

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования «Кабардино-Балкарский государственный  
университет им. Х.М. Бербекова» (КБГУ)

ИНСТИТУТ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА И ЦИФРОВЫХ ТЕХНОЛОГИЙ  
КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

СОГЛАСОВАНО

Руководитель образовательной  
программы \_\_\_\_\_ М.М. Лафишева

« 12 » \_\_\_\_\_ 04 \_\_\_\_\_ 2023г.



ФОНД ОЦЕНОЧНЫХ СРЕДСТВ (ОЦЕНОЧНЫХ МАТЕРИАЛОВ)  
ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

«ТЕОРИЯ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ»

02.03.02 Фундаментальная информатика и информационные технологии  
(код и наименование направления подготовки)

«Проектирование систем искусственного интеллекта»  
(наименование профиля подготовки)

Бакалавр

Квалификация (степень) выпускника

Очная

Форма обучения

Нальчик - 2023

## СОДЕРЖАНИЕ

1. Перечень компетенций и этапы их формирования .....	3
2. Методические материалы и типовые контрольные задания, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы.....	6
1. Перечень контрольных заданий и иных материалов, необходимых для оценки знаний, умений, навыков и опыта деятельности .....	6
4. Вопросы к экзамену (5 семестр) по дисциплине «Системное и прикладное программное обеспечение» .....	17

## 1. Перечень компетенций и этапы их формирования

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ОПОП ВО по данному направлению подготовки 02.03.02 Фундаментальная информатика и информационные технологии (уровень бакалавриата):  
**профессиональных (ПКС):**

Коды	Содержание компетенций
ПКС-2	Способен создавать и исследовать новые математические модели в естественных науках, промышленности и бизнесе, с учетом возможностей современных информационных технологий, программирования и компьютерной техники

### Общая характеристика компетенции

**Тип компетенции:** общепрофессиональная компетенция выпускника образовательной программы по направлению подготовки высшего образования 02.03.02 Фундаментальная информатика и информационные технологии, уровень ВО бакалавр.

#### 1.1. Этапы формирования компетенций и средства оценивания

Результаты обучения (компетенции)	Индикаторы достижения компетенции	Освоенные показатели оценки результатов обучения	Виды оценочного материала, обеспечивающий формирование компетенций
ПКС-2. Способен создавать и исследовать новые математические модели в естественных науках, промышленности и бизнесе, с учетом возможностей современных информационных технологий, программирования и компьютерной техники	ПКС-2.1. Способен использовать основные методы проектирования и производства программного продукта и программных комплексов, их сопровождения, администрирования и развития (эволюции)	ПКС-2.1. З-1. Знает арсенал и области применения современных научных методов и информационных технологий, необходимых для решения задач, имеющих естественно-научное содержание и возникающих при выполнении профессиональных функций ПКС-2.1. У-1. Умеет описывать проблемы и ситуации профессиональной деятельности на основе знаний математического аппарата и естественнонаучных дисциплин и формулировать задачу профессиональной деятельности в области прикладной математики и информатики аппарата и естественнонаучных дисциплин ПКС-2.1. В-1. Владеет навыками производить статистические расчеты с применением соответствующих математических методов и информационных технологий, а	Типовые оценочные материалы для устного опроса (п. 5.1.1); типовые оценочные материалы для контрольной работы (п. 5.2.1); типовые тестовые задания (п. 5.2.2); типовые оценочные материалы к экзамену (п. 5.2.3)

		также проводить последующую аналитическую работу с полученными данными	
	<b>ПКС-2.2.</b> Способен использовать методы проектирования и производства программного продукта, принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного продукта	<b>ПКС-2.2.</b> 3-1. Знает методологии разработки программного обеспечения и технологии программирования <b>ПКС-2.2.</b> У-1. Умеет использовать существующие типовые решения и шаблоны проектирования программного обеспечения, применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов <b>ПКС-2.2.</b> В-1. Владеет навыками программирования элементов компьютерной графики и навыками создания правильных, геометрических и реалистичных изображений на экране компьютера	

## 1.2. Критерии формирования оценок на различных этапах их формирования

### Текущий и рубежный контроль

Этап (уровень)	Первый этап (уровень)	Второй этап (уровень)	Третий этап (уровень)
<b>Баллы</b>	36-50 баллов	51-60 баллов	61-70 баллов
<b>Характеристика</b>	Полное или частичное посещение аудиторных занятий. Частичное выполнение домашнего задания. Частичное выполнение заданий контрольных работ, тестовых заданий на оценку «удовлетворительно».	Полное или частичное посещение аудиторных занятий. Полное выполнение домашнего задания. Выполнение заданий на коллоквиуме на оценку «хорошо».	Полное посещение аудиторных занятий. Полное выполнение домашнего задания, заданий контрольных работ. Выполнение заданий на коллоквиуме на оценку «отлично».

На первом (начальном) этапе формирования компетенции формируются знания, умения и навыки, составляющие базовую основу компетенции, без которой невозможно ее дальнейшее развитие. Обучающийся воспроизводит термины, факты, методы, понятия, принципы и правила; решает учебные задачи по образцу.

На втором (основном) этапе формирования компетенции приобретает опыт деятельности, когда отдельные компоненты компетенции начинают «работать» в комплексе и происходит выработка индивидуального алгоритма продуктивных действий, направленных на достижение поставленной цели. На этом этапе обучающийся осваивает аналитические действия с предметными знаниями по конкретной дисциплине, способен самостоятельно решать учебные задачи, внося коррективы в алгоритм действий, осуществляя координирование хода работы, переносит знания и умения на новые условия.

Третий (завершающий) этап – это овладение компетенцией. Обучающийся способен использовать знания, умения, навыки при решении задач повышенной сложности и в нестандартных условиях. По результатам этого этапа обучающийся демонстрирует итоговый уровень сформированности компетенции.

**Промежуточная аттестация**  
**5 семестр-экзамен**

Семестр	Шкала оценивания			
	Неудовлетворительно (36-60 баллов)	Удовлетворительно (61-80 баллов)	Хорошо (81-90 баллов)	Отлично (91-100 баллов)
<b>1</b>	<p>Студент имеет 36-60 баллов по итогам текущего и рубежного контроля, на экзамене не дал полного ответа ни на один вопрос, не сделал пример.</p> <p>Студент имеет 36-45 баллов по итогам текущего и рубежного контроля, на экзамене дал полный ответ только на один вопрос, а пример сделан неправильно.</p>	<p>Студент имеет 36-50 баллов по итогам текущего и рубежного контроля, на экзамене дал полный ответ на один вопрос и частично (полностью) ответил на второй, а пример сделан не верно.</p> <p>Студент имеет 46-60 баллов по итогам текущего и рубежного контроля, на экзамене дал полный ответ на один вопрос или частично ответил на оба вопроса, а пример не сделан.</p> <p>Студент имеет по итогам текущего и рубежного контроля 61-70 баллов на экзамене не дал полного</p>	<p>Студент имеет 51-60 баллов по итогам текущего и рубежного контроля, на экзамене дал полный ответ на один вопрос и частично (полностью) ответил на второй. Пример сделан верно.</p> <p>Студент имеет 61 – 65 баллов по итогам текущего и рубежного контроля, на экзамене дал полный ответ на один вопрос и частично ответил на второй, и в примере есть недочеты, которые не повлияли на ответ.</p> <p>Студент имеет 66-70 баллов по</p>	<p>Студент имеет 61-70 баллов по итогам текущего и рубежного контроля, на экзамене дал полный ответ на один вопрос и частично (полностью) ответил на второй, и пример сделан правильно.</p> <p>Или же студент на оба вопроса ответил верно, а в задаче, есть неточности, которые не повлияли на ответ.</p>

		ответа ни на один вопрос. В решении примера есть грубая ошибка, которая повлияла на ответ, вследствие чего пример сделан не верно	итогам текущего и рубежного контроля, на экзамене дал полный ответ только на один вопрос. В примере есть неточности, которые не повлияли на ответ.	
--	--	---	--	--

**2. Методические материалы и типовые контрольные задания, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы**

**Перечень оценочных средств**

<b>№</b>	<b>Наименование оценочного средства</b>	<b>Краткая характеристика оценочного средства</b>	<b>Представление оценочного средства в фонде</b>
1.	Коллоквиум	Средство контроля усвоения учебного материала темы, раздела или разделов дисциплины, организованное как учебное занятие в виде собеседования преподавателя с обучающимися.	Вопросы по темам/разделам дисциплины
2.	Тест	Система стандартизированных заданий, позволяющая автоматизировать процедуру измерения уровня знаний и умений обучающегося.	Фонд тестовых заданий

**1. Перечень контрольных заданий и иных материалов, необходимых для оценки знаний, умений, навыков и опыта деятельности**

**3.1. Задания для коллоквиумов (вопросы для оценки компетенции (ПКС-2) :**

Тема 1. Введение. Цели, задачи и проблемы параллельных вычислений

2. Понятие параллельных вычислений. Условия достижения параллелизма.
3. Примеры параллельных вычислительных систем. Суперкомпьютеры. Рейтинг суперкомпьютеров. Тест LINPACK для оценки производительности.
4. Классификация вычислительных систем.
5. Типовые схемы коммуникации в многопроцессорных ВС.
6. Системные платформы для построения кластеров.

Тема 2. Моделирование и анализ параллельных алгоритмов

1. Модель вычислений в виде графа "операции - операнды".
2. Описание схемы выполнения параллельного алгоритма.

3. Определение времени выполнения параллельного алгоритма.
4. Показатели эффективности параллельного алгоритма.
5. Пример. Вычисление частных сумм последовательности чисел.
6. Оценка максимально достижимого параллелизма.
7. Анализ масштабируемости параллельных вычислений.

#### Тема 3. Оценка коммуникационной сложности параллельных алгоритмов

1. Общая характеристика механизмов передачи данных. Алгоритмы маршрутизации. Методы передачи данных.
2. Анализ трудоемкости основных операций передачи данных: между двумя процессорами сети, от одного процессора всем остальным, от всех процессоров всем процессорам, обобщенная передача данных.
3. Логическое представление топологии коммуникационной среды.
4. Оценка трудоемкости операций передачи данных для кластерных систем.

#### Тема 4. Принципы разработки параллельных методов

1. Моделирование параллельных программ.
2. Этапы разработки параллельных алгоритмов: разделение вычислений на независимые части; выделение информационных зависимостей; масштабирование набора подзадач; распределение подзадач между процессорами.
3. Параллельное решение гравитационной задачи N тел: разбор всех этапов разработки параллельного алгоритма.

#### Тема 5. Интерфейс передачи сообщений MPI

1. Основные понятия и определения.
2. Введение в разработку параллельных программ с использованием MPI: инициализация и завершение MPI-программ; определение количества и ранга процессов; передача сообщений; прием сообщений; первая параллельная программа с использованием MPI.
3. Определение времени выполнения MPI-программы. Обзор коллективных операций передачи данных.

#### Тема 6. Операции передачи данных между двумя процессами в MPI

1. Режимы передачи данных.
2. Организация неблокирующих обменов данными между процессами.
3. Одновременное выполнение передачи и приема.
4. Обобщенная передача данных от одного процесса всем процессам.
5. Обобщенная передача данных от всех процессов одному процессу.
6. Общая передача данных от всех процессов всем процессам.
7. Дополнительные операции редукации данных.

#### Тема 7. Производные типы данных в MPI

1. Понятие производного типа данных.
2. Способы конструирования производных типов данных: непрерывный способ конструирования; векторный способ конструирования; индексный способ конструирования; структурный способ конструирования.
3. Объявление производных типов и их удаление.
4. Формирование сообщений при помощи упаковки и распаковки данных.

#### Тема 8. Управление группами процессов и коммутаторами. Виртуальные топологии

1. Группы, управление группами. Создание новых групп на основе существующих с использованием операций объединения, пересечения и разности.

2. Коммуникаторы, создание и управление коммуникаторами. Коммуникаторы по умолчанию. Способы создания новых коммуникаторов: дублирование, подмножество процессов.
3. Декартовы топологии (решетки).
4. Топологии графа.

***Критерии формирования оценок по контрольным точкам (коллоквиум)***

«отличный (высокий) уровень компетенции» (5 баллов) - ставится в случае, когда обучающийся демонстрирует знание теоретического материала на 100%;

«хороший (нормальный) уровень компетенции» (4 баллов) - ставится в случае, когда обучающийся демонстрирует знание теоретического материала на 70%;

«удовлетворительный (минимальный, пороговый) уровень компетенции» (3 балла) – ставится в случае, когда обучающийся затрудняется с правильной формулировкой теоретического материала, дает неполный ответ, демонстрирует знание теоретического материала на 50%;

«неудовлетворительный (ниже порогового) уровень компетенции» (2 и менее баллов) – ставится в случае, когда обучающийся дает неверную формулировку теоретического материала, дает неверный ответ, демонстрирует незнание теоретического материала или знание материала менее чем на 40%.

**3.2. Оценочные материалы для контрольной работы:  
контролируемая компетенция «(ПКС-2)»:**

1. В чем состоят необходимые условия для возможности организации параллельных вычислений:

- (+) избыточность вычислительных устройств и независимость их функционирования;
- (-) организация режима разделения времени;
- (-) наличие сети передачи данных между процессорами.

2. Какие способы распределения данных между процессорами вычислительной системы?

- (+) ленточное разделение данных;
- (-) диагональное разделение матрицы;
- (+) блочное разделение данных.

3. Модель вычислений – это:

- (+) ациклический ориентированный граф;
- (-) бинарное дерево;
- (-) циклический ориентированный граф.

4. Для организации параллельных вычислений в вычислительных системах с распределенной памятью необходимо:

- (-) обеспечить информационное взаимодействие между процессорами;
- (+) выделить информационно независимые фрагменты вычислений, провести их программную реализацию, разместить полученные части программы на разных процессорах и затем организовать информационное взаимодействие между процессорами;
- (-) распределить исполняемые модули параллельной программы по узлам системы.

5. При выполнении параллельного алгоритма, основанного на разделении данных на горизонтальные полосы, сбор данных полученных результатов выполняется при помощи:

- (-) операции передачи сообщений типа "точка-точка";
- (+) операции обобщенного сбора данных;
- (-) операция редукации данных.

6. Под параллельной программой в рамках MPI понимается:

- (+) множество одновременно выполняемых процессов;
- (-) множество одновременно выполняемых потоков;
- (-) множество одновременно работающих процессоров.

7. При выполнении параллельного алгоритма, основанного на ленточной схеме разделения данных, основной коммуникационной операцией является:

- (-) операция обобщенного сбора данных;
- (+) операция циклического сдвига;
- (-) операция редукации данных.

8. Среди предусмотренных в составе MPI операций передачи сообщений различают:

- (+) парные и коллективные операции;
- (-) парные и групповые операции;
- (-) индивидуальные и коллективные операции.

9. Прием сообщения при помощи функции MPI\_Recv может быть инициирован:

- (+) до момента, в момент или после момента начала отправки сообщения;
- (-) только в момент начала отправки сообщения;
- (-) только после момента начала отправки сообщения.

10. Управление распределением нагрузки для процессоров необходимо:

- (+) для вычислительных систем с распределенной памятью;
- (-) для вычислительных систем с общей памятью;
- (-) для вычислительных систем с распределенной памятью и для систем с общей памятью.

11. Граф это:

- (+) множество точек (вершин) вместе с набором линий (дуг), которые соединяют определенные пары вершин
- (-) множество линий (дуг), соединенных друг с другом
- (-) множество точек (вершин) вместе с набором линий (дуг) определенного веса, которые соединяют определенные пары вершин

12. Какие способы распределения данных между процессорами вычислительной системы изложены в данной лекции?

- (+) ленточное разделение данных
- (-) диагональное разделение матрицы
- (+) блочное разделение данных

13. Вычислительный эксперимент в системе ПараЛаб – это:

- (-) выполнение параллельной программы при использовании только одного процессора
- (-) выполнение параллельной программы на реальной многопроцессорной вычислительной системе
- (+) демонстрация процесса решения задачи в режиме имитации параллельных вычислений

14. Модель вычислений – это:

- (+) ациклический ориентированный граф

- (-) бинарное дерево
- (-) циклический ориентированный граф

15. Алгоритмы маршрутизации определяют:

- (+) путь передачи данных от процессора-источника сообщения до процессора, к которому сообщение должно быть доставлено
- (-) длину наименьшего пути, который проходит сообщение в коммуникационной сети
- (-) все возможные пути передачи данных между процессорами

16. Распределение подзадач между процессорами должно быть выполнено таким образом, чтобы:

- (+) информационные связи между подзадачами были бы минимальными
- (-) информационные связи между подзадачами были бы максимальными
- (-) загруженность процессоров была бы минимальной

17. Для организации параллельных вычислений в вычислительных системах с распределенной памятью необходимо:

- (-) обеспечить информационное взаимодействие между процессорами
  - (+) выделить информационно независимые фрагменты вычислений, провести их программную реализацию, разместить полученные части программы на разных процессорах и затем организовать информационное взаимодействие между процессорами
  - (-) распределить исполняемые модули параллельной программы по узлам системы
- Какие способы распределения элементов матрицы между процессорами вычислительной системы изложены в данной лекции?
- (-) поэлементное разделение матрицы
  - (+) ленточное разделение матрицы
  - (+) блочное разделение матрицы

18. Какие схемы разделения данных используются при разработке параллельных алгоритмов умножения матриц?

- (+) ленточная схема разделения данных
- (+) блочная схема разделения данных
- (-) данные дублируются между процессорами

19. Какая схема разделения данных используется при реализации параллельного алгоритма Гаусса?

- (-) ленточная последовательная схема разделения данных
- (+) ленточная циклическая схема разделения данных
- (-) блочная схема разделения данных
- (-) данные дублируются между процессорами

20. Задача сортировки данных обычно формулируется как:

- (+) задача размещения элементов неупорядоченного набора значений в порядке монотонного возрастания или убывания
- (-) задача разделения элементов набора значений на несколько частей
- (-) задача разделения элементов набора значений с использованием некоторого ведущего значения

21. Какую компьютерную систему можно отнести к суперкомпьютерам:

- (+) систему с максимально-достижимыми на данный момент времени показателями производительности
- (-) компьютер, производительность которого превышает величины в 1 Tflops

(-) систему, способную решать сложные вычислительные задачи

Задача поиска всех кратчайших путей обычно формулируется как:

(+) для данного графа найти минимальные длины путей между каждой парой его вершин

(-) для данного графа найти минимальное расстояние среди всех пар его вершин

(-) для данного графа найти максимальную среди минимальных длин путей между всеми парами вершин

22. При выполнении параллельного алгоритма, основанного на разделении данных на горизонтальные полосы, сбор данных полученных результатов выполняется при помощи:

(-) операции передачи сообщений типа "точка-точка"

(+) операции обобщенного сбора данных

(-) операция редукции данных

23. В каком из режимов можно провести вычислительный эксперимент?

(+) в режиме имитации параллельных вычислений на обычном последовательном компьютере без использования дополнительных программных средств с визуализацией процесса решения

(-) в режиме локальных параллельных вычислений на последовательном компьютере пользователя с использованием библиотеки передачи сообщений MPI

(+) в режиме удаленного доступа к многопроцессорному вычислительному кластеру

Ускорение параллельных вычислений – это:

(+) отношение времени последовательного алгоритма ко времени параллельного решения задачи

(-) отношение времени последовательного алгоритма ко времени параллельного решения задачи при использовании максимально возможного количества процессоров

(-) отношение времени параллельного алгоритма ко времени последовательного решения задачи

24. Длительность времени передачи одного слова данных по одному каналу передачи данных определяется:

(+) полосой пропускания коммуникационных каналов в сети

(-) временем, необходимым для передачи служебных данных

(-) временем, необходимым для начальной подготовки передачи

25. Граф "подзадачи – сообщения" представляет собой:

(+) агрегированное представление графа информационных зависимостей

(-) детализированное представление графа информационных зависимостей

(-) агрегированное представление графа " процессы – каналы "

26. Под параллельной программой в рамках MPI понимается:

(+) множество одновременно выполняемых процессов

(-) множество одновременно выполняемых потоков

(-) множество одновременно работающих процессоров

27. При выполнении параллельного алгоритма, основанного на разделении матрицы на горизонтальные полосы, сбор данных результирующего вектора выполняется при помощи:

(-) операции передачи сообщений типа «точка-точка»

(+) операции обобщенного сбора данных

(-) операция редукции данных

28. При выполнении параллельного алгоритма, основанного на ленточной схеме разделения данных, основной коммуникационной операцией является:

- (-) операция обобщенного сбора данных
- (+) операция циклического сдвига
- (-) операция редукции данных

29. При выполнении параллельного алгоритма Гаусса основными коммуникационными операциями являются:

- (-) операция обобщенного сбора данных
- (+) операции широковещательной рассылки и редукции данных
- (-) операция циклического сдвига

30. Базовая операция "сравнить и переставить" состоит из:

- (+) сравнения пары значений из сортируемого набора данных и перестановки этих значений, если их порядок не соответствует условиям сортировки
- (-) сравнения пары значений из сортируемого набора данных и сохранения наименьшего из них
- (-) обмена имеющимися на процессорах  $P_i$  и  $P_j$  значениями, сравнения этих значений на каждом из процессоров и разделения данных между процессорами

31. Под кластером обычно понимается:

- (+) множество отдельных компьютеров, объединенных в сеть, для которых при помощи специальных аппаратно-программных средств обеспечивается возможность унифицированного управления, надежного функционирования и эффективного использования
- (-) множество отдельных компьютеров, объединенных в локальную вычислительную сеть
- (-) множество отдельных компьютеров, подключенных к сети Интернет

32. Число итераций параллельного алгоритма Флойда равно:

- (+)  $n$
- (-)  $\lceil \log_2 p \rceil$
- (-)  $n^2$

33. Для параллельных алгоритмов для систем с общей памятью при проведении вычислительных экспериментов может наблюдаться сверхлинейное ускорение. Каковы возможные причины достижения этого эффекта?

- (+) наличие у каждого из процессоров вычислительного сервера своей быстрой кэш-памяти
- (+) параллельный алгоритм может выполнять меньшее количество итераций
- (-) снижение времени передачи данных между потоками

34. При построении графических зависимостей для экспериментов, проведенных в режиме имитации, используются:

- (-) накопленные результаты выполненных экспериментов
- (+) теоретические оценки, применяемые в системе ПараЛаб
- (-) теоретические оценки, определенные пользователем системы

35. Каскадная схема используется для:

- (+) вычисления всех частных сумм последовательности числовых значений при четном количестве суммируемых элементов
- (-) вычисления суммы последовательности числовых значений

(-) вычисления всех частных сумм последовательности числовых значений

36. При использовании метода передачи сообщений:

(+) процессор-отправитель готовит весь объем данных для передачи, определяет процессор-получатель и запускает операцию пересылки данных, а процессор-получатель осуществляет прием полностью всех пересылаемых данных и только затем приступает к пересылке принятого сообщения далее по маршруту

(-) процесс-отправитель разбивает пересылаемые сообщения на блоки информации меньшего размера, при этом процессор-получатель может осуществлять пересылку данных по дальнейшему маршруту непосредственно сразу после приема очередной порции данных

(-) в зависимости от загруженности сети процесс-отправитель может, как разбить данные на блоки информации меньшего размера, так и отправить весь объем данных, при этом процессор-получатель может разбивать и собирать сообщение для пересылки по дальнейшему маршруту

37. Канал передачи данных можно рассматривать как:

(+) очередь сообщений, в которую один или несколько процессов могут отправлять пересылаемые данные и из которой процесс-адресат может извлекать сообщения, отправляемые другими процессами

(-) стек сообщений, в который один или несколько процессов могут отправлять пересылаемые данные и из которого процесс-адресат может извлекать сообщения, отправляемые другими процессами

(-) очередь сообщений, в которую только один процесс может отправлять пересылаемые данные и из которой процесс-адресат может извлекать сообщения, отправляемые другими процессами

38. Среди предусмотренных в составе MPI операций передачи сообщений различают:

(+) парные и коллективные операции

(-) парные и групповые операции

(-) индивидуальные и коллективные операции

39. Для эффективного выполнения параллельного алгоритма умножения матрицы на вектор, основанного на блочном разделении матрицы, необходимо, чтобы процессоры вычислительной системы были объединены в топологию:

(+) решетка или полный граф

(-) гиперкуб

(-) звезда

40. Для эффективного выполнения алгоритма Фокса необходимо, чтобы процессоры вычислительной системы были организованы в топологию:

(1) решетка или полный граф

(2) гиперкуб

(3) звезда

41. За основу организации параллельных вычислений при реализации метода сопряженных градиентов выбирается:

(-) одновременное выполнение итераций метода процессорами вычислительной системы

(+) распараллеливание операции умножения матрицы на вектор

(-) распараллеливание операции скалярного произведения векторов

42. Трудоемкость алгоритма пузырьковой сортировки оценивается выражением:

- (+)  $T \sim n^2$
- (-)  $T \sim n \log_2 n$
- (-)  $T \sim n^{1.25}$

43. В основе классификации вычислительных систем в систематике Флинна используются:

- (-) показатели производительности вычислительных систем
- (+) понятия потоков команд и данных
- (-) количество имеющихся процессоров и принцип разделения памяти между процессорами

44. Охватывающим деревом (или остовом) неориентированного графа называется:

- (+) подграф, который является деревом и содержит все вершины исходного графа
- (-) бинарное дерево, содержащее все вершины исходного графа
- (-) подграф, который является деревом минимального веса и содержит все вершины исходного графа

45. За счет чего увеличивается число передач данных между процессорами при блочном представлении сетки области расчетов на системах с распределенной памятью?

- (-) увеличение объема передаваемых данных
- (+) увеличение граничных строк
- (-) увеличение объема служебных передаваемых данных

46. Какие из перечисленных ниже классы задач поддерживает система имитационного моделирования ПараЛаб:

- (+) обработка графов
- (+) решение системы линейных уравнений
- (-) интегрирование уравнений математической физики

$$y = \sum_{i=1}^N a_i b_i$$

47. Пусть есть задача вычисления суммы следующего вида  $y = \sum_{i=1}^N a_i b_i$ . Пусть  $N = 8$  и применяется каскадная схема. Какая в этом случае минимально возможная высота дерева модели вычисления:

- (-) 3
- (+) 4
- (-) 5

48. Для рассылки от одного процессора всем остальным процессорам сети при использовании топологии типа гиперкуб достаточно ( $N = \log_2 p$ ):

- (+)  $N$ -этапной процедуры передачи данных
- (-)  $N \log_2 N$ -этапной процедуры передачи данных
- (-)  $N^2$ -этапной процедуры передачи данных

49. Выбор способа разделения вычислений на независимые части основывается:

- (+) а анализе вычислительной схемы решения исходной задачи
- (-) на результатах вычислительных экспериментов
- (-) на анализе графа "процессы – каналы"

50. Все данные для передачи в качестве сообщения MPI описываются с помощью триады:

- (+) адрес памяти, количество и тип элементов данных
- (-) адрес памяти, ранг процесса-отправителя, используемый коммуникатор
- (-) адрес памяти, ранг процесса-получателя, используемый коммуникатор

51. На основании результатов экспериментов, представленных в лекции, можно сказать, что наибольшее ускорение демонстрирует:

- (-) алгоритм, основанный на разделении матрицы на горизонтальные полосы
- (+) алгоритм, основанный на разделении матрицы на вертикальные полосы
- (-) алгоритм, основанный на блочном разделении матрицы

52. Из представленных в лекции алгоритмов, лучшей масштабируемостью обладает:

- (-) алгоритм, основанный на разнотипном ленточном разделении матриц
- (-) алгоритм, основанный на однотипном ленточном разделении матриц
- (+) алгоритм Фокса и алгоритм Кэннона

53. Из представленных в лекции алгоритмов, лучшей масштабируемостью обладает:

- (+) алгоритм Гаусса
- (-) метод сопряженных градиентов
- (-) оба алгоритма обладают приблизительно одинаковыми показателями масштабируемости

54. Параллельный вариант алгоритма Шелла состоит в следующем:

- (+) на первом этапе осуществляется взаимодействие процессоров, являющихся соседними в структуре гиперкуба, второй этап состоит в реализации обычных итераций параллельного алгоритма чет-нечетной перестановки
- (-) на первом этапе осуществляется взаимодействие процессоров, являющихся соседними в структуре полного графа, второй этап состоит в реализации обычных итераций последовательного алгоритма Шелла
- (-) на первой итерации метода осуществляется деление исходного набора данных на две части, все значения набора, меньшие некоего среднего элемента, переносятся в первый формируемый блок, все остальные значения образуют второй блок набора; на второй итерации сортировки описанные правила применяются рекурсивно для обоих сформированных блоков и т. д.

55. Типовые топологии сети передачи данных определяются:

- (-) только с учетом возможности технической реализации
- (+) с учетом возможности технической реализации и эффективного использования при решении вычислительно-трудоемких задач
- (-) только с учетом возможности эффективного использования при решении вычислительно-трудоемких задач

56. Количество выполняемых операций при определении номера ближайшей вершины до охватывающего дерева и корректировке расстояний после расширения МОД ограничивается сверху величиной:

- (+)  $\lceil n/p \rceil$
- (-)  $\lfloor n/\log p \rfloor$
- (-)  $\lceil \log n/p \rceil$

57. На каких топологиях сети в системе ParaLab реализована быстрая сортировка:

- (+) гиперкуб
- (-) кольцо
- (-) линейка

$$y = \prod_{i=1}^N a_i$$

58. Пусть есть задача вычисления произведения всех элементов вектора  $y = \prod_{i=1}^N a_i$ . Пусть  $N = 6$  и применяется каскадная схема с минимально возможной высотой дерева модели вычисления. Чему в этом случае равно ускорение при использовании неограниченного числа вычислительных элементов:

- (+) 2
- (-) 2,5
- (-) 3

59. Уплотнение дуг это:

- (+) максимальное количество дуг логической топологии, отображаемых в одну линию передачи физической топологии
- (-) отношение количества вершин в логической и физической топологиях
- (-) путь максимальной длины физической топологии, на который отображается дуга логической топологии

60. Для локальной схемы передачи данных характерно:

- (+) выполнение взаимодействий только между небольшим числом подзадач, располагаемых, как правило, на соседних процессорах
- (-) выполнение взаимодействий только с соседними подзадачами
- (-) отсутствие взаимодействия между подзадачами

61. Рассмотрим задачу перемножения матрицы на вектор. Пусть размер перемножаемой матрицы  $100 \times 100$ . На вычислительной системе все операции сложения и умножения выполняются одинаковое время  $\tau = 2$  нсек. Латентности сети  $\alpha = 50$  нсек. Пропускная способность сети 60 Мбайт/сек. Элементы матрицы имеют тип double и занимают  $w = 8$  байт. Если при распараллеливании использовать разделение матрицы на строки чему будет равно теоретическое ускорение при использовании 2 процессоров:

- (-) 3
- (-) 1.5
- (+) 1.98

62. Рассмотрим задачу перемножения матриц. Пусть размер перемножаемой матрицы  $100 \times 100$ . На вычислительной системе все операции сложения и умножения выполняются одинаковое время  $\tau = 2$  нсек. Латентности сети  $\alpha = 500$  нсек. Пропускная способность сети  $\beta = 50$  Мбайт/сек. Элементы матрицы имеют тип double и занимают  $w = 8$  байт. Если при распараллеливании использовать разделение матрицы на ленты, чему будет равно теоретическое ускорение при использовании 4 процессоров:

- (-) 3,9
- (-) 2,5
- (+) 1,8

63. Рассмотрим задачу поиска решения системы линейных уравнений. Пусть размер матрицы системы линейных уравнений  $100 \times 100$ . На вычислительной системе все операции сложения и умножения выполняются одинаковое время  $\tau = 2$  нсек. Латентности сети  $\alpha = 500$  нсек. Пропускная способность сети  $\beta = 50$  Мбайт/сек. Элементы матрицы системы линейных уравнений имеют тип double и занимают  $w = 8$  байт. Если при распараллеливании алгоритма Гауса использовалось 4 процессора, то какое в этом случае достигается теоретическое ускорение:

- (-) 3,1

- (-) 1,5  
(+) 0,18

**Критерии формирования оценок по тестовым заданиям:**

По итогам выполнения тестовых заданий оценка производится по пятибалльной шкале. При правильных ответах на:

- 89-100% заданий – «5» (баллов);
- 70-88% заданий – «4» (баллов);
- 50-69% заданий – «3» (балла);
- 30-49% заданий – «2» (балла);
- 10-29% заданий – «1» (балл);
- менее 10% заданий – «0» (баллов).

**4. Вопросы к зачету (8 семестр) по дисциплине «Теория параллельных вычислений»**

Вопрос	Код компетенции (согласно РПД)
1. Понятие параллельных вычислений. Условия достижения параллелизма. Конвейерная организация вычислений. Суперкомпьютеры.	ПКС-2
2. Классификация вычислительных систем.	ПКС-2
3. Топологии сети передачи данных. Характеристики топологии сети.	ПКС-2
4. Моделирование и анализ параллельных вычислений: модель вычислений в виде графа, операции, операнды.	ПКС-2
5. Описание схемы выполнения параллельного алгоритма.	ПКС-2
6. Определение времени выполнения параллельного алгоритма.	ПКС-2
7. Показатели эффективности параллельного алгоритма. Пример: вычисление общей суммы набора значений.	ПКС-2
8. Определение показателей эффективности в задаче вычисления частных сумм набора значений.	ПКС-2
9. Оценка максимально достижимого параллелизма. Закон Амдаля. Закон Густавсона-Барсиса.	ПКС-2
10. Анализ масштабируемости параллельных вычислений.	ПКС-2
11. Оценка коммуникационной сложности параллельных алгоритмов. Временные характеристики коммуникационных операций. Методы передачи данных.	ПКС-2
12. Логическое представление топологии коммуникационной среды.	ПКС-2
13. Оценка трудоемкости операций передачи данных для кластерных систем.	ПКС-2
14. Схема разработки параллельных алгоритмов. Краткая характеристика этапов.	ПКС-2
15. Моделирование параллельных программ. Основные виды и особенности моделей.	ПКС-2
16. Этапы разработки параллельных алгоритмов: разделение вычислений на независимые части.	ПКС-2
17. Этапы разработки параллельных алгоритмов: выделение информационных зависимостей.	ПКС-2
18. Этапы разработки параллельных алгоритмов: масштабирование набора подзадач.	ПКС-2
19. Этапы разработки параллельных алгоритмов: распределение подзадач между процессорами.	ПКС-2

20. MPI: общая характеристика. Основные понятия и определения в MPI: процесс; ранг процесса.	ПКС-2
21. Основные понятия и определения в MPI: операции передачи данных; группы и коммутаторы; типы данных; виртуальные топологии.	ПКС-2
22. Структура параллельной программы на базе MPI. Инициализация, финализация программы. Определение количества процессов и ранга процесса.	ПКС-2
23. Операции парного обмена данными в MPI.	ПКС-2
24. Виды коллективных операций обмена данными в MPI.	ПКС-2
25. Синхронизация вычислений в MPI. Назначение синхронизации.	ПКС-2
26. Режимы передачи данных в операциях парного обмена. Стандартный, синхронный, буферизованный режимы передачи данных, режим передачи по готовности.	ПКС-2
27. Производные типы данных в MPI: понятие производного типа, карта типа, характеристики производного типа.	ПКС-2
28. Производные типы данных в MPI: способы конструирования.	ПКС-2
29. Производные типы данных в MPI: формирование сообщений при помощи упаковки и распаковки данных.	ПКС-2
30. Группы и коммутаторы в MPI: назначение, способы создания, удаление.	ПКС-2
31. Виды параллелизма: параллелизм по данным, функциональный параллелизм.	ПКС-2
32. OpenMP: понятие, назначение, основной принцип реализации параллелизма. Многопоточность. Сравнение OpenMP и MPI.	ПКС-2
33. Модель параллельной программы OpenMP. Схема выполнения параллельной программы OpenMP. Структура параллельной программы. Процесс выполнения программы.	ПКС-2
34. Директивы OpenMP: классификация. Опции директив. Функции OpenMP.	ПКС-2
35. Модель данных OpenMP. Гонка данных, причины ее возникновения.	ПКС-2
36. Распределение работы между потоками: низкоуровневое распараллеливание, параллельные циклы; параллельные секции.	ПКС-2