

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Сыров Игорь Анатольевич
Должность: Директор
Дата подписания: 21.08.2025 20:05:35
Уникальный программный ключ:
b683afe664d7e9f64175886cf9626a198149ad36

СТЕРЛИТАМАКСКИЙ ФИЛИАЛ
ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО
УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
«УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ»

Факультет
Кафедра

Математики и информационных технологий
Математического моделирования

Оценочные материалы по дисциплине (модулю)

дисциплина

Параллельное программирование

*Блок Б1, часть, формируемая участниками образовательных отношений,
Б1.В.ДВ.01.02*

цикл дисциплины и его часть (обязательная часть или часть, формируемая участниками образовательных отношений)

Направление

01.03.02

Прикладная математика и информатика

код

наименование направления

Программа

Программирование мобильных, облачных и интеллектуальных систем

Форма обучения

Очная

Для поступивших на обучение в
2020 г.

Разработчик (составитель)

к.ф.-м.н., доцент

Викторов С. В.

ученая степень, должность, ФИО

1. Перечень компетенций, индикаторов достижения компетенций и описание показателей и критериев оценивания результатов обучения по дисциплине (модулю)	3
2. Оценочные средства, необходимые для оценки результатов обучения по дисциплине (модулю)	6
3. Методические материалы, определяющие процедуры оценивания результатов обучения по дисциплине (модулю), описание шкал оценивания	26

1. Перечень компетенций, индикаторов достижения компетенций и описание показателей и критериев оценивания результатов обучения по дисциплине (модулю)

Формируемая компетенция (с указанием кода)	Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине (модулю)	Показатели и критерии оценивания результатов обучения по дисциплине (модулю)				Вид оценочного средства
			1	2	3	4	
			неуд.	удовл.	хорошо	отлично	
ПК-3. Способен к разработке параллельных алгоритмов и программ для решения профессиональных задач	ПК-3.1. Знать основные понятия и способы построения параллельных алгоритмов и программ для решения задач профессиональной деятельности; принципы построения, организации, архитектуры и структуры вычислительных систем параллельной обработки данных	Обучающийся должен: знать основные понятия и способы построения параллельных алгоритмов и программ для решения задач профессиональной деятельности	Отсутствие знаний об основных понятиях и способах построения параллельных алгоритмов и программ для решения задач профессиональной деятельности	Неполные сформированные систематические представления об основных понятиях и способах построения параллельных алгоритмов и программ для решения задач профессиональной деятельности	Сформированные, но содержащие отдельные пробелы систематического представления об основных понятиях и способах построения параллельных алгоритмов и программ для решения задач профессиональной деятельности	Сформированные в полном объеме систематические представления об основных понятиях и способах построения параллельных алгоритмов и программ для решения задач профессиональной деятельности	Устный опрос

	ПК-3.2. Уметь строить и программно реализовывать алгоритмы параллельного вычисления для получения, хранения, переработки информации при решении профессиональных задач; проводить обоснованный выбор компьютерных систем параллельной обработки данных	Обучающийся должен: уметь строить и программно реализовывать алгоритмы параллельного вычисления для получения, хранения, переработки информации при решении профессиональных задач	Отсутствие умений строить и программно реализовывать алгоритмы параллельного вычисления для получения, хранения, переработки информации при решении профессиональных задач	В целом успешное, но не систематическое умение строить и программно реализовывать алгоритмы параллельного вычисления для получения, хранения, переработки информации при решении профессиональных задач	Сформированно е, но содержащее отдельные пробелы в умении строить и программно реализовывать алгоритмы параллельного вычисления для получения, хранения, переработки информации при решении профессиональных задач	Сформированно е в полном объеме умение строить и программно реализовывать алгоритмы параллельного вычисления для получения, хранения, переработки информации при решении профессиональных задач	Тестовые задания
	ПК-3.3. Владеть теоритическими и практическими навыками программирования параллельных алгоритмов для решения	Обучающийся должен: владеть теоретическими и практическими навыками программирования параллельных	Отсутствие теоретических и практических навыков программирования параллельных алгоритмов для решения профессиональ	В целом успешное, но непоследовательное владение теоретическими и практическими навыками программирования параллельных алгоритмов для	В целом успешное, но содержащее отдельные пробелы во владении теоретическими и практическими навыками	Успешное и последовательное владение теоретическими и практическими навыками программирования параллельных	лабораторная работа; практическое задание; контрольная работа.

	профессиональ ных задач; навыками работы с программными средствами параллельной обработки данных.	алгоритмов для решения профессиональ ных задач	ных задач	решения профессиональн ых задач	программирова ния параллельных алгоритмов для решения профессиональ ных задач	алгоритмов для решения профессиональ ных задач	
--	---	---	-----------	---------------------------------------	---	---	--

2. Оценочные средства, необходимые для оценки результатов обучения по дисциплине (модулю)

Задания для устного опроса

Задания для устного опроса предназначены для оценки уровня сформированности компетенции ПК-3
(индикатор достижения компетенции ПК-3.1).

1. В чем заключаются основные способы достижения параллелизма?
2. В чем могут состоять различия параллельных вычислительных систем?
3. Что положено в основу классификации Флинна?
4. В чем состоит принцип разделения многопроцессорных систем на мультипроцессоры и мультикомпьютеры?
5. Какие классы систем известны для мультипроцессоров?
6. В чем состоят положительные и отрицательные стороны симметричных мультипроцессоров?
7. Какие классы систем известны для мультикомпьютеров?
8. В чем состоят положительные и отрицательные стороны кластерных систем?
9. Какие топологии сетей передачи данных наиболее широко используются при построении многопроцессорных систем?
10. В чем состоят особенности сетей передачи данных для кластеров?
11. Каковы основные характеристики сетей передачи данных?
12. Какие системные платформы могут быть использованы для построения кластеров?
13. Как определяется модель «операции – операнды»?
14. Как определяется расписание для распределения вычислений между процессорами?
15. Как определяется время выполнения параллельного алгоритма?
16. Какое расписание является оптимальным?
17. Как определить минимально возможное время решения задачи?
18. Что понимается под паракомпьютером и для чего может оказаться полезным данное понятие?
19. Какие оценки следует использовать в качестве характеристики времени последовательного решения задачи?
20. Как определить минимально возможное время параллельного решения задачи по графу «операнды – операции»?
21. Какие зависимости могут быть получены для времени параллельного решения задачи при увеличении или уменьшении числа используемых процессоров?
22. При каком числе процессоров могут быть получены времена выполнения параллельного алгоритма, сопоставимые по порядку с оценками минимально возможного времени решения задачи?
23. Как определяются понятия ускорения и эффективности?
24. Возможно ли достижение сверхлинейного ускорения?
25. В чем состоит противоречивость показателей ускорения и эффективности?
26. Как определяется понятие стоимости вычислений?
27. В чем состоит понятие стоимостно-оптимального алгоритма?
28. В чем заключается проблема распараллеливания последовательного алгоритма суммирования числовых значений?

29. В чем состоит каскадная схема суммирования? С какой целью рассматривается модифицированный вариант данной схемы?
30. В чем состоит различие показателей ускорения и эффективности для рассматриваемых вариантов каскадной схемы суммирования?
31. В чем состоит параллельный алгоритм вычисления всех частных сумм последовательности числовых значений?
32. Как формулируется закон Амдаля? Какой аспект параллельных вычислений позволяет учесть данный закон?
33. Какие предположения используются для обоснования закона Густавсона – Барсиса?
34. Как определяется функция изоэффективности?
35. Какой алгоритм является масштабируемым? Приведите примеры методов с разным уровнем масштабируемости.
36. Какие основные характеристики используются для оценки топологии сети передачи данных? Приведите значения характеристик для конкретных типов коммуникационных структур (полный граф, линейка, решетка и др.).
37. Какие основные методы применяются при маршрутизации передаваемых данных по сети?
38. В чем состоят основные методы передачи данных? Приведите для этих методов аналитические оценки времени выполнения.
39. Какие операции передачи данных могут быть выделены в качестве основных?
40. В чем состоят алгоритмы выполнения передачи данных от одного процессора всем процессорам сети для топологий кольца, решетки и гиперкуба? Приведите оценки временной трудоемкости для этих алгоритмов.
41. В чем состоят алгоритмы выполнения передачи данных от всех процессоров всем процессорам сети для топологий кольца, решетки гиперкуба? Приведите оценки временной трудоемкости для этих алгоритмов.
42. В чем состоят возможные алгоритмы выполнения операции редукции? Какой из алгоритмов является наилучшим по времени выполнения?
43. В чем состоит алгоритм выполнения операции циклического сдвига?
44. В чем состоит полезность использования логических топологий? Приведите примеры алгоритмов логического представления структуры коммуникационной сети.
45. В чем состоит различие моделей для оценки времени выполнения операций передачи данных в кластерных вычислительных системах? Какая модель является более точной? Какая модель может быть использована для предварительного анализа временной трудоемкости коммуникационных операций?
46. В чем состоят исходные предположения для возможности применения рассмотренной в лекции методики разработки параллельных алгоритмов?
47. Каковы основные этапы проектирования и разработки методов параллельных вычислений?
48. Как определяется модель «подзадачи – сообщения»?
49. Как определяется модель «процессы – каналы»?
50. Какие основные требования должны быть обеспечены при разработке параллельных алгоритмов?
51. В чем состоят основные действия на этапе выделения подзадач?
52. Каковы основные действия на этапе определения информационных зависимостей?
53. В чем состоят основные действия на этапе масштабирования имеющегося набора подзадач?

54. В чем состоят основные действия на этапе распределения подзадач по процессорам вычислительной системы?
55. Как происходит динамическое управление распределением вычислительной нагрузки при помощи схемы «менеджер – исполнитель»?
56. Какой метод параллельных вычислений был разработан для решения гравитационной задачи N-тел?
57. Какой способ выполнения операции обобщенного сбора данных является более эффективным?
58. Какой минимальный набор средств является достаточным для организации параллельных вычислений в системах с распределенной памятью?
59. В чем состоит важность стандартизации средств передачи сообщений?
60. Что следует понимать под параллельной программой?
61. В чем различие понятий процесса и процессора?
62. Какой минимальный набор функций MPI позволяет начать разработку параллельных программ?
63. Как описываются передаваемые сообщения?
64. Как можно организовать прием сообщений от конкретных процессов?
65. Как определить время выполнения MPI-программы?
66. В чем различие парных и коллективных операций передачи данных?
67. Какая функция MPI обеспечивает передачу данных от одного процесса всем процессам?
68. Что понимается под операцией редукции?
69. В каких ситуациях следует применять барьерную синхронизацию?
70. Какие режимы передачи данных поддерживаются в MPI?
71. Как организуется неблокирующий обмен данными в MPI?
72. В чем состоит понятие тупика? Когда функция одновременного выполнения передачи и приема гарантирует отсутствие тупиковых ситуаций?
73. Какие коллективные операции передачи данных предусмотрены в MPI?
74. Что понимается под производным типом данных в MPI?
75. Какие способы конструирования типов имеются в MPI?
76. В каких ситуациях может быть полезна упаковка и распаковка данных?
77. Что понимается в MPI под коммуникатором?
78. Для чего может потребоваться создание новых коммуникаторов?
79. Что понимается в MPI под виртуальной топологией?
80. Какие виды топологий предусмотрены в MPI?
81. Для чего может оказаться полезным использование виртуальных топологий?
82. В чем состоят особенности разработки параллельных программ с использованием MPI на алгоритмическом языке Fortran?
83. Какие основные дополнительные возможности предусмотрены в стандарте MPI-2?
84. Назовите основные способы распределения элементов матрицы между процессорами вычислительной системы.
85. В чем состоит постановка задачи умножения матрицы на вектор?
86. Какова вычислительная сложность последовательного алгоритма умножения матрицы на вектор?
87. Почему при разработке параллельных алгоритмов умножения матрицы на вектор допустимо дублировать вектор-операнд на все процессоры?
88. Какие подходы могут быть предложены для разработки параллельных алгоритмов умножения матрицы на вектор?
89. Представьте общие схемы рассмотренных параллельных алгоритмов умножения матрицы на вектор.

5) Какой из вариантов общепринятых сокращений не относится к параллельному программированию?

- 1) *FIFA* 2) *MPI* 3) *CUDA*

6) В каком году был разработан программный интерфейс передачи сообщений MPI?

- 1) 1994 2) 1995 3) 2005

7) Какая из указанных функций инициализирует параллельный режим программы?

- 1) *MPI_Init* 2) *MPI_Initialize* 3) *MPI_Config*

8) Какая подпрограмма вызывается для определения числа процессов в заданной группе?

- 1) *MPI_Comm_rank*
2) *MPI_Comm_size*
3) *MPI_Comm_count*

9) Какой из представленных вариантов правильно характеризует различные виды функций библиотеки MPI, определяющие коммуникационные операции?

- 1) Парные и коллективные
2) Парные и непарные
3) Коллективные и неколлективные

10) Выберите верное утверждение?

1) Коллективные операции передачи данных не могут выполняться в рамках одного коммуникатора

2) Парные операции передачи данных применяются одновременно для всех процессов коммуникатора

3) Парные операции передачи данных выполняются для процессов, принадлежащих одному и тому же коммуникатору

11) Все имеющиеся в параллельной программе процессы входят в состав создаваемого по умолчанию коммуникатора с идентификатором _____

- 1) *MPI_COMM_GLOBAL*
2) *MPI_COMM_GENERAL*
3) *MPI_COMM_WORLD*

12) Какая функция завершает параллельную часть программы?

- 1) *MPI_Close* 2) *MPI_Finalize* 3) *MPI_Stop*

13) Какие функции являются обязательными и должны быть выполнены каждым процессом параллельной программы?

- 1) *MPI_Init* и *MPI_Initialized*
2) *MPI_Init* и *MPI_Finalize*
3) *MPI_Initialized* и *MPI_Finalize*

14) Установите соответствие?

- | | |
|-------------------------|--|
| 1) <i>MPI_Comm_size</i> | а) определение числа процессов |
| 2) <i>MPI_Comm_rank</i> | б) определение идентификатора процесса |
| 3) <i>MPI_Send</i> | в) функция отправки сообщения |

- 15) Какая функция выполняется внутри вызывающего процесса. Ее завершение не требует коммуникаций.
- 1) нелокальная функция
 - 2) локальная функция
 - 3) глобальная функция
- 16) Для завершения какой функции требуется выполнение MPI-процедуры другим процессом.
- 1) глобальная функция
 - 2) нелокальная функция
 - 3) локальная функция
- 17) Какая функция должна выполняться всеми процессами группы. Несоблюдение этого условия может приводить к зависанию задачи.
- 1) локальная функция
 - 2) нелокальная функция
 - 3) глобальная функция
- 18) Верно ли утверждение: «Типы MPI_BYTE и MPI_PACKED используются для передачи двоичной информации без дополнительных преобразований»?
- 1) да
 - 2) нет
- 19) Какая функция позволяет определить системное время?
- 1) MPI_Time
 - 2) MPI_Gettimer
 - 3) MPI_Wtime
- 20) MPI_Isend – это функция с блокирующим режимом или нет?
- 1) нет
 - 2) да
- 21) Что означает префикс S перед базовой функцией?
- 1) согласованный
 - 2) буферизованный режим
 - 3) синхронный режим
- 22) Что означает префикс B перед базовой функцией?
- 1) синхронный режим
 - 2) буферизованный режим
 - 3) базовый режим
- 23) Что означает префикс R перед базовой функцией?
- 1) несинхронный режим
 - 2) согласованный режим
 - 3) синхронный режим
- 24) Установить соответствие?
- | | |
|-----------|----------------------------|
| 1) source | а) адрес отправителя |
| 2) dest | б) адрес получателя |
| 3) tag | г) идентификатор сообщения |
- 25) Что делает операция MPI_Sendrecv?
- 1) объединяет все процессы в приложении
 - 2) возвращает количество процессов в области связи коммутатора
 - 3) совмещает выполнение операций передачи и приема
- 26) Установите соответствие для названий параметров функции int MPI_Init(int *argc, char ***argv) и их значений

- 1) argc а) указатель на количество параметров командной строки
- 2) argv б) параметры командной строки

27) Какие функции не возвращают в качестве своего значения код завершения?

- 1) MPI_Wtime и MPI_Wtick
- 2) MPI_Wtime и MPI_Recv
- 3) MPI_Init и MPI_Send

28) Сколько параметров имеет функция MPI_Comm_rank?

- 1) 2
- 2) 1
- 3) 0

29) Какой из указанных вариантов в классификации параллельных компьютеров М. Флину (основанной на понятии потока последовательности команд или данных, обрабатываемых процессором), является описанием который соответствует «Одиночный поток команд и одиночный поток данных».

- 1) SISD - это обычные последовательные компьютеры
- 2) SIMD - большинство современных ЭВМ относятся к этой категории
- 3) MISD - вычислительных машин такого класса мало
- 4) MIMD -это реализация нескольких потоков команд и потоков данных

30) MPI - это ...

- 1) специальная ОС для параллельного программирования
- 2) интерфейс, содержащий набор функций, типов и констант для параллельного программирования
- 3) модуль параллельной обработки в системе
- 4) организация, координирующая разработку параллельных интерфейсов

31) Какие действия связаны с выполнением функции MPI_Send(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

- 1) принимаются данные целого типа, размещенные в памяти процесса с номером 0 начиная с адреса, значение которого хранит переменная A
- 2) посылаются данные целого типа, размещенные в памяти процесса с номером 0 начиная с адреса, значение которого хранит переменная A
- 3) посылаются данные целого типа, размещенные в памяти текущего процесса начиная с адреса, значение которого хранит переменная A

32) Какие действия связаны с выполнением функции MPI_Send(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

- 1) посылаются данные целого типа, размещенные в памяти текущего процесса начиная с адреса, значение которого хранит переменная A
- 2) посылаются данные целого типа, размещенные в памяти процесса с номером 10 начиная с адреса, значение которого хранит переменная A
- 3) принимаются данные целого типа, размещенные в памяти процесса с номером 10 начиная с адреса, значение которого хранит переменная A

33) Какие действия связаны с выполнением функции MPI_Send(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

- 1) принимаются данные целого типа, размещенные в памяти процесса с идентификатором MPI_COMM_WORLD начиная с адреса, значение которого хранит переменная A
- 2) посылаются данные целого типа, размещенные в памяти процесса с идентификатором MPI_COMM_WORLD начиная с адреса, значение которого хранит переменная A
- 3) посылаются данные целого типа, размещенные в памяти текущего процесса начиная с адреса, значение которого хранит переменная A

34) Какие действия связаны с выполнением функции MPI_Send(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

- 1) иницируется отправка сообщения с идентификатором 0 для процесса с номером 10
- 2) иницируется отправка сообщения с идентификатором 10 для процесса с номером 0
- 3) иницируется прием сообщения с идентификатором 0 для процесса с номером 10

35) Какие действия связаны с выполнением функции MPI_Send(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

- 1) иницируется прием сообщения с идентификатором A для процесса с номером 10
- 2) иницируется отправка сообщения с идентификатором A для процесса с номером 10
- 3) иницируется отправка сообщения с идентификатором 10 для процесса с номером 0

36) Какие действия связаны с выполнением функции MPI_Send(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

- 1) иницируется прием сообщения с идентификатором 10 для процесса с номером A
- 2) иницируется отправка сообщения с идентификатором 10 для процесса с номером A
- 3) иницируется отправка сообщения с идентификатором 10 для процесса с номером 0

37) Какие действия связаны с выполнением функции MPI_Send(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

- 1) процессы с номерами от 1 до 10 принимают 10 значений целой переменной A, хранящейся в памяти процесса 0
- 2) посылаются 10 значений целой переменной A из памяти процесса 0 процессам с номерами от 1 до 10
- 3) посылаются массив из 10 элементов целого типа процессу с номером 0

38) Какие действия связаны с выполнением функции MPI_Send(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

- 1) переменная целого типа A, хранящаяся в памяти процесса с номером 0 принимает значение 10
- 2) в переменную целого типа A, хранящуюся в памяти процесса с номером 0 посылается значение 10
- 3) посылается массив из 10 элементов целого типа процессу с номером 0

39) Какие действия связаны с выполнением функции MPI_Send(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

- 1) процесс с номером 10 принимает элементы массива A с индексами от 0 до 10
- 2) процессу с номером 10 посылаются элементы массива A с индексами от 0 до 10
- 3) посылается массив из 10 элементов целого типа процессу с номером 0

40) Какие действия связаны с выполнением функции MPI_Send(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

- 1) посылаются данные из памяти текущего процесса в память процесса с номером 0
- 2) посылаются данные из памяти процесса с номером 0 в память текущего процесса
- 3) принимаются данные из памяти процесса с номером 0 в память текущего процесса

41) Какие действия связаны с выполнением функции MPI_Send(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

- 1) принимаются данные из памяти процесса с номером 0 в память процесса с номером 10
- 2) посылаются данные из памяти процесса с номером 0 в память процесса с номером 10
- 3) посылаются данные из памяти текущего процесса в память процесса с номером 0

42) Какие действия связаны с выполнением функции MPI_Send(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

- 1) посылаются данные из памяти процесса с номером A в память процесса с номером 10
- 2) посылаются данные из памяти текущего процесса в память процесса с номером 0
- 3) принимаются данные из памяти процесса с номером A в память процесса с номером 10

43) Какие действия связаны с выполнением функции MPI_Recv(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

- 1) посылаются данные целого типа, размещенные в памяти процесса с номером 0 начиная с адреса, значение которого хранит переменная A
- 2) принимаются данные целого типа, в область памяти текущего процесса, начиная с адреса, хранящегося в переменной A

3) принимаются данные целого типа в область памяти процесса с номером 0 начиная с адреса, значение которого хранит переменная A

44) Какие действия связаны с выполнением функции MPI_Recv(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

1) принимаются данные целого типа в область памяти процесса с номером 10, начиная с адреса, хранящегося в переменной A

2) посылаются данные целого типа, размещенные в памяти процесса с номером 10 начиная с адреса, значение которого хранит переменная A

3) принимаются данные целого типа, в область памяти текущего процесса, начиная с адреса, хранящегося в переменной A

45) Какие действия связаны с выполнением функции MPI_Recv(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

1) посылаются данные целого типа, размещенные в памяти процесса с идентификатором MPI_COMM_WORLD начиная с адреса, значение которого хранит переменная A

2) принимаются данные целого типа, в область памяти текущего процесса, начиная с адреса, хранящегося в переменной A

3) принимаются данные целого типа в область памяти процесса с идентификатором MPI_COMM_WORLD, начиная с адреса, значение которого хранит переменная A

46) Какие действия связаны с выполнением функции MPI_Recv(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

1) инициируется прием сообщения с идентификатором 0 от процесса с номером 10

2) инициируется отправка сообщения с идентификатором 0 для процесса с номером 10

3) инициируется прием сообщения с идентификатором 10 от процесса с номером 0

47) Какие действия связаны с выполнением функции MPI_Recv(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

1) инициируется отправка сообщения с идентификатором A для процесса с номером 10

2) инициируется прием сообщения с идентификатором 10 от процесса с номером 0

3) инициируется прием сообщения с идентификатором A от процесса с номером 10

48) Какие действия связаны с выполнением функции MPI_Recv(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)

1) инициируется прием сообщения с идентификатором 10 от процесса с номером 0

2) инициируется отправка сообщения с идентификатором 10 для процесса с номером A

3) инициируется прием сообщения с идентификатором 10 от процесса с номером A

49) Какие действия связаны с выполнением функции `MPI_Recv(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)`

- 1) принимается массив из 10 элементов целого типа от процесса с номером 0
- 2) посылается 10 значений целой переменной A из памяти процесса 0 процессам с номерами от 1 до 10
- 3) процессы с номерами от 1 до 10 принимают 10 значений целой переменной A, хранящейся в памяти процесса 0

50) Какие действия связаны с выполнением функции `MPI_Recv(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)`

- 1) переменная целого типа A, хранящаяся в памяти процесса с номером 0 принимает значение 10
- 2) в переменную целого типа A, хранящуюся в памяти процесса с номером 0 посылается значение 10
- 3) принимается массив из 10 элементов целого типа от процесса с номером 0

51) Какие действия связаны с выполнением функции `MPI_Recv(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)`

- 1) процессу с номером 10 посылаются элементы массива A с индексами от 0 до 10
- 2) принимается массив из 10 элементов целого типа от процесса с номером 0
- 3) процесс с номером 10 принимает элементы массива A с индексами от 0 до 10

52) Какие действия связаны с выполнением функции `MPI_Recv(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)`

- 1) посылаются данные из памяти процесса с номером 0 в память текущего процесса
- 2) принимаются данные в память текущего процесса из памяти процесса с номером 0
- 3) принимаются данные в память процесса с номером 0 из памяти текущего процесса

53) Какие действия связаны с выполнением функции `MPI_Recv(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)`

- 1) принимаются данные из памяти процесса с номером 0 в память процесса с номером 10
- 2) посылаются данные из памяти процесса с номером 0 в память процесса с номером 10
- 3) принимаются данные в память текущего процесса из памяти процесса с номером 0

54) Какие действия связаны с выполнением функции `MPI_Recv(A, 10, MPI_INT, 0, 10, MPI_COMM_WORLD)`

- 1) посылаются данные из памяти процесса с номером A в память процесса с номером 10
- 2) принимаются данные в память текущего процесса из памяти процесса с номером 0

3) принимаются данные из памяти процесса с номером A в память процесса с номером 10

55) Какие действия связаны с выполнением функции MPI_Bcast(&A, 1, MPI_DOUBLE, 10, MPI_COMM_WORLD)

- 1) Текущий процесс рассылает из своего буфера передачи 10 вещественных чисел всем процессам области связи коммуникатора MPI_COMM_WORLD
- 2) Текущий процесс рассылает значение переменной A из своего буфера передачи всем процессам области связи коммуникатора MPI_COMM_WORLD
- 3) Процесс с номером 10 рассылает значение переменной A из своего буфера передачи всем процессам области связи коммуникатора MPI_COMM_WORLD

56) Какие действия связаны с выполнением функции MPI_Bcast(&A, 1, MPI_DOUBLE, 10, MPI_COMM_WORLD)

- 1) Текущий процесс рассылает из своего буфера передачи 10 вещественных чисел всем процессам области связи коммуникатора MPI_COMM_WORLD
- 2) Текущий процесс выполняет широковещательную рассылку вещественного значения из переменной A
- 3) Процесс с номером 10 выполняет широковещательную рассылку вещественного значения из переменной A

57) Какие действия связаны с выполнением функции MPI_Bcast(&A, 1, MPI_DOUBLE, 10, MPI_COMM_WORLD)

- 1) Процесс с номером 10 рассылает вещественное число всем процессам области связи коммуникатора MPI_COMM_WORLD
- 2) Текущий процесс рассылает вещественное число всем процессам области связи коммуникатора MPI_COMM_WORLD
- 3) Процесс с номером A рассылает вещественное число всем процессам области связи коммуникатора MPI_COMM_WORLD

58) Какие действия связаны с выполнением функции MPI_Bcast(&A, 1, MPI_DOUBLE, 10, MPI_COMM_WORLD)

- 1) Процесс с номером 10 рассылает значение переменной A из своего буфера передачи всем процессам области связи коммуникатора MPI_COMM_WORLD
- 2) Текущий процесс рассылает значение переменной A из своего буфера передачи всем процессам области связи коммуникатора MPI_COMM_WORLD
- 3) Текущий процесс принимает в свой буфер приема значение переменной A от процесса с номером 10

59) Какие действия связаны с выполнением функции MPI_Bcast(&A, 1, MPI_DOUBLE, 10, MPI_COMM_WORLD)

- 1) Процесс с номером 10 выполняет широковещательную рассылку вещественного значения из переменной A
- 2) Текущий процесс выполняет широковещательную рассылку вещественного значения из переменной A
- 3) Текущий процесс принимает массив из 10 вещественных чисел от процесса с номером 1

60) Какие действия связаны с выполнением функции MPI_Bcast(&A, 1, MPI_DOUBLE, 10, MPI_COMM_WORLD)

- 1) Процесс с номером 10 рассылает вещественное число всем процессам области связи коммутатора MPI_COMM_WORLD
- 2) Текущий процесс рассылает вещественное число всем процессам области связи коммутатора MPI_COMM_WORLD
- 3) Процесс с номером 10 принимает по одному вещественному числу от всех процессов области связи коммутатора MPI_COMM_WORLD

60) Сколько параметров имеет функция MPI_Barrier?

- 1) 2
- 2) 1
- 3) 4

61) Какой из вариантов не соответствует описанию функции MPI_Barrier?

- 1) Завершение работы этой функции возможно только всеми процессами одновременно
- 2) Функция выполняет синхронизацию, блокируя работу вызвавшего ее процесса до тех пор, пока все другие процессы группы также не вызовут эту функцию
- 3) Функция блокирует работу процесса до тех пор, пока другой процесс не снимет блокировку с помощью вызова специальной коммуникационной процедуры

62) Какой из вариантов не соответствует описанию функции MPI_Barrier?

- 1) Функция останавливает работу всех процессов, кроме того который ее вызвал
- 2) Функция необходима для завершения всеми процессами некоторого этапа решения задачи, результаты которого будут использоваться на следующем этапе
- 3) Функция гарантирует, что ни один из процессов не приступит раньше времени к выполнению следующего этапа, пока результат работы предыдущего не будет окончательно сформирован

63) Какая коммуникационная функция выполняет синхронизацию, блокируя работу вызвавшего ее процесса до тех пор, пока все другие процессы группы также не вызовут эту функцию?

- 1) MPI_Barrier
- 2) MPI_Testall
- 3) MPI_Testany

64) Какая коммуникационная функция необходима для завершения всеми процессами некоторого этапа решения задачи, результаты которого будут использоваться на следующем этапе?

- 1) MPI_Barrier
- 2) MPI_Test
- 3) MPI_Comm_rank

65) Какая функция производит сборку блоков данных одинаковой длины, посылаемых всеми процессами группы, в один массив в памяти указанного процесса?

- 1) MPI_Gather
- 2) MPI_Scatter
- 3) MPI_Allgather

66) Какая функция производит сборку блоков данных одинаковой длины, посылаемых всеми процессами группы, в один массив в памяти указанного процесса?

- 1) MPI_Gather
- 2) MPI_Scatterv
- 3) MPI_Gatherv

67) Какая функция производит сборку блоков данных одинаковой длины, посылаемых всеми процессами группы, в один массив в памяти указанного процесса?

- 1) MPI_Gather
- 2) MPI_Scatter
- 3) MPI_Allgatherv

68) Какая функция производит сборку блоков данных разной длины, посылаемых всеми процессами группы, в один массив в памяти указанного процесса?

- 1) MPI_Gatherv
- 2) MPI_Scatter
- 3) MPI_Scatterv

69) Какая функция разбивает сообщение из буфера отправки указанного процесса на равные части и посылает i -ю часть в буфер приема процесса с номером i (в том числе и самому себе)?

- 1) MPI_Scatter
- 2) MPI_Gatherv
- 3) MPI_Scatterv

70) Какая функция разбивает сообщение из буфера отправки указанного процесса на части разной длины и посылает i -ю часть в буфер приема процесса с номером i (в том числе и самому себе)?

- 1) MPI_Scatterv
- 2) MPI_Gatherv
- 3) MPI_Scatter

Критерии оценки (в баллах) каждого тестового задания:

- 1 балл выставляется студенту, если он правильно выполняет тестовое задание;
- 0 баллов выставляется студенту, если задание не выполнено или решено неверно.

Задания для практических работ

Задания для практических работ предназначены для оценки уровня сформированности компетенции ПК-3 (индикатор достижения компетенции ПК-3.3).

Практическая работа №1.

Задание: Средствами языка C++ разработать консольную программу для обработки в соответствии с вариантом числовых данных из массива, передаваемых в программу качестве через параметры командной строки.

Варианты:

1. Найти максимальный элемент последовательности, меньше заданного.
2. Найти минимальный элемент последовательности, больше заданного.
3. Найти сумму элементов последовательности, стоящих на четных местах.
4. Найти сумму элементов последовательности, стоящих на нечетных местах.
5. Найти максимальное количество идущих подряд одинаковых элементов массива.

Практическая работа №2.

Задание: Реализовать последовательность действий коммуникационной процедуры MPI_Bcast с помощью пары функций двухточечного обмена MPI_Send, MPI_Recv.

Практическая работа №3.

Задание: С помощью пары функций двухточечного обмена MPI_Send, MPI_Recv реализовать последовательность, аналогичных функции MPI_Reduce.

Практическая работа №4.

Задание: Используя утилиты пакета MPICH развернуть кластер из персональных компьютеров в локальной сети.

Критерии оценки (в баллах)

- 3 балла выставляется студенту, если он правильно выполняет одно задание, демонстрирует его решение у доски и отвечает на дополнительные вопросы;
- 2 балла выставляется студенту, если он правильно выполняет одно задание у доски и отвечает на дополнительные вопросы;
- 1 балл выставляется студенту, если он выполняет задание у доски при помощи наводящих вопросов и/или не отвечает на дополнительные вопросы;
- 0 баллов выставляется студенту, если он не может выполнить задание и ответить на дополнительные вопросы.

Задания для лабораторных работ

Задания для лабораторных работ предназначены для оценки уровня сформированности компетенции ПК-3 (индикатор достижения компетенции ПК-3.3).

Лабораторная работа №1.

Задание: Применяя коммуникационные операции «точка-точка» межпроцессорного взаимодействия найти:

- 1) произведение квадратной матрицы $A(n,n)$ и вектора $x(n)$,
- 2) произведение матрицы $A(n,n)$ на матрицу $B(n,n)$.

Размерность n должна быть больше числа процессов.

Лабораторная работа №2.

Задание типового варианта:

1. Вычислить сумму ряда $\frac{\sin(x)}{x} = 1 - \frac{x^2}{3!} + \frac{x^4}{5!} - \frac{x^6}{7!} + \dots$

Лабораторная работа №3.

Задание: Каждый процесс заполняет свой массив размером n случайными числами. Для решения задачи использовать операции приведения с собственной функцией для решения задачи. Результат - вектор размером n , каждый элемент которого получен по правилу определенной в задаче функции.

Типовой вариант:

1. Найти минимальную степень 2, превосходящую максимальное значение (0, если нет положительных).
2. Найти сумму неравных заданному значению.

Лабораторная работа №4.

Задание типового варианта:

Создать MPI-программу параллельного вычисления системы линейных алгебраических уравнений методом Гаусса.

Критерии оценки (в баллах):

- 5 баллов выставляется студенту, если им выполнены все задания лабораторной работы, даны четкие разъяснения по программному коду, даны развернутые ответы на теоретические вопросы по соответствующему разделу;
- 4 балла выставляется студенту, если им выполнены все задания лабораторной работы, даны четкие разъяснения по программному коду, но даны не полные ответы на теоретические вопросы по соответствующему разделу;
- 3 балла выставляется студенту, если им выполнены все задания лабораторной работы, даны четкие разъяснения по программному коду, не может ответить на теоретические вопросы по соответствующему разделу;
- 2 балла выставляется студенту, если им выполнены все задания лабораторной работы, но не даны четкие разъяснения по программному коду, и/или не может ответить на теоретические вопросы по соответствующему разделу;
- 1 балл выставляется студенту, если им выполнены не все задания лабораторной работы, и/или не разбирается в программном коде, и/или не может ответить на теоретические вопросы по соответствующему разделу;
- 0 баллов выставляется студенту, если лабораторная работа не выполнена.

Задания для контрольной работы

Задания для контрольной работы предназначены для оценки уровня сформированности компетенции ПК-3 (индикатор достижения компетенции ПК-3.3).

Описание контрольной работы:

Контрольная работа представляет собой письменное задание, рассчитанное на выполнение в течение 90 минут, и включает в себя ответы на 3 теоретических и 2 практических вопроса.

Типовой вариант контрольной работы

1. Дайте определение термину параллельного программирования.
2. Опишите этапы передачи данных при широковещательной рассылке данных.
3. Перечислите функции, реализующие двухточечный обмен данными, и опишите их параметры.
4. Приведите пример использования функцию MPI_Bcast.
5. Напишите основной фрагмент программного кода, реализующего параллельное вычисление векторного произведения двух векторов.

Описание методики оценивания:

Контрольная работа проводится с целью обеспечения своевременной обратной связи, для коррекции обучения, активизации самостоятельной работы студентов. Критериями оптимального усвоения знаний при проведении контрольной работы являются объем, системность, осмысленность, прочность и действенность знаний обучающихся.

Результаты контрольной работы оцениваются в соответствии с рейтинг-планом дисциплины.

Критерии оценки (в баллах) каждого задания:

- 21–25 баллов студенту выставляется, если он в полном объеме выполняет все задания;
- 16–20 баллов студенту выставляется, если он в полном объеме выполняет задания 1-4, и/или частично выполняет с недочетами задание 5;
- 11–15 баллов студенту выставляется, если он в полном объеме выполняет задания 1-3, и/или частично выполняет с недочетами задание 4-5;

- 6–10 баллов студенту выставляется, если он в полном объеме выполняет задания 1 и 2, и/или частично выполняет с недочетами задания 3-4;
- 1–5 баллов выставляется студенту, если он в полном объеме выполняет задание 1, и/или частично выполняет с недочетами задания 2-4;
- 0 баллов выставляется студенту, если не выполнено ни одно из 4-х заданий.

Перечень вопросов к зачету

1. Каковы основные характеристики сетей передачи данных?
2. Какие системные платформы могут быть использованы для построения кластеров?
3. Как определяется расписание для распределения вычислений между процессорами?
4. Как определяется время выполнения параллельного алгоритма?
5. Как определить минимально возможное время решения задачи?
6. Какие оценки следует использовать в качестве характеристики времени последовательного решения задачи?
7. Какие зависимости могут быть получены для времени параллельного решения задачи при увеличении или уменьшении числа используемых процессоров?
8. При каком числе процессоров могут быть получены времена выполнения параллельного алгоритма, сопоставимые по порядку с оценками минимально возможного времени решения задачи?
9. Как определяются понятия ускорения и эффективности?
10. В чем состоит противоречивость показателей ускорения и эффективности?
11. Как определяется понятие стоимости вычислений?
12. В чем состоит понятие стоимостно-оптимального алгоритма?
13. В чем заключается проблема распараллеливания последовательного алгоритма суммирования числовых значений?
14. В чем состоит каскадная схема суммирования? С какой целью рассматривается модифицированный вариант данной схемы?
15. В чем состоит различие показателей ускорения и эффективности для рассматриваемых вариантов каскадной схемы суммирования?
16. В чем состоит параллельный алгоритм вычисления всех частных сумм последовательности числовых значений?
17. Как формулируется закон Амдаля? Какой аспект параллельных вычислений позволяет учесть данный закон?
18. Какие предположения используются для обоснования закона Густавсона – Барсиса?
19. Какой алгоритм является масштабируемым? Приведите примеры методов с разным уровнем масштабируемости.
20. Какие основные характеристики используются для оценки топологии сети передачи данных? Приведите значения характеристик для конкретных типов коммуникационных структур (полный граф, линейка, решетка и др.).
21. Какие основные методы применяются при маршрутизации передаваемых данных по сети?
22. В чем состоят основные методы передачи данных? Приведите для этих методов аналитические оценки времени выполнения.
23. Какие операции передачи данных могут быть выделены в качестве основных?

24. В чем состоят алгоритмы выполнения передачи данных от одного процессора всем процессорам сети для топологий кольца, решетки и гиперкуба? Приведите оценки временной трудоемкости для этих алгоритмов.
25. В чем состоят алгоритмы выполнения передачи данных от всех процессоров всем процессорам сети для топологий кольца, решетки гиперкуба? Приведите оценки временной трудоемкости для этих алгоритмов.
26. В чем состоят возможные алгоритмы выполнения операции редукции? Какой из алгоритмов является наилучшим по времени выполнения?
27. В чем состоит алгоритм выполнения операции циклического сдвига?
28. В чем состоит полезность использования логических топологий? Приведите примеры алгоритмов логического представления структуры коммуникационной сети.
29. В чем состоит различие моделей для оценки времени выполнения операций передачи данных в кластерных вычислительных системах? Какая модель является более точной? Какая модель может быть использована для предварительного анализа временной трудоемкости коммуникационных операций?
30. В чем состоят исходные предположения для возможности применения рассмотренной в лекции методики разработки параллельных алгоритмов?
31. Каковы основные этапы проектирования и разработки методов параллельных вычислений?
32. Какие основные требования должны быть обеспечены при разработке параллельных алгоритмов?
33. В чем состоят основные действия на этапе выделения подзадач?
34. Каковы основные действия на этапе определения информационных зависимостей?
35. В чем состоят основные действия на этапе масштабирования имеющегося набора подзадач?
36. В чем состоят основные действия на этапе распределения подзадач по процессорам вычислительной системы?
37. Какой способ выполнения операции обобщенного сбора данных является более эффективным?
38. Какой минимальный набор средств является достаточным для организации параллельных вычислений в системах с распределенной памятью?
39. В чем состоит важность стандартизации средств передачи сообщений?
40. Что следует понимать под параллельной программой?
41. В чем различие понятий процесса и процессора?
42. Какой минимальный набор функций MPI позволяет начать разработку параллельных программ?
43. Как описываются передаваемые сообщения?
44. Как можно организовать прием сообщений от конкретных процессов?
45. Как определить время выполнения MPI-программы?
46. В чем различие парных и коллективных операций передачи данных?
47. Какая функция MPI обеспечивает передачу данных от одного процесса всем процессам?
48. Что понимается под операцией редукции?
49. В каких ситуациях следует применять барьерную синхронизацию?
50. Какие режимы передачи данных поддерживаются в MPI?
51. Как организуется неблокирующий обмен данными в MPI?
52. В чем состоит понятие тупика? Когда функция одновременного выполнения передачи и приема гарантирует отсутствие тупиковых ситуаций?
53. Какие коллективные операции передачи данных предусмотрены в MPI?

54. Что понимается под производным типом данных в MPI?
55. Какие способы конструирования типов имеются в MPI?
56. В каких ситуациях может быть полезна упаковка и распаковка данных?
57. Что понимается в MPI под коммуникатором?
58. Для чего может потребоваться создание новых коммуникаторов?
59. Что понимается в MPI под виртуальной топологией?
60. Какие виды топологий предусмотрены в MPI?
61. Для чего может оказаться полезным использование виртуальных топологий?
62. Какие основные дополнительные возможности предусмотрены в стандарте MPI-2?
63. Назовите основные способы распределения элементов матрицы между процессорами вычислительной системы.
64. В чем состоит постановка задачи умножения матрицы на вектор?
Какова вычислительная сложность последовательного алгоритма умножения матрицы на вектор?

Критерии оценки для зачета

Если по дисциплине по итогам текущего и рубежного контроля студент набрал:

- не менее 60 баллов, то он получает «зачтено»;

- менее 60 баллов, то он добирает баллы, выполняя несделанные лабораторные работы, контрольные работы и тестирование.

3. Методические материалы, определяющие процедуры оценивания результатов обучения по дисциплине (модулю), описание шкал оценивания

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные средства
ПК-3.1. Знать основные понятия и способы построения параллельных алгоритмов и программ для решения задач профессиональной деятельности; принципы построения, организации, архитектуры и структуры вычислительных систем параллельной обработки данных	Знать основные понятия о системах параллельной обработки данных для решения задач профессиональной деятельности	Устный опрос
ПК-3.2. Уметь строить и программно реализовывать алгоритмы параллельного вычисления для	Уметь использовать методы параллельного программирования для получения, хранения, переработки информации при решении профессиональных задач	Тестовые задания

<i>получения, хранения, переработки информации при решении профессиональных задач; проводить обоснованный выбор компьютерных систем параллельной обработки данных</i>		
<i>ПК-3.3. Владеть теоритическими и практическими навыками программирования параллельных алгоритмов для решения профессиональных задач; навыками работы с программными средствами параллельной обработки данных.</i>	<i>Владеть теоритическими и практическими навыками применения параллельных вычислительных систем для решения профессиональных задач</i>	<i>лабораторная работа; практическое задание; контрольная работа.</i>

Рейтинг-план дисциплины

Виды учебной деятельности студентов	Балл за конкретное задание	Число заданий за семестр	Баллы	
			Минимальный	Максимальный
Модуль 1				50
Текущий контроль				25
1) Выполнение практического задания	3	2	0	6
2) Работа на практическом занятии (устный опрос)	1	9	0	9
3) Отчет по лабораторным работам	5	2	0	10
Рубежный контроль				25
1. Письменная контрольная работа	5	5	0	25
Модуль 2				50
Текущий контроль				25
1) Выполнение практического задания	3	2	0	6
2) Работа на практическом занятии (устный опрос)	1	9	0	9
3) Отчет по лабораторным работам	5	2	0	10
Рубежный контроль				25
1. Тестирование	25	1	0	25
Поощрительные баллы				10
1. Участие в научной конференции	5	1	0	5
2. Публикация тезиса, статьи	5	1	0	5
Посещаемость (баллы вычитаются из общей суммы набранных баллов)				
1. Посещение лекционных занятий			0	-6
2. Посещение практических (семинарских, лабораторных занятий)			0	-10
Итоговый контроль				
1. Зачет			0	0

Результаты обучения по дисциплине (модулю) у обучающихся оцениваются по итогам текущего контроля количественной оценкой, выраженной в рейтинговых баллах. Оценке подлежит каждое контрольное мероприятие.

При оценивании сформированности компетенций применяется четырехуровневая

шкала «неудовлетворительно», «удовлетворительно», «хорошо», «отлично».

Максимальный балл по каждому виду оценочного средства определяется в рейтинг-плане и выражает полное (100%) освоение компетенции.

Уровень сформированности компетенции «хорошо» устанавливается в случае, когда объем выполненных заданий соответствующего оценочного средства составляет 80-100%; «удовлетворительно» – выполнено 40-80%; «неудовлетворительно» – выполнено 0-40%

Рейтинговый балл за выполнение части или полного объема заданий соответствующего оценочного средства выставляется по формуле:

Рейтинговый балл = $k \times$ Максимальный балл,

где $k = 0,2$ при уровне освоения «неудовлетворительно», $k = 0,4$ при уровне освоения «удовлетворительно», $k = 0,8$ при уровне освоения «хорошо» и $k = 1$ при уровне освоения «отлично».

Оценка на этапе промежуточной аттестации выставляется согласно Положению о модульно-рейтинговой системе обучения и оценки успеваемости студентов УУНиТ:

На зачете выставляется оценка:

- зачтено - при накоплении от 60 до 110 рейтинговых баллов (включая 10 поощрительных баллов),
- не зачтено - при накоплении от 0 до 59 рейтинговых баллов.

При получении на экзамене оценок «отлично», «хорошо», «удовлетворительно», на зачёте оценки «зачтено» считается, что результаты обучения по дисциплине (модулю) достигнуты и компетенции на этапе изучения дисциплины (модуля) сформированы.