




МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего профессионального образования  
«Дальневосточный федеральный университет»  
(ДФУ)

**ШКОЛА ЕСТЕСТВЕННЫХ НАУК**

«СОГЛАСОВАНО»

Руководитель образовательной программы

  
28.08

И.Л. Артемьева

2015 г.

«УТВЕРЖДАЮ»

Заведующая кафедрой прикладной математики, механики,  
управления и программного обеспечения

  
И.Л. Артемьева

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

Параллельное программирование

**Направление подготовки 02.03.03 Математическое обеспечение и администрирование  
информационных систем**

профиль «Технология программирования»

**Форма подготовки очная**

курс 4 семестр 7

лекции 36 час.

практические занятия 0 час.

лабораторные работы 18 час.

в том числе с использованием МАО лек. 18 час. / пр. 0 час. / лаб. 18 час.

в том числе в электронной форме лек. \_\_\_\_/пр. \_\_\_\_/лаб. 18 час.

всего часов аудиторной нагрузки 54 час.

в том числе с использованием МАО 36 час.

в том числе в электронной форме \_\_18\_\_ час.

самостоятельная работа 90 час.

в том числе на подготовку к экзамену 27 час.

контрольные работы (количество) не предусмотрены

курсовая работа / курсовой проект не предусмотрены

зачет не предусмотрен

экзамен 7 семестр

Рабочая программа составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования, утвержденного приказом Министерства образования и науки РФ от 12 марта 2015 г. № 222

Рабочая программа обсуждена на заседании кафедры прикладной математики, механики, управления и программного обеспечения, протокол № 7 от « 4 » июля 2015 г.

Заведующая кафедрой прикладной математики, механики, управления и программного обеспечения, профессор, д.т.н. Артемьева И.Л.

Составители: доцент Чусов А.А., к.т.н.

**Оборотная сторона титульного листа РПУД**

**I. Рабочая программа пересмотрена на заседании кафедры:**

Протокол от «\_\_\_\_\_» \_\_\_\_\_ 20\_\_ г. № \_\_\_\_\_

Заведующий кафедрой \_\_\_\_\_  
(подпись) (И.О. Фамилия)

**II. Рабочая программа пересмотрена на заседании кафедры:**

Протокол от «\_\_\_\_\_» \_\_\_\_\_ 20\_\_ г. № \_\_\_\_\_

Заведующий кафедрой \_\_\_\_\_  
(подпись) (И.О. Фамилия)

## ABSTRACT

**Bachelor's/Specialist's/Master's degree in 02.03.03 – Software and Administration of Information Systems**

**Study profile/ Specialization/ Master's Program “Title” Programming technology**

**Course title:** Parallel programming

**Variable part of Block 1, 4 credits**

**Instructor:** Velichko A., Golenkov E.

**At the beginning of the course a student should be able to:** solve the standard problems of professional activity using information and bibliographic cultures and applying information and communication technologies and taking into account the main requirements of information security; apply knowledge of the mathematical bases of computer science in professional activity; use the skills of choice, design, implementation, quality assessment and efficiency analysis of software to solve problems from various subject spheres.

**Learning outcomes:** knowledge of the development trends of computers with traditional (nontraditional) architecture and the development trends of functions and architectures of problem-oriented program systems and complexes; ability to use knowledge of the methods of software design and production, construction principles, structure and the methods of work with instrumental tools used for creating software.

**Course description:** multiprocessor computational systems and the bases of parallel data processing, the bases of parallel programming with the help of the library of functions «Message passing interface» (MPI).

### **Main course literature:**

1. Gergel V.P. Vysokoproizvoditel'niye vychisleniya dlya mnogoprotsessornykh mnogoyadernykh sistem [High productive calculations for multiprocessor multicore systems]. Moscow, FIZMATLIT, 2010. 543 p.

<https://lib.dvfu.ru:8443/lib/item?id=chamo:660921&theme=FEFU>

2. Linev A.V., Bogolepov D.K., Bastrakov S.I. Tekhnologii parallel'nogo programmirovaniya dlya protsessorov novykh arkhitektur [Parallel programming technologies for processors with new architecture]. Moscow, Moscow Univeristy Press, 2010. 151 p.

<https://lib.dvfu.ru:8443/lib/item?id=chamo:660909&theme=FEFU>

3. Korniyakov K.V., Kustikova V.D., Meerov I.B. Instrumenty parallel'nogo programmirovaniya v sistemakh s obshchey pamyat'yu [Parallel programming tools in general memory systems]. Moscow, Moscow University Press, 2010. 266 p. <https://lib.dvfu.ru:8443/lib/item?id=chamo:660912&theme=FEFU>

**Form of final control:** exam.

## АННОТАЦИЯ

Дисциплина «Параллельное программирование» предназначена для студентов 4 курса, обучающихся по направлению 02.03.03 «Математическое обеспечение и администрирование информационных систем», профиль «Технология программирования». Трудоемкость дисциплины 4 зачетных единицы (144 часа). Дисциплина основана на знаниях, полученных студентом в курсах информатики, программирования на ЭВМ.

Дисциплина реализуется в 7 семестре. В 7 семестре дисциплина содержит 36 часов лекций, 0 часов практических занятий, 18 часов лабораторных работ, из них 18 часов лекций, 0 часов практических занятий, 18 часов лабораторных работ с использованием методов активного обучения. На самостоятельную работу студентов отводится 90 часов, из них на подготовку к экзамену 27 часов.

Содержание дисциплины охватывает следующий круг вопросов: многопроцессорные вычислительные системы и основы параллельной обработки данных, основы параллельного программирования с использованием операций библиотеки функций «Message passing interface» (MPI).

**Цель** дисциплины - на основе актуальной научной литературы в области высокопроизводительных и распределенных вычислительных технологий изучить и научиться использовать алгоритмы и методы параллельной обработки данных для прикладных задач; изучить соответствующее программное обеспечение, библиотеки, пакеты программ и инструментальные средства; научиться применять современные суперкомпьютеры в проводимых исследованиях.

### **Задачи дисциплины:**

- изучение основ параллельного программирования, в том числе для графических устройств;
- изучение методов организации параллельных вычислений;
- получение навыков создания параллельных программ.

Для успешного изучения дисциплины «Параллельное программирование» у обучающихся должны быть сформированы следующие предварительные компетенции:

- способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности;

- способность применять в профессиональной деятельности знания математических основ информатики;

- готовность использовать навыки выбора, проектирования, реализации, оценки качества и анализа эффективности программного обеспечения для решения задач в различных предметных областях.

В результате изучения данной дисциплины у обучающихся формируются следующие общекультурные, общепрофессиональные, профессиональные компетенции (элементы компетенций).

Код и формулировка компетенции	Этапы формирования компетенции	
ОПК-5 - владение информацией о направлениях развития компьютеров с традиционной (нетрадиционной) архитектурой; о тенденциях развития функций и архитектур проблемно-ориентированных программных систем и комплексов	Знает	характеристики архитектуры вычислительных средств и свойства знаковых систем, лежащих в их основе; ограничения возможностей классических архитектур ЭВМ и основные направления развития информационных технологий на основе высокопараллельных вычислительных сред
	Умеет	оценивать возможности архитектурных решений и соответствие задачам создания и развития информационных технологий в современных и перспективных приложениях
	Владеет	навыками применения знаний для решения конкретных задач в области разработки перспективных информационных технологий
ОПК-8 - способность использовать знания методов проектирования и производства программного продукта, принципов построения, структуры и приемов работы с инструментальными средствами, поддерживающими создание программного обеспечения	Знает	основы математического аппарата описания процессов; возможности перспективных информационных технологий, построенных на базе развитых знаковых систем в составе потоковых формализмов
	Умеет	обоснованно выбирать архитектурные решения для реализации конкретных задач и информационных технологий
	Владеет	навыками разработки программных приложений, ориентированных на реализацию распределенной обработки на сетях общего пользования
ПК-7 владение знаниями о содержании, основных этапах и тенденциях развития программирования, математического	Знает	ограничения возможностей классических архитектур ЭВМ и основные направления развития информационных технологий на основе высокопараллельных вычислительных сред
	Умеет	Использовать методы распараллеливания при решении задач

обеспечения и информационных технологий	Владеет	Методами выбора схем организации параллельных вычислений
---	---------	--

Для формирования вышеуказанных компетенций в рамках дисциплины «Параллельное программирование» применяются неимитационные методы активного/интерактивного обучения: выполнение проектов с использованием компьютерных технологий и специализированного программного обеспечения.

## **I. СТРУКТУРА И СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА**

### **Тема 1. Основы параллельного программирования (2 час.).**

Общая архитектура многопроцессорной системы. Пример многопроцессорных систем. Основные понятия параллельного программирования. Классификация Флинна: одиночный и множественный поток команд и данных. Законы Мура, Амдала и Густавсона-Барсиса. Архитектурные параллельные особенности современных процессоров (многоядерность, суперскалярность, векторизация, прямой доступ к памяти). Векторные наборы инструкций современных процессоров и векторизация последовательных выражений алгоритмов. Методы и средства параллельной обработки информации. Методы и языки параллельного программирования.

### **Тема 2. Парадигма параллельного программирования: синхронизация (4 час.).**

Архитектура многопроцессорной системы с общей памятью. Проблема синхронизации потоков. Проблема условий гонок. Объекты синхронизации: семафоры и мьютексы. Многопоточное программирование и управление потоками в прикладных параллельных программах. Технология программирования OpenMP. Распараллеливание циклов. Группы потоков и распределение нагрузки.

### **Тема 3. Парадигма параллельного программирования: передача сообщений (8 час.).**

Архитектура многопроцессорной системы с распределенной архитектурой памяти. Модель взаимодействия открытых систем. Понятие протокола

передачи данных. Базовые операции передачи данных: Send-Receive. Организация передачи данных на прикладном уровне в распределенных и параллельных программах. Библиотека сокетов и реализация клиент-серверной модели взаимодействия. Стандарт интерфейса передачи сообщений (MPI). Парная передача сообщений. Групповая передача сообщений. Топологии процессов.

#### **Тема 4. Программирование графических ускорителей (4 час.).**

Архитектура графических ускорителей. Технология программирования CUDA. Передача данных между оперативной памятью компьютера и памятью графического ускорителя. Асинхронный вызов исполняемого кода графического ускорителя.

#### **Тема 5. Архитектура многопроцессорных вычислительных систем (2 часа).**

1. Классификация ЭВМ Флинна.
2. МВС с общей (разделяемой) и распределенной (индивидуальной) памятью.
3. Сетевые топологии.

#### **Тема 6. Эмпирические способы формальной оценки эффективности параллельных программ (2 часа).**

1. Количественные методы и подходы к формальной оценке эффективности параллельных программ.
2. Особенности и параметры функции определения числа процессов в области связи.
2. Особенности и параметры функции определения номера процесса.
3. Особенности и параметры функции отсчета времени.

#### **Тема 7. Основы работы в ОС Linux (2 часа).**

1. Особенности операционной системы. Связи с ОС Unix. Ядро системы.
2. Основные команды и параметры системы.
3. Запуск и контроль исполняемых задач и процессов.

#### **Тема 8. Теоретические аспекты программирования операций межточечных взаимодействий в MPI (2 часа).**

1. Теоретические схемы работы функции инициализации MPI\_Init.
2. Теоретические схемы работы функции завершения MPI\_Finalize.



3. Теоретические схемы работы функции определения числа процессов в области связи MPI\_Comm\_size
4. Теоретические схемы работы функции передачи сообщения MPI\_Send.
5. Теоретические схемы работы функции приема сообщения MPI\_Recv.

### **Тема 9. Практические аспекты программирования операций межточечных взаимодействий в MPI (2 часа).**

1. Практические особенности работы функции инициализации MPI\_Init.
2. Практические особенности работы функции завершения MPI программ MPI\_Finalize.
3. Практические особенности работы функции определения числа процессов в области связи MPI\_Comm\_size
4. Практические особенности работы функции передачи сообщения MPI\_Send.
5. Практические особенности работы функции приема сообщения MPI\_Recv.

### **Тема 10. Теоретические аспекты программирования коллективных операций в MPI (2 часа).**

1. Теоретические схемы работы функции рассылки информации (MPI\_Bcast).
2. Теоретические схемы работы функции сборки распределенного по процессам массива (MPI\_Gather, MPI\_Gatherv).
3. Теоретические схемы работы функции сборки распределенного массива с его рассылкой (MPI\_Allgather, MPI\_Allgatherv).
4. Теоретические схемы работы функции разбиения массива и рассылки его фрагментов (MPI\_Scatter, MPI\_Scatterv).
5. Теоретические схемы работы прочих функций коллективных операций (MPI\_Alltoall, MPI\_Alltoallv).

### **Тема 11. Практические аспекты программирования коллективных операций в MPI (2 часа).**

1. Практические особенности работы функции рассылки информации (MPI\_Bcast).
2. Практические особенности работы функции сборки распределенного по процессам массива (MPI\_Gather, MPI\_Gatherv).
3. Практические особенности работы функции сборки распределенного массива с его рассылкой (MPI\_Allgather, MPI\_Allgatherv).
4. Практические особенности работы функции разбиения массива и рассылки его фрагментов (MPI\_Scatter, MPI\_Scatterv).

5. Практические особенности работы прочих функций коллективных операций (MPI\_Alltoall, MPI\_Alltoallv).

### **Тема 12. Теоретические аспекты программирования распределенных операций в MPI (2 часа).**

1. Теоретические схемы работы функции редукции с сохранением результата в адресном пространстве одного процесса (MPI\_Reduce).
2. Теоретические схемы работы функции редукции с сохранением результата в адресном пространстве всех процессов (MPI\_Allreduce).
3. Теоретические схемы работы префиксной операции редукции (MPI\_Scan).
4. Теоретические схемы работы совмещенной операции Reduce/Scatter (MPI\_Reduce\_scatter).

### **Тема 13. Практические аспекты программирования распределенных операций в MPI (2 часа).**

1. Практические особенности работы функции редукции с сохранением результата в адресном пространстве одного процесса (MPI\_Reduce).
2. Практические особенности работы функции редукции с сохранением результата в адресном пространстве всех процессов (MPI\_Allreduce).
3. Практические особенности работы префиксной операции редукции (MPI\_Scan).
4. Практические особенности работы совмещенной операции Reduce/Scatter (MPI\_Reduce\_scatter).

## **II. СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА**

### **Практические занятия (0 часов)**

Не предусмотрены

### **Лабораторные работы (18 часов)**

#### **Лабораторная работа 1. Измерение эффективности параллельных программ (2 часа).**

В процессе выполнения практической работы необходимо познакомиться с основными способами формальной (абсолютной, относительной, сравнительной) оценки эффективности параллельных программ, в том числе, с функциями системы, позволяющими получать количественные параметры для соответствующей оценки эффективности.

## **Лабораторная работа 2. Основы работы в ОС Linux (4 часа).**

В процессе выполнения практической работы необходимо познакомиться с основными элементами синтаксиса ОС Linux, а также подробно раскрыть следующие аспекты: особенности операционной системы, сравнение с ОС Unix, особенности ядра системы, перечень основных команд и параметров системы, способы запуска и контроля исполняемых задач и процессов.

## **Лабораторная работа 3. Программирование операций межточечных взаимодействий в MPI (4 часа).**

В процессе выполнения практической работы необходимо познакомиться с основными функциями, обеспечивающими программирование операций поточечных взаимодействий параллельных программ, в том числе, функции инициализации `MPI_Init`, функции завершения MPI программ `MPI_Finalize`, функции определения числа процессов в области связи `MPI_Comm_size`, функции передачи сообщения `MPI_Send`, функции приема сообщения `MPI_Recv`, функции синхронизации процессов `MPI_Barrier`.

## **Лабораторная работа 4. Коллективные операции в MPI (4 часа).**

Изучение коллективных коммуникационных операций, в том числе:

- рассылка информации от одного процесса всем остальным членам некоторой области связи (`MPI_Bcast`);
- сборка (`gather`) распределенного по процессам массива в один массив с сохранением его в адресном пространстве выделенного (`root`) процесса (`MPI_Gather`, `MPI_Gatherv`);
- сборка (`gather`) распределенного массива в один массив с рассылкой его всем процессам некоторой области связи (`MPI_Allgather`, `MPI_Allgatherv`);
- разбиение массива и рассылка его фрагментов (`scatter`) всем процессам области связи (`MPI_Scatter`, `MPI_Scatterv`);
- совмещенная операция `Scatter/Gather (All-to-All)`, каждый процесс делит данные из своего буфера передачи и разбрасывает фрагменты всем остальным процессам, одновременно собирая фрагменты, посланные другими процессами в свой буфер приема (`MPI_Alltoall`, `MPI_Alltoallv`).

## **Лабораторная работа 5. Распределенные операции в MPI (4 часа).**

Изучение распределенных операций, в том числе операций редукции в следующих вариантах:

- с сохранением результата в адресном пространстве одного процесса (MPI\_Reduce);
- с сохранением результата в адресном пространстве всех процессов (MPI\_Allreduce);
- префиксной операции редукции (MPI\_Scan);
- совмещенной операции Reduce/Scatter (MPI\_Reduce\_scatter).

### **III. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ**

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине «Параллельное программирование» представлено в Приложении 1 и включает в себя:

план-график выполнения самостоятельной работы по дисциплине, в том числе примерные нормы времени на выполнение по каждому заданию;

характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению;

требования к представлению и оформлению результатов самостоятельной работы;

критерии оценки выполнения самостоятельной работы.

### **IV. КОНТРОЛЬ ДОСТИЖЕНИЯ ЦЕЛЕЙ КУРСА**

№ п/п	Контролируемые разделы/темы дисциплины	Коды и этапы формирования компетенций		Оценочные средства	
				наименование	
				текущий контроль	промежуточная аттестация
1	Тема 1. Основы параллельного программирования	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
2	Тема 2 Парадигма параллельного программирования: синхронизация	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
3	Тема 3. Парадигма параллельного программирования: передача сообщений	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
4	Тема 4. Программирование графических ускорителей	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №

5	Тема 5. Архитектура многопроцессорных вычислительных систем	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
6	Тема 6. Эмпирические способы формальной оценки эффективности параллельных программ	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
7	Тема 7. Основы работы в ОС Linux	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
			умеет	ПР6 лабораторная работа	
			владеет		
8	Тема 8. Теоретические аспекты программирования операций межточечных взаимодействий в MPI	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
			умеет	ПР6 лабораторная работа	
			владеет		
9	Тема 9. Практические аспекты программирования операций межточечных взаимодействий в MPI	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
			умеет	ПР6 лабораторная работа	
			владеет		
10	Тема 10. Теоретические аспекты программирования коллективных операций в MPI	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
			умеет	ПР6 лабораторная работа	
			владеет		
11	Тема 11. Практические аспекты программирования коллективных операций в MPI	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
			умеет	ПР6 лабораторная работа	
			владеет		
12	Тема 12. Теоретические аспекты программирования распределенных операций в MPI	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
			умеет	ПР6 лабораторная работа	
			владеет		
13	Тема 13. Практические аспекты программирования распределенных операций в MPI	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
			умеет	ПР6 лабораторная работа	
			владеет		

Типовые контрольные задания, методические материалы, определяющие процедуры оценивания знаний, умений и навыков и (или) опыта

деятельности, а также критерии и показатели, необходимые для оценки знаний, умений, навыков и характеризующие этапы формирования компетенций в процессе освоения образовательной программы, представлены в Приложении 2.

## **V. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

### **Основная литература**

*(электронные и печатные издания)*

1. Высокопроизводительные вычисления для многопроцессорных многоядерных систем: учебник для вузов / В. П. Гергель. М.: Физматлит: Изд-во Московского университета, 2010. – 543 с.  
<https://lib.dvfu.ru:8443/lib/item?id=chamo:660921&theme=FEFU>
2. Технологии параллельного программирования для процессоров новых архитектур: учебник для вузов / А. В. Линева, Д. К. Боголепов, С. И. Бастратов ; под ред. В. П. Гергеля ; Нижегородский государственный университет. Москва: Изд-во Московского университета, 2010. – 151 с.  
<https://lib.dvfu.ru:8443/lib/item?id=chamo:660909&theme=FEFU>
3. Инструменты параллельного программирования в системах с общей памятью: учебник для вузов / К. В. Корняков, В. Д. Кустикова, И. Б. Мееров [и др.]; под ред. В. П. Гергеля; Нижегородский государственный университет. М.: Изд-во Московского университета, 2010. – 266 с.  
<https://lib.dvfu.ru:8443/lib/item?id=chamo:660912&theme=FEFU>
4. Гергель В.П. Теория и практика параллельных вычислений [Электронный ресурс]: учебное пособие/ Гергель В.П.— Электрон. текстовые данные.— М.: БИНОМ. Лаборатория знаний, Интернет-Университет Информационных Технологий (ИНТУИТ), 2007.— 423 с.— Режим доступа: <http://www.iprbookshop.ru/16100> .
5. Богачев К.Ю. Основы параллельного программирования [Электронный ресурс]: учебное пособие/ Богачев К.Ю. - Электрон. текстовые данные. - М.: БИНОМ. Лаборатория знаний, 2013. - 342 с. - Режим доступа: <http://www.iprbookshop.ru/20702>.
6. Федотов И.Е. Модели параллельного программирования [Электронный ресурс]/ Федотов И.Е. - Электрон. текстовые данные. - М.: СОЛОН-ПРЕСС, 2012 - 384 с. - Режим доступа: <http://www.iprbookshop.ru/20877>.

## Дополнительная литература (печатные и электронные издания)

1. Энтони Уильямс Параллельное программирование на C++ в действии. Практика разработки многопоточных программ [Электронный ресурс] : учебное пособие. - М. : ДМК Пресс, 2012. - 672 с. - Режим доступа: [http://e.lanbook.com/books/element.php?pl1\\_id=4813](http://e.lanbook.com/books/element.php?pl1_id=4813).
2. Антонов А.С. Параллельное программирование с использованием технологии MPI: Учебное пособие. - М.: Изд-во МГУ, 2004. URL: <http://parallel.ru/info/parallel/antonov/>.
3. Гергель В.П. Многопроцессорные вычислительные системы и параллельное программирование / НижГу. - Нижний Новгород. URL: <http://www.software.unn.ac.ru/ccam/?doc=14>.
4. Немнюгин С.А. Средства программирования для многопроцессорных вычислительных систем / СПбГУ. - СПб., 2007. URL: <http://www.phys.spbu.ru/library/studentlectures/nemnugin/nemnugin>.
5. Букатов А.А., Дацюк В.Н., Жегуло А.И. Многопроцессорные системы и параллельное программирование / РГГУ. - Ростов, 2003. URL: <http://rsusu1.rnd.runnet.ru/tutor/method/index.html>.

### Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. <http://www.parallel.ru/info/education/> Информационно-аналитический центр по параллельным вычислениям. Раздел «Учебные материалы». URL:
2. <http://window.edu.ru/resource/979/23979> Антонов А.С. Параллельное программирование с использованием технологии MPI. Учебное пособие. - М.: Изд-во МГУ, 2004. - 71 с.
3. [http://www.hpcc.unn.ru/files/HTML\\_Version/index.html](http://www.hpcc.unn.ru/files/HTML_Version/index.html) Гергель В.П., Стронгин Р.Г Основы параллельных вычислений для многопроцессорных вычислительных систем. Учебное пособие. Издание 2-е, дополненное. - Н. Новгород: Изд-во ННГУ, 2003.
4. <http://window.edu.ru/resource/974/67974> Демьянович Ю.К., Евдокимова Т.О. Теория распараллеливания и синхронизация: курс лекций. - СПб.: Изд-во С.-Пб. ун-та, 2004. - 110 с.
5. <http://window.edu.ru/resource/184/58184> Баканов В.М. Параллельные вычисления: Учебное пособие. - М.: МГУПИ, 2006. - 124 с.

6. <http://window.edu.ru/resource/897/71897> Высокопроизводительные вычисления на кластерах: Учебное пособие / Под ред. А.В. Старченко. - Томск: Изд-во Том. ун-та, 2008. - 198 с.
7. <http://www.studentlibrary.ru/book/ISBN9785211059337.html> Вычислительная математика и структура алгоритмов: 10 лекций о том, почему трудно решать задачи на вычислительных системах параллельной архитектуры и что надо знать дополнительно, чтобы успешно преодолевать эти трудности [Электронный ресурс]: учебник / Воеводин В.В. - 2-е издание, стереотипное. - М.: Издательство Московского государственного университета, 2010. - (Серия "Суперкомпьютерное образование"). -

### **Перечень информационных технологий и программного обеспечения**

Занятия проводятся с использованием проектора и мультимедийного комплекса для демонстрации мультимедийного контента внутренней системы портала ДВФУ. Лабораторные занятия проводятся в специализированном компьютерном классе.

При осуществлении образовательного процесса по дисциплине используется свободно распространяемое программное обеспечение: компилятор языка Си, OpenMPI, Octave с модулем «MPI».

## **VI. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ**

### **Рекомендации по планированию и организации времени, отведенного на изучение дисциплины, описание последовательности действий обучающихся**

Освоение дисциплины следует начинать с изучения рабочей учебной программы, которая содержит основные требования к знаниям, умениям и навыкам. Обязательно следует учитывать рекомендации преподавателя, данные в ходе установочных занятий. Затем – приступать к изучению отдельных разделов и тем в порядке, предусмотренном программой.

Получив представление об основном содержании раздела, темы, необходимо изучить материал с помощью рекомендуемой основной литературы. Целесообразно составить краткий конспект или схему, отображающую смысл и связи основных понятий данного раздела и включенных в него тем. Обязательно следует записывать возникшие вопросы, на которые не удалось ответить самостоятельно.

Подготовку к началу обучения включает несколько необходимых пунктов:



1) Необходимо создать для себя рациональный и эмоционально достаточный уровень мотивации к последовательному и планомерному изучению дисциплины.

2) Необходимо изучить список рекомендованной основной и дополнительной литературы и убедиться в её наличии у себя дома или в библиотеке в бумажном или электронном виде.

3) Необходимо иметь «под рукой» специальные и универсальные словари, справочники и энциклопедии, для того, чтобы постоянно уточнять значения используемых терминов и понятий. Пользование словарями и справочниками необходимо сделать привычкой. Опыт показывает, что неудовлетворительное усвоение предмета зачастую коренится в неточном, смутном или неправильном понимании и употреблении понятийного аппарата учебной дисциплины.

4) Желательно в самом начале периода обучения возможно тщательнее спланировать время, отводимое на работу с источниками и литературой по дисциплине, представить этот план в наглядной форме (график работы с датами) и в дальнейшем его придерживаться, не допуская срывов графика индивидуальной работы и «аврала» в предсессионный период. Пренебрежение этим пунктом приводит к переутомлению и резкому снижению качества усвоения учебного материала.

### **Рекомендации по работе с литературой**

1) Всю учебную литературу желательно изучать «под конспект». Чтение литературы, не сопровождаемое конспектированием, даже пусть самым кратким – бесполезная работа. Цель написания конспекта по дисциплине – сформировать навыки по поиску, отбору, анализу и формулированию учебного материала.

2) Написание конспекта должно быть творческим – нужно не переписывать текст из источников, но пытаться кратко излагать своими словами содержание ответа, при этом максимально структурируя конспект, используя символы и условные обозначения. Копирование и «заучивание» неосмысленного текста трудоемко и по большому счету не имеет большой познавательной и практической ценности.

3) При написании конспекта используется тетрадь, поля в которой обязательны. Страницы нумеруются, каждый новый вопрос начинается с нового листа, для каждого экзаменационного вопроса отводится 1-2 страницы конспекта. На полях размещается вся вспомогательная информация – ссылки, вопросы, условные обозначения и т.д.

4) В итоге данной работы «идеальным» является полный конспект по программе дисциплины, с выделенными определениями, узловыми пунктами, примерами, неясными моментами, проставленными на полях вопросами.

5) При работе над конспектом обязательно выявляются и отмечаются трудные для самостоятельного изучения вопросы, с которыми уместно обратиться к преподавателю при посещении установочных лекций и консультаций, либо в индивидуальном порядке.

6) При чтении учебной и научной литературы всегда следить за точным и полным пониманием значения терминов и содержания понятий, используемых в тексте. Всегда следует уточнять значения по словарям или энциклопедиям, при необходимости записывать.

7) При написании учебного конспекта обязательно указывать все прорабатываемые источники, автор, название, дата и место издания, с указанием использованных страниц.

#### **Подготовка к промежуточной аттестации по дисциплине: экзамену (зачету)**

К аттестации допускаются студенты, которые систематически в течение всего семестра посещали и работали на занятиях и показали уверенные знания в ходе выполнении практических заданий и лабораторных работ.

Непосредственная подготовка к аттестации осуществляется по вопросам, представленным в рабочей учебной программе. Тщательно изучите формулировку каждого вопроса, вникните в его суть, составьте план ответа. Обычно план включает в себя:

- определение сущности рассматриваемого вопроса, основных положений, утверждений, определение необходимости их доказательства;
- запись обозначений, формул, необходимых для полного раскрытия вопроса;
- графический материал (таблицы, рисунки, графики), необходимые для раскрытия сущности вопроса;
- роль и значение рассматриваемого материала для практической деятельности, примеры использования в практической деятельности.

## **VII. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

Для осуществления образовательного процесса по дисциплине необходима лекционная аудитория мультимедийного типа (мультимедийный проектор, настенный экран, документ-камера) и компьютерный класс с персональными компьютерами с доступом в сеть «Интернет».



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего профессионального образования  
«Дальневосточный федеральный университет»  
(ДВФУ)

---

ШКОЛА ЕСТЕСТВЕННЫХ НАУК

**УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ  
РАБОТЫ ОБУЧАЮЩИХСЯ**

**по дисциплине «Параллельное программирование»**

**Направление подготовки 02.03.03 Математическое обеспечение и  
администрирование информационных систем**

**Профиль «Технология программирования»**

**Форма подготовки очная**

**Владивосток  
2015**

## План-график выполнения самостоятельной работы по дисциплине

№ п/п	Дата/сроки выполнения	Вид самостоятельной работы	Примерные нормы времени на выполнение	Форма контроля
1	1-4 неделя	Повторение теоретического и практического материала дисциплины, заслушиваемого и конспектируемого в ходе аудиторных занятий; изучение основной и дополнительной литературы, указанной в рабочей учебной программе дисциплины, самоконтроль ответов на основные проблемные вопросы по темам занятий	20 часов	Собеседование УО-1
2	5-6 неделя	Самостоятельный разбор заданий и задач, решаемых на практических занятиях; самостоятельный повтор действий, осуществляемых в ходе выполнения лабораторных работ, в том числе при работе со специальным программным обеспечением	10 часов	Проект ПР-9
3	7-10 неделя	Повторение теоретического и практического материала дисциплины, заслушиваемого и конспектируемого в ходе аудиторных занятий; изучение основной и дополнительной литературы, указанной в рабочей учебной программе дисциплины, самоконтроль ответов на основные проблемные вопросы по темам занятий	20 часов	Собеседование УО-1
4	11-12 неделя	Самостоятельный разбор заданий и задач, решаемых на практических занятиях; самостоятельный повтор действий, осуществляемых в ходе выполнения лабораторных работ,	10 часов	Проект ПР-9

		в том числе при работе со специальным программным обеспечением		
5	16 неделя	Повторение теоретического и практического материала дисциплины, заслушиваемого и конспектируемого в ходе аудиторных занятий; изучение основной и дополнительной литературы, указанной в рабочей учебной программе дисциплины, самоконтроль ответов на основные проблемные вопросы по темам занятий	20 часов	Собеседование УО-1
6	17-18 неделя	Самостоятельный разбор заданий и задач, решаемых на практических занятиях; самостоятельный повтор действий, осуществляемых в ходе выполнения лабораторных работ, в том числе при работе со специальным программным обеспечением	10 часов	Проект ПР-9

### **Характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению**

1. Разобрать первую параллельную программу MPI для определения каждым процессом количества процессов и своего ранга. Дополнить программу процедурой записи каждым процессом сообщения о своем ранге в отдельный файл (воспользоваться функциями `fopen`, `fprintf`, `fclose`). Дополнить программу измерением времени выполнения процесса.

Текст программы:

```
#include <stdio.h>
#include <stdlib.h>
#include <mpi.h>
#include <time.h>
```

```

int main(int argc, char **argv)
{
int size, rank;
FILE *f;
char str[10];
double t;
struct timeval *time0, *time1;
time0=(struct timeval*)malloc(sizeof(struct timeval));
time1=(struct timeval*)malloc(sizeof(struct timeval));
MPI_Status status;
MPI_Init(&argc,&argv);
MPI_Comm_size(MPI_COMM_WORLD,&size);
MPI_Comm_rank(MPI_COMM_WORLD,&rank);
gettimeofday(time0,NULL);
t=MPI_Wtime();
sprintf(str,"output%d",rank);
f=fopen(str,"w");
fprintf(f,"size=%d\n",size);
fprintf(f,"rank=%d\n",rank);
fclose(f);
t=MPI_Wtime()-t;
printf("t=%e ",t);
gettimeofday(time1,NULL);
printf("time=%f\n",time1->tv_sec-time0->tv_sec/1.+time1->tv_usec/1.e6-time0->tv_usec/1.e6);
MPI_Finalize();
}

```

2. Написать программу, реализующую межточечные взаимодействия между процессорами: "один каждому" (вычислительная модель "клиент-сервер"), "каждый каждому" (одноранговая вычислительная модель) с помощью функций MPI\_Send, MPI\_Recv, MPI\_Sendrecv).

Текст программы:

```

#include <stdio.h>
#include <stdlib.h>
#include <mpi.h>
#include <string.h>

```

```

int main(int argc, char **argv)
{
int size, rank;
MPI_Status status;
MPI_Init(&argc, &argv);
MPI_Comm_size(MPI_COMM_WORLD, &size);
MPI_Comm_rank(MPI_COMM_WORLD, &rank);
if(rank==0)
{
int *i1;
char *c1;
int j, r;
i1=(int*)malloc(10*sizeof(int));
c1=(char*)malloc(28*sizeof(char));
c1="Hello, thanks for waiting!!!";
for(j=0;j<=9;j++)
{
i1[j]=j;
}
for(j=1;j<=5;j++)
{
MPI_Send(i1, 10, MPI_INT, j, 0, MPI_COMM_WORLD);

MPI_Sendrecv(c1, 28, MPI_CHAR, j, 1, &r, 1, MPI_INT, j, 0, MPI_CO
MM_WORLD, &status);
printf("Ранг=%d\n", r);
}
}
if(rank!=0)
{
FILE *f;
int *i1, i;
char *name;
char *c1;
int r;
r=rank;
c1=(char*)malloc(28*sizeof(char));
name=(char*)malloc(5*sizeof(char));

```



```

    i1=(int*)malloc(10*sizeof(int));

MPI_Sendrecv(&r,1,MPI_INT,0,0,c1,28,MPI_CHAR,0,1,MPI_COMM_WORLD,&status);
    sprintf(name,"proc%d",rank);
    f=fopen(name,"w");
    MPI_Recv(i1,10,MPI_INT,0,0,MPI_COMM_WORLD,&status);
    fprintf(f,"Номер процесса=%d\nЗначение c1=%s\n",rank,c1);
    fprintf(f,"Номер процесса=%d\nЗначение i1=",rank);
    for(i=0;i<=9;i++){
        fprintf(f,"-",i1[i]);
    }
    fclose(f);
}
MPI_Finalize();
}

```

3. Написать программу, реализующую коллективные функции MPI (MPI\_Gather, MPI\_Scatter).

```

#include <stdio.h>
#include <stdlib.h>
#include <mpi.h>
#include <string.h>

int main(int argc, char **argv)
{
    int size, rank;
    MPI_Status status;
    MPI_Init(&argc,&argv);
    MPI_Comm_size(MPI_COMM_WORLD,&size);
    MPI_Comm_rank(MPI_COMM_WORLD,&rank);

    int i;
    int sendarray[10];
    int root;
    int *rbuf;
    int size_t=strlen("This is the spam message!!!");;

```

```

char *str_recive, str[size_t*size];
root=0;
str_recive=(char*)malloc(size_t*sizeof(char));
if(rank==root)
{
    rbuf=(int*)malloc(10*size*sizeof(int));
    strcpy(str,"This is the spam message!!!");
    if(size>=2)
        for(i=1;i<size;i++)
            strcat(str,"This is the spam message!!!");
}
for(i=0;i<=9;i++)
    sendarray[i]=i;

MPI_Scatter(str,size_t,MPI_CHAR,str_recive,size_t,MPI_C
HAR,root,MPI_COMM_WORLD);

MPI_Gather(sendarray,10,MPI_INT,rbuf,10,MPI_INT,root,MP
I_COMM_WORLD);
    printf("Процесс          %d          получил
сообщение:%s\n",rank,str_recive);
    if(rank==root)
    {
        int j;
        printf("\nArray=");
        for(j=0;j<=(10*size-1);j++)
        {
            printf("-",rbuf[j]);
        }
        printf("\n");
    }
MPI_Finalize();
}

```

### **Требования к представлению и оформлению результатов самостоятельной работы**

Самостоятельная работа включает в себя повторение теоретического и практического материала дисциплины, заслушиваемого и конспектируемого в ходе аудиторных занятий; изучение основной и дополнительной

литературы, указанной в рабочей учебной программе дисциплины, самоконтроль ответов на основные проблемные вопросы по темам занятий; самостоятельный разбор заданий и задач, решаемых на практических занятиях; самостоятельный повтор действий, осуществляемых в ходе выполнения лабораторных работ, в том числе при работе со специальным программным обеспечением.

Результаты самостоятельной работы представляются и оформляются в виде ответов на основные положения теоретического и практического материала дисциплины по темам; письменного разбора процесса решения практических заданий и задач; собственных действий, осуществляемых в ходе выполнения лабораторных работ.

В случае подготовки слайдов для защиты проекта, они должны быть контрастными (рекомендуется черный цвет шрифта на светлом фоне), кегль текста слайдов – не менее 22pt, заголовков – 32pt. Основная цель использования слайдов - служить вспомогательным инструментом к подготовленному выступлению, цитирование больших фрагментов текста на слайдах не допускается. Приветствуется использование рисунков, графиков, таблиц, интерактивного материала, однако, следует предусмотреть выбор цвета и толщину линий.

Слайды должны содержать титульный лист, цели и задачи (не более 2-х слайдов с обзором актуальности, новизны, теоретической и практической значимости работы), основные публикации с их кратким обзором (1-2 слайда), формальную постановку задачи и формулировку моделей (1-2 слайда), краткое тезисное (!) изложение ключевых положений работы (разумное количество слайдов с учетом общего времени выступления), заключение (с изложением результатов работы, подведением выводов, обсуждением практического использования работы, возможностей проведения дальнейших исследований и разработок в данной области).

Как правило, 12-15 слайдов оказывается достаточным для полного представления работы.

### **Критерии оценки выполнения самостоятельной работы**

Общие критерии оценки выполнения самостоятельной работы – правильность ответов на вопросы по темам теоретической части дисциплины, верность получаемых ответов в ходе решения практических заданий и задач, достижение правильного результата при осуществлении собственных действий по лабораторным работам.

Оценивание знаний в форме собеседования проводится по критериям:

- логичность изложения, знание и понимание основных аспектов и дискуссионных проблем по теме;

- владение методами и приемами анализа теоретических и/или практических аспектов по теме.

Оценивание знаний в форме проекта проводится по критериям:

- завершенность и полнота выполненных заданий в рамках проекта;

- владение методами и приемами решения конкретных задач и самостоятельность использования специализированного программного обеспечения;

- качество оформления письменного отчета в соответствии с правилами и стандартами оформления.



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего профессионального образования  
«Дальневосточный федеральный университет»  
(ДВФУ)

---

ШКОЛА ЕСТЕСТВЕННЫХ НАУК

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**  
по дисциплине «Параллельное программирование»  
Направление подготовки 02.03.03 Математическое обеспечение и  
администрирование информационных систем  
Профиль «Технология программирования»  
Форма подготовки очная

Владивосток  
2015

**Паспорт  
фонда оценочных средств  
по дисциплине «Параллельное программирование»**

<b>Код и формулировка компетенции</b>	<b>Этапы формирования компетенции</b>	
ОПК-5 - владение информацией о направлениях развития компьютеров с традиционной (нетрадиционной) архитектурой; о тенденциях развития функций и архитектур проблемно-ориентированных программных систем и комплексов	Знает	характеристики архитектуры вычислительных средств и свойства знаковых систем, лежащих в их основе;
	Умеет	оценивать возможности архитектурных решений и соответствие задачам создания и развития информационных технологий в современных и перспективных приложениях
	Владеет	навыками применения знаний для решения конкретных задач в области разработки перспективных информационных технологий
ОПК-8 - способность использовать знания методов проектирования и производства программного продукта, принципов построения, структуры и приемов работы с инструментальными средствами, поддерживающими создание программного обеспечения	Знает	основы математического аппарата описания процессов; возможности перспективных информационных технологий, построенных на базе развитых знаковых систем в составе потоковых формализмов
	Умеет	обоснованно выбирать архитектурные решения для реализации конкретных задач и информационных технологий
	Владеет	навыками разработки программных приложений, ориентированных на реализацию распределенной обработки на сетях общего пользования
ПК-7 владение знаниями о содержании, основных этапах и тенденциях развития программирования, математического обеспечения и информационных технологий	Знает	ограничения возможностей классических архитектур ЭВМ и основные направления развития информационных технологий на основе высокопараллельных вычислительных сред
	Умеет	Использовать методы распараллеливания при решении задач
	Владеет	Методами выбора схем организации параллельных вычислений

№ п/п	Контролируемые разделы/темы дисциплины	Коды и этапы формирования компетенций		Оценочные средства	
				текущий контроль	промежуточная аттестация
1	Тема 1. Основы параллельного программирования	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
2	Тема 2 Парадигма параллельного программирования: синхронизация	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
3	Тема 3. Парадигма параллельного программирования: передача сообщений	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
4	Тема 4. Программирование графических ускорителей	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
5	Тема 5. Архитектура многопроцессорных вычислительных систем	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
6	Тема 6. Эмпирические способы формальной оценки эффективности параллельных программ	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
7	Тема 7. Основы работы в ОС Linux	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
			умеет	ПР6 лабораторная работа	
			владеет		
8	Тема 8. Теоретические аспекты программирования операций межточечных взаимодействий в MPI	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
			умеет	ПР6 лабораторная работа	
			владеет		
9	Тема 9. Практические аспекты программирования операций межточечных взаимодействий в MPI	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
			умеет	ПР6 лабораторная работа	
			владеет		
10	Тема 10. Теоретические аспекты программирования коллективных операций в MPI	ОПК5, ОПК8	знает	Собеседование (УО-1)	Экзамен, вопросы №
			умеет	ПР6 лабораторная работа	
			владеет		
11	Тема 11. Практические	ОПК5,	знает	Собеседование	Экзамен, вопросы

	аспекты программирования коллективных операций в MPI	ОПК8	умеет владеет	е (УО-1) ПР6 лабораторная работа	№
12	Тема 12. Теоретические аспекты программирования распределенных операций в MPI	ОПК5, ОПК8	знает умеет владеет	Собеседование (УО-1) ПР6 лабораторная работа	Экзамен, вопросы №
13	Тема 13. Практические аспекты программирования распределенных операций в MPI	ОПК5, ОПК8	знает умеет владеет	Собеседование (УО-1) ПР6 лабораторная работа	Экзамен, вопросы №

### Шкала оценивания уровня сформированности компетенций

Код и формулировка компетенции	Этапы формирования компетенции		критерии	показатели
ОПК-5 - владение информацией о направлениях развития компьютеров с традиционной (нетрадиционной) архитектурой; о тенденциях развития функций и архитектур проблемно-ориентированных программных систем и комплексов	знает (пороговый уровень)	характеристики архитектуры вычислительных средств и свойства знаковых систем, лежащих в их основе; ограничения возможностей классических архитектур ЭВМ и основные направления развития информационных технологий на основе высокопараллельных вычислительных сред	знание вопросов, связанных с архитектурой и возможностями высокопараллельных вычислительных комплексов	способность дать ответы на вопросы по характеристикам архитектуры и ограничениям
	умеет (продвинутой)	оценивать возможности архитектурных решений и соответствие	умение оценить соответствие архитектуры для поставленных задач	способность обосновать возможность использования параллельных



		задачам создания и развития информационных технологий в современных и перспективных приложениях		архитектур для поставленных задач
	владеет (высокий)	навыками применения знаний для решения конкретных задач в области разработки перспективных информационных технологий	владение методами использования архитектур при решении прикладных задач	способность предложить методы использования параллельных архитектур для поставленных задач
ОПК-8 - способность использовать знания методов проектирования и производства программного продукта, принципов построения, структуры и приемов работы с инструментальными средствами, поддерживающими создание программного обеспечения	знает (пороговый уровень)	основы математического аппарата описания процессов; возможности перспективных информационных технологий, построенных на базе развитых знаковых систем в составе потоковых формализмов	знание математического аппарата и операций	способность дать ответы на вопросы о математическом аппарате описания процессов
	умеет (продвинутой)	обоснованно выбирать архитектурные решения для реализации конкретных задач и информационных технологий	умение выбирать архитектурные решения	способность дать обоснование выбора
	владеет (высокий)	навыками разработки программных приложений, ориентированных на реализацию распределенной обработки на сетях общего пользования	владение методами создания программного обеспечения	способность создать программу

ПК-7 владение знаниями о содержании, основных этапах и тенденциях развития программирования, математического обеспечения и информационных технологий	знает (пороговый уровень)	ограничения возможностей классических архитектур ЭВМ и основные направления развития информационных технологий на основе высокопараллельных вычислительных сред	Знание основных методы создания параллельных программ	Способность ответить на вопросы
	умеет (продвинутой)	Использовать методы распараллеливания при решении задач	Умение использовать методы при создании приложений	Способность создать параллельную программу
	владеет (высокий)	Методами выбора схем организации параллельных вычислений	Владение методами обоснования выбора	Способность пояснить выбор

### **Методические рекомендации, определяющие процедуры оценивания результатов освоения дисциплины**

**Текущая аттестация студентов.** Текущая аттестация студентов по дисциплине «Параллельное программирование» проводится в соответствии с локальными нормативными актами ДВФУ и является обязательной.

Текущая аттестация по дисциплине «Параллельное программирование» проводится в форме контрольных мероприятий (собеседования) по оцениванию фактических результатов обучения студентов и осуществляется ведущим преподавателем.

Объектами оценивания выступают:

- учебная дисциплина (активность на занятиях, своевременность выполнения различных видов заданий, посещаемость всех видов занятий по аттестуемой дисциплине);
- степень усвоения теоретических знаний;
- уровень овладения практическими умениями и навыками по всем видам учебной работы;
- результаты самостоятельной работы.

**Промежуточная аттестация студентов.** Промежуточная аттестация студентов по дисциплине «Параллельное программирование» проводится в соответствии с локальными нормативными актами ДВФУ и является обязательной.

По дисциплине «Параллельное программирование» предусмотрен экзамен (устный опрос в форме ответов на вопросы экзаменационных билетов).

### **Оценочные средства для промежуточной аттестации**

#### **Вопросы для подготовки к экзамену по дисциплине «Параллельное программирование»**

1. МВС с общей (разделяемой) памятью.
2. МВС с распределенной (индивидуальной) памятью.
3. Сетевые топологии
4. Эффективность параллельных программ.
5. Законы Амдала.
6. Модели и средства программирования систем с общей памятью.
7. Модели и средства программирования систем с распределенной памятью.
8. Основы работы с инструментальными средствами операционной системы Linux.
9. Операции межпроцессорного обмена в библиотеке MPI.
10. Коллективные операции в библиотеке MPI.
11. Распределенные операции в библиотеке MPI.

#### **Критерии выставления оценки студенту на экзамене по дисциплине «Параллельное программирование»:**

<b>Баллы</b>	<b>Оценка зачета/ экзамена</b>	<b>Требования к сформированным компетенциям</b>
--------------	--------------------------------	---

86-100	«зачтено»/ «отлично»	Оценка «отлично» выставляется студенту, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.
<b>Баллы</b>	<b>Оценка зачета/ экзамена (стандартная)</b>	<b>Требования к сформированным компетенциям</b>
76-85	«зачтено»/ «хорошо»	Оценка «хорошо» выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения.
61-75	«зачтено»/ «удовлетворительно»	Оценка «удовлетворительно» выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ.
0-60	«не зачтено»/ «неудовлетворительно»	Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.

# Оценочные средства для текущей аттестации

## Вопросы для собеседования

по дисциплине «Параллельное программирование»

### Раздел 1. Архитектура многопроцессорных вычислительных систем.

1. Классификация ЭВМ Флинна.
2. МВС с общей (разделяемой) и распределенной (индивидуальной) памятью.
3. Сетевые топологии.

### Раздел 2. Эмпирические способы формальной оценки эффективности параллельных программ.

1. Количественные методы и подходы к формальной оценке эффективности параллельных программ.
2. Особенности и параметры функции определения числа процессов в области связи.
2. Особенности и параметры функции определения номера процесса.
3. Особенности и параметры функции отсчета времени.

### Раздел 3. Основы работы в ОС Linux.

1. Особенности операционной системы. Связи с ОС Unix. Ядро системы.
2. Основные команды и параметры системы.
3. Запуск и контроль исполняемых задач и процессов.

### Раздел 4. Теоретические аспекты программирования операций межточечных взаимодействий в MPI.

1. Теоретические схемы работы функции инициализации MPI\_Init.
2. Теоретические схемы работы функции завершения MPI программ MPI\_Finalize.
3. Теоретические схемы работы функции определения числа процессов в области связи MPI\_Comm\_size
4. Теоретические схемы работы функции передачи сообщения MPI\_Send.
5. Теоретические схемы работы функции приема сообщения MPI\_Recv.

### Раздел 5. Практические аспекты программирования операций межточечных взаимодействий в MPI.

1. Практические особенности работы функции инициализации MPI\_Init.
2. Практические особенности работы функции завершения MPI программ MPI\_Finalize.
3. Практические особенности работы функции определения числа процессов в области связи MPI\_Comm\_size
4. Практические особенности работы функции передачи сообщения MPI\_Send.
5. Практические особенности работы функции приема сообщения MPI\_Recv.

## **Раздел 6. Теоретические аспекты программирования коллективных операций в MPI.**

1. Теоретические схемы работы функции рассылки информации (MPI\_Bcast).
2. Теоретические схемы работы функции сборки распределенного по процессам массива (MPI\_Gather, MPI\_Gatherv).
3. Теоретические схемы работы функции сборки распределенного массива с его рассылкой (MPI\_Allgather, MPI\_Allgatherv).
4. Теоретические схемы работы функции разбиения массива и рассылки его фрагментов (MPI\_Scatter, MPI\_Scatterv).
5. Теоретические схемы работы прочих функций коллективных операций (MPI\_Alltoall, MPI\_Alltoallv).

## **Раздел 7. Практические аспекты программирования коллективных операций в MPI.**

1. Практические особенности работы функции рассылки информации (MPI\_Bcast).
2. Практические особенности работы функции сборки распределенного по процессам массива (MPI\_Gather, MPI\_Gatherv).
3. Практические особенности работы функции сборки распределенного массива с его рассылкой (MPI\_Allgather, MPI\_Allgatherv).
4. Практические особенности работы функции разбиения массива и рассылки его фрагментов (MPI\_Scatter, MPI\_Scatterv).
5. Практические особенности работы прочих функций коллективных операций (MPI\_Alltoall, MPI\_Alltoallv).

## **Раздел 8. Теоретические аспекты программирования распределенных операций в MPI.**

1. Теоретические схемы работы функции редукции с сохранением результата в адресном пространстве одного процесса (MPI\_Reduce).
2. Теоретические схемы работы функции редукции с сохранением результата в адресном пространстве всех процессов (MPI\_Allreduce).
3. Теоретические схемы работы префиксной операции редукции (MPI\_Scan).
4. Теоретические схемы работы совмещенной операции Reduce/Scatter (MPI\_Reduce\_scatter).

## **Раздел 9. Практические аспекты программирования распределенных операций в MPI.**

1. Практические особенности работы функции редукции с сохранением результата в адресном пространстве одного процесса (MPI\_Reduce).
2. Практические особенности работы функции редукции с сохранением результата в адресном пространстве всех процессов (MPI\_Allreduce).
3. Практические особенности работы префиксной операции редукции (MPI\_Scan).
4. Практические особенности работы совмещенной операции Reduce/Scatter (MPI\_Reduce\_scatter).

### **Критерии оценки устного ответа**

✓ 100-86 баллов - если ответ показывает глубокое и систематическое знание всего программного материала и структуры конкретного вопроса, а также основного содержания и новаций лекционного курса по сравнению с учебной литературой. Студент демонстрирует отчетливое и свободное владение концептуально-понятийным аппаратом, научным языком и терминологией соответствующей научной области. Знание основной литературы и знакомство с дополнительно рекомендованной литературой. Логически корректное и убедительное изложение ответа.

✓ 85-76 - баллов - знание узловых проблем программы и основного содержания лекционного курса; умение пользоваться концептуально-понятийным аппаратом в процессе анализа основных проблем в рамках данной темы; знание важнейших работ из списка рекомендованной литературы. В целом логически корректное, но не всегда точное и аргументированное изложение ответа.

✓ 75-61 - балл – фрагментарные, поверхностные знания важнейших разделов программы и содержания лекционного курса; затруднения с использованием научно-понятийного аппарата и терминологии учебной

дисциплины; неполное знакомство с рекомендованной литературой; частичные затруднения с выполнением предусмотренных программой заданий; стремление логически определенно и последовательно изложить ответ.

✓ 60-50 баллов – незнание, либо отрывочное представление о данной проблеме в рамках учебно-программного материала; неумение использовать понятийный аппарат; отсутствие логической связи в ответе.