трудоемкость алгоритма чет-нечетной сортировки.
12. Распараллеливание алгоритма чет-нечетной перестановки для систем с общей распределенной памятью. Ускорение и эффективность параллельного алгоритма чет-нечетной сортировки.
13. Последовательная сортировка Шелла, вычислительная трудоемкость алгоритма сортировка Шелла.
14. Распараллеливание сортировки Шелла для систем с общей распределенной памятью. Ускорение и эффективность параллельного алгоритма чет-нечетной сортировки..
15. Последовательная быстрая сортировка, вычислительная трудоемкость алгоритма сортировка Шелла.
16. Распараллеливание быстрой сортировки для систем с общей распределенной памятью. Ускорение и эффективность параллельного алгоритма быстрой сортировки.
17. Принципы построения генетических алгоритмов на примере задачи о ферзях. Вычислительная трудоемкость генетических алгоритмов.
18. Подходы к распараллеливанию генетического алгоритма. Теоретические оценки ускорения и эффективности распараллеливания генетических алгоритмов.
19. Что такое параллельные вычисления?
20. Какие преимущества предоставляют параллельные вычисления?
21. Какие основные типы параллельных вычислений существуют?
22. Какие языки программирования подходят для реализации параллельных вычислений?

23. Что такое конкурентное выполнение вычислений?
24. Какие методы синхронизации процессов используются в параллельных вычислениях?
25. Каковы основные проблемы параллельных вычислений?
26. Что такое поток (Thread)?
27. В чем разница между потоками и процессами?
28. Какие средства предоставляет язык программирования для работы с потоками?
29. Каковы преимущества и недостатки использования потоков?
30. Что такое многозадачность (Multitasking)?
31. Какие типы многозадачности существуют?
32. Как реализуется многозадачность в операционной системе?
33. Что такое распределенные вычисления?
34. Каковы преимущества и недостатки распределенных вычислений?
35. Какие протоколы используются в распределенных вычислениях?
36. Что такое кластерные вычисления?
37. Каковы преимущества и недостатки кластерных вычислений?
38. Что такое параллельные алгоритмы?
39. Какие методы параллельного программирования вы знаете?
40. Что такое локализация вычислений?
41. Каким образом можно уменьшить конфликты при доступе к общим ресурсам в параллельных вычислениях?
42. Какие методы декомпозиции используются в параллельных вычислениях?
43. Что такое динамическое распределение задач?
44. Что такое статическое распределение задач?
45. Каким образом можно увеличить производительность параллельных вычислений?
46. Что такое мультипроцессорность?
47. Какие типы архитектур мультипроцессорных систем существуют?
48. Что такое SIMD (Single Instruction, Multiple Data)?
49. Какие языки программирования поддерживают SIMD?
50. Каким образом работает SIMD-инструкция?
51. Что такое MIMD (Multiple Instruction, Multiple Data)?
52. Какие языки программирования поддерживают MIMD?
53. Каким образом работает MIMD-архитектура?
54. Что такое распараллеливание циклов?
55. Какие методы распараллеливания циклов существуют?
56. Каким образом оценивается эффективность распараллеливания циклов?
57. Что такое параллельная база данных?
58. Какие проблемы возникают при работе с параллельными базами данных?
59. Каким образом можно увеличить производительность параллельных баз данных?
60. Что такое параллельная обработка запросов?



61. Каким образом можно распараллелить обработку запросов в базе данных?
62. Что такое параллельный алгоритм сортировки?
63. Какие существуют методы параллельной сортировки данных?
64. Что такое параллельное программное обеспечение?
65. Какие инструменты используются для разработки параллельного программного обеспечения?
66. Каким образом происходит отладка параллельных программ?
67. Что такое параллельные архитектуры вычислительных систем?
68. Какие существуют основные типы параллельных архитектур?
69. Каким образом происходит обмен данными между процессорами в параллельной архитектуре?
70. Что такое разделение памяти в параллельных архитектурах?
71. Каким образом работает кэш-коерцитивность в параллельных архитектурах?
72. Что такое параллельный поиск?
73. Какие методы параллельного поиска данных существуют?
74. Что такое параллельная симуляция?
75. Каким образом можно распараллелить симуляцию системы?
76. Что такое параллельная оптимизация?
77. Каким образом происходит параллельная оптимизация процесса?