

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Таныгин Максим Олегович

Должность: и.о. декана факультета фундаментальной и прикладной информатики

Дата подписания: 01.06.2022 10:37:39

Уникальный программный ключ:

65ab2aa0d384efe8480e6a4c688eddbc475e411a

## **Аннотация к рабочей программе дисциплины**

### **«Программирование распределенных и массивно-параллельных систем»**

#### **Цель преподавания дисциплины**

Формирование у студентов систематизированных знаний основных принципов организации параллельных вычислений с использованием различных современных аппаратных платформ, инструментальных средств и технологий разработки ПО.

#### **Задачи изучения дисциплины**

- ознакомление студентов с теоретическими основами программирования параллельных вычислительных архитектур
- ознакомление студентов с распространенными стандартами и протоколами, используемыми в параллельном программировании.

#### **Компетенции, формируемые в результате освоения дисциплины**

ПК-3 – решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности;

ПСК-2 – способностью осваивать современные и перспективные направления развития вычислительных комплексов и систем.

#### **Разделы дисциплины**

1. Симметричная многопроцессорность
2. Программирование кластерных систем
3. Программирование в рамках концепции GUDA с использованием компилятора командной строки
4. Определение параметров видеокарты с поддержкой технологии GUDA в среде Microsoft Visual Studio
5. Измерение пропускной способности памяти видеокарт с поддержкой технологии GUDA
6. Оптимизация работы с глобальной памятью для видеокарт с поддержкой технологии GUDA

МИНОБРНАУКИ РОССИИ  
Юго-Западный государственный университет

УТВЕРЖДАЮ:

И.о. декана факультета  
фундаментальной и прикладной  
информатики

 Т.А. Ширабакина

« 31 » 08 2016 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Программирование распределенных и массивно-параллельных систем

*(наименование дисциплины)*

направление подготовки бакалавров 09.03.01

*шифр согласно ФГОС*

Информатика и вычислительная техника

*наименование направления подготовки (специальности)*

Вычислительные машины, комплексы, системы и сети

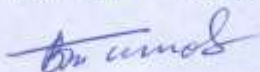
*наименование профиля*

форма обучения заочная

Рабочая программа составлена в соответствии с Федеральным государственным образовательным стандартом высшего образования направления подготовки 09.03.01 Информатика и вычислительная техника и на основании учебного плана направления подготовки 09.03.01 Информатика и вычислительная техника, одобренного Ученым советом университета протокол №7 «29» февраля 2016 г.

Рабочая программа обсуждена и рекомендована к применению в образовательном процессе для обучения студентов по направлению подготовки 09.03.01 Информатика и вычислительная техника на заседании кафедры вычислительной техники протокол № 4 от «30» 08 2016 г.

Зав. кафедрой ВТ



В.С. Титов

Разработчик программы

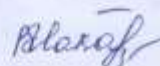
к.т.н.



Э.И. Ватутин

Согласовано:

Директор научной библиотеки



В.Г. Макаровская

Рабочая программа пересмотрена, обсуждена и рекомендована к применению в образовательном процессе на основании учебного плана направления подготовки 09.03.01 Информатика и вычислительная техника, одобренного Ученым советом университета протокол № 7 от «29» 02 2016 г. на заседании кафедры вычислительной техники протокол № 1 «29» 06 2017 г.

Зав. кафедрой ВТ



В.С. Титов

Рабочая программа пересмотрена, обсуждена и рекомендована к применению в образовательном процессе на основании учебного плана направления подготовки 09.03.01 Информатика и вычислительная техника, одобренного Ученым советом университета протокол № 7 от «29» 02 2016 г. на заседании кафедры вычислительной техники протокол № 1 «29» 06 2018 г.

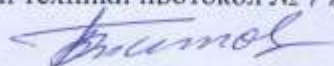
Зав. кафедрой ВТ



В.С. Титов

Рабочая программа пересмотрена, обсуждена и рекомендована к применению в образовательном процессе на основании учебного плана направления подготовки 09.03.01 Информатика и вычислительная техника, одобренного Ученым советом университета протокол № 7 от «29» 02 2016 г. на заседании кафедры вычислительной техники протокол № 18 «27» 06 2019 г.

Зав. кафедрой ВТ

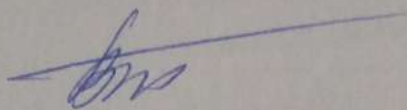


В.С. Титов



Рабочая программа пересмотрена, обсуждена и рекомендована к применению в образовательном процессе на основании учебного плана направления подготовки 09.03.01 «Информатика и вычислительная техника», утвержденного Ученым советом университета протокол № 9 «26» 03 2018 г. на заседании кафедры вычислительной техники, протокол № 17 от 02.07.2020 г.

Зав. кафедрой



В.С. Титов

Рабочая программа пересмотрена, обсуждена и рекомендована к применению в образовательном процессе на основании учебного плана направления подготовки 09.03.01 «Информатика и вычислительная техника», утвержденного Ученым советом университета протокол № \_\_ «\_\_» \_\_ 20\_\_ г. на заседании кафедры вычислительной техники, протокол № \_\_ от \_\_\_\_\_ г.

Зав. кафедрой

В.С. Титов

Рабочая программа пересмотрена, обсуждена и рекомендована к применению в образовательном процессе на основании учебного плана направления подготовки 09.03.01 «Информатика и вычислительная техника», утвержденного Ученым советом университета протокол № \_\_ «\_\_» \_\_ 20\_\_ г. на заседании кафедры вычислительной техники, протокол № \_\_ от \_\_\_\_\_ г.

Зав. кафедрой

В.С. Титов

## **1. Планируемые результаты обучения, соотнесенные с планируемыми результатами освоения ОП**

### **1.1 Цель преподавания дисциплины**

Целью изучения дисциплины «Программирование распределенных и массивно-параллельных систем» является формирование у студентов систематизированных специализированных знаний основных принципов организации параллельных вычислений с использованием метакомпьютинга и массивно-параллельных аппаратных платформ, инструментальных средств и технологий разработки ПО для них.

### **1.2 Задачи изучения дисциплины**

К задачам изучения дисциплины относятся:

- ознакомление студентов с теоретическими основами программирования параллельных вычислительных архитектур;
- ознакомление студентов с распространенными стандартами и протоколами, используемыми в параллельном программировании

### **1.3 Компетенции, формируемые в результате освоения дисциплины**

В результате освоения дисциплины студент формирует и демонстрирует следующие компетенции:

- способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности (ОПК-5);
- способностью обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности (ПК-3);
- способностью осваивать современные и перспективные направления развития вычислительных комплексов и систем (ПСК-2).

## **2. Место дисциплины в структуре образовательной программы**

Для изучения дисциплины Б1.В.ДВ.14.2 «Программирование распределенных и массивно-параллельных систем» необходимо предварительно усвоить следующие дисциплины, предусмотренные учебным планом:

- информатика;
- программирование;
- философия;
- дискретная математика;
- вычислительная математика;
- организация ЭВМ и систем;
- системное программное обеспечение;
- параллельное программирование.

Дисциплина преподается в 5–7 семестрах на 3–4 курсах.

## **3. Содержание учебной дисциплины**

### **3.1. Содержание дисциплины и лекционных занятий**

Общая трудоемкость (объем) дисциплины составляет 3 зачетных единицы (3 зе), 108 часов.

Таблица 3.1 - Объем дисциплины по видам учебных занятий

Объем дисциплины	Всего, часов
Общая трудоемкость дисциплины	108
Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)	8,2
в том числе:	
лекции	0
лабораторные занятия	8
практические занятия	0
экзамен	не предусмотрен
зачет	0,2
курсовая работа (проект)	не предусмотрена
расчетно-графическая (контрольная) работа	не предусмотрена
Аудиторная работа (всего):	54
в том числе:	
лекции	0
лабораторные занятия	8
практические занятия	0
Самостоятельная работа обучающихся (всего)	96
Контроль/экзамен (подготовка к зачету)	0

Таблица 3.1 Содержание дисциплины и ее методическое обеспечение.

№ п/п	Раздел, темы дисциплины	Виды деятельности			Учебно-методические материалы	Формы текущего контроля успеваемости	Компетенции
		лек	лаб	пр.			
1	2	3	4	5	6	7	8
1.	<b>Симметричная многопроцессорность.</b> Организация и принципы программирования		2		У1,МУ1	С, КП	ОПК-5
2.	<b>Программирование кластерных систем.</b> Организация и принципы программирования		2		У1	С, КО	ПК-3
3.	<b>Системы с массовым параллелизмом.</b> Организация и принципы программирования		4		У1	С, КО	ПСК-2
	Итого		8				

Примечание:

У – учебное пособие, учебник;

МУ – методические указания;

С – собеседование;

КО – контрольный опрос;

МК – машинный контроль.

Таблица 3.2 - Краткое содержание лекционного курса

№п.п	Раздел (тема) дисциплины	Содержание
1.	Симметричная многопроцессорность	Организация параллельного исполнения кода в операционной системе. Потoki и процессы. Примитивы синхронизации. Семафоры, мьютексы, критические секции. Проблема блокировок.
2.	Программирование кластерных систем	Принципы организации и программирования кластерных систем. Интерфейс MPI. Блокирующие и неблокирующие обмены. Примитивы синхронизации. Топологии обмена данными. Виды обменов.
3.	Системы с массовым параллелизмом	Концепция GPGPU. Организация GPU на аппаратном уровне. Организация обмена данными через шину PCI Express. Сетка, блоки и нити. Группы потоков (WARP'ы). Барьерная синхронизация. Типы памяти GPU.

### 3.2 Лабораторные и (или) практические занятия

#### 3.2.1. Лабораторные работы

Таблица 3.2 Лабораторные и практические занятия

№ п/п	Наименование лабораторной работы	Объем, час.
1.	<b>Программирование симметричных многопроцессорных систем.</b> Использование директив компилятора для подсказок к распараллеливанию (5 семестр)	2
2.	<b>Программирование кластерных систем.</b> Обзор инструментария MPI (6 семестр)	2
3.	<b>Программирование систем с массовым параллелизмом.</b> Обзор средств технологий CUDA, STREAM, OpenCL (7 семестр)	4
Итого:		8

### 3.3 Самостоятельная работа студентов (СРС)

Таблица 3.3 Самостоятельная работа студентов (СРС)

№	Наименование раздела дисциплины	Срок выполнения	Время, затрачиваемое на выполнение СРС, час
3	Введение в программирование векторных расширений	4-11	32
9	Многopоточное программирование	12-18	32
6-10	Программирование GPU	19-35	32
Итого:			96

#### 4. Учебно-методическое обеспечение для самостоятельной работы

Для выполнения самостоятельной работы достаточно ресурсов кафедры, включая:

4.1 Методические указания к выполнению лабораторных работ, которые содержатся в отдельно изданных методических указаниях и в кафедральной базе данных.

4.2 Тексты рубежных тестов кафедральной базы тестов (приложение А).

4.3 Вопросы к экзаменам, находящиеся в открытом доступе кафедральной базы данных (приложение Б).

#### 5. Образовательные технологии

Структурная составляющая компетенции **знания** формируется путем чтения лекций и выполнения части самостоятельной работы, ориентированной на приобретение знаний. Источником знаний кроме конспекта лекций являются соответствующие учебники, учебные пособия, статьи в профессиональных журналах и сведения, получаемые с помощью интернет технологий. Приобретение **умений** и **навыков** обеспечивается в ходе выполнения лабораторных работ и самостоятельной работы студентов.

Расчетный удельный вес занятий, проводимых в интерактивной форме не менее 10 часов.

Таблица 5.1 Интерактивные образовательные технологии, используемые при проведении аудиторных занятий.

№ п/п	Наименование раздела (лекции и лабораторные занятия)	Используемые интерактивные образовательные технологии	Объем в часах
1.	Грид-системы. Организация и принципы программирования	Кейс технология	2
2.	Многопоточное программирование с использованием WinAPI-функций	Диалог с аудиторией	2
3.	Симметричная многопроцессорность. Организация и принципы программирования	Кейс технология	2
4.	Программирование кластерных систем. Организация и принципы программирования	Разбор ситуаций	2
5.	Системы с массовым параллелизмом. Организация и принципы программирования	Кейс технология	2
Итого:		В часах	12

#### 6. Фонд оценочных тестов для проведения промежуточной аттестации.

Оценка успешности образовательного процесса в соответствии с принятой в университете концепцией балльно-рейтинговых оценок формируется следующим образом.

Для контроля знаний студентов в течение семестра (до экзаменационной сессии) организуется текущий контроль, в ходе которого оценивается качество усвоения студентами теоретических разделов дисциплины, знаний, умений и навыков, полученных во время лабораторных работ, а так же в ходе выполнения курсового проекта (если предусмотрено) и самостоятельной работы.

Формой промежуточного контроля по дисциплине является зачет.

Перечень вопросов к зачету:



1. Запуск двух вычислительно-интенсивных потоков на двух физических ядрах процессора в условиях отсутствия прочих конфликтов способен дать выигрыш в реальной производительности

- \* около 2 раз +
- \* 10-20%
- \* выигрыша не будет

2. Запуск двух вычислительно-интенсивных потоков на двух логических ядрах процессора в условиях отсутствия прочих конфликтов способен дать выигрыш в реальной производительности

- \* около 2 раз
- \* 10-20% +
- \* выигрыша не будет

3. Запуск двух вычислительно-интенсивных потоков на одноядерном процессоре в условиях отсутствия прочих конфликтов способен дать выигрыш в реальной производительности

- \* около 2 раз
- \* 10-20% +
- \* выигрыша не будет +

4. Запуск двух потоков с большим числом обращений в память на двух физических ядрах процессора в условиях отсутствия прочих конфликтов способен дать выигрыш в реальной производительности

- \* около 2 раз
- \* 10-20%
- \* выигрыша не будет +

5. Какие из задач наиболее неудобны для параллельных вычислений?

- \* набор независимых подзадач
- \* слабосвязанные
- \* сильносвязанные +

6. Какой из типов оптимизации кода программы не относится к программной оптимизации?

- \* высокоуровневая
- \* алгоритмическая
- \* параллельная +
- \* микроархитектурная

7. Какие задачи возможно эффективно решать с использованием грид-систем?

- \* слабосвязанные +
- \* сильносвязанные
- \* все вышеперечисленные

8. Какие задачи возможно эффективно решать с использованием суперкомпьютеров?

- \* слабосвязанные
- \* сильносвязанные
- \* все вышеперечисленные +

9. Программный интерфейс MPI применяется для

- \* программирования под кластеры и суперкомпьютеры +
- \* программирования под GPU

- \* программирования под грид-системы
- \* программирования векторных расширений

10. Программный интерфейс OpenMP применяется для
- \* программирования под кластеры и суперкомпьютеры
  - \* программирования под GPU
  - \* программирования под грид-системы
  - \* программирования векторных расширений
  - \* многопоточного программирования +

## 6.1 Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

Таблица 6.1 Этапы формирования компетенции

Код компетенции, содержание компетенции	Дисциплины (модули), при изучении которых формируется данная компетенция
<p>способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности (ОПК-5)</p>	<p>математика, инженерная и компьютерная графика, сети и телекоммуникации, защита информации, введение в направление подготовки, алгебра и геометрия, микропроцессорные системы, параллельное программирование, теоретические основы организации многопроцессорных комплексов и систем, методы оптимизации, программирование, математические основы теории бифуркаций электронных схем, основы комбинаторной оптимизации, теория принятия решений, теория нечёткой логики и множеств, моделирование, математические основы теории динамических систем, радиоприёмные устройства, комплексы бортового оборудования, основы построения систем цифровой обработки сигналов, информационные технологии проектирования авионики, организация систем искусственного интеллекта, системное проектирование радиоэлектронной аппаратуры, теория проектирования приборных комплексов, основы комплексной автоматизации проектирования ЭВМ, антенно-фидерные устройства систем цифровой обработки сигналов, конструирование и стандартизация, организация и методология научных исследований, программирование распределённых и массивно-параллельных систем</p>
<p>способностью обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности (ПК-3)</p>	<p>Математика, электротехника, электроника и схемотехника, сети и телекоммуникации, защита информации, метрология, стандартизация и сертификация, введение в направление подготовки, алгебра и геометрия, параллельное программирование, теоретические основы организации многопроцессорных комплексов и систем, методы оптимизации, математические основы теории бифуркаций электронных схем, основы комбинаторной оптимизации, теория</p>

	<p>принятия решений, теория нечёткой логики и множеств, моделирование, математические основы теории динамических систем, радиоприёмные устройства, комплексы бортового оборудования, основы построения систем цифровой обработки сигналов, информационные технологии проектирования авионики, системное проектирование радиоэлектронной аппаратуры, теория проектирования приборных комплексов, основы теории цепей и сигналов, вычислительные системы повышенной надёжности, антенно-фидерные устройства систем цифровой обработки сигналов, конструирование и стандартизация, программирование распределённых и массивно-параллельных систем</p>
<p>способностью осваивать современные и перспективные направления развития вычислительных комплексов и систем (ПСК-2)</p>	<p>ЭВМ и периферийные устройства. Защита информации. Введение в направление подготовки. Теоретические основы организации многопроцессорных комплексов и систем. Специальные процессоры, машины и сети. Микропроцессорные системы для автоматизации технологических процессов. Микропроцессорные системы в системах передачи и обработки данных. Технические и программные средства комплексного моделирования и стендовой отладки бортовых систем. Информационные технологии проектирования авионики. Организация систем искусственного интеллекта. Организация и методология научных исследований. Программирование распределённых и массивно-параллельных систем.</p>

Оценивание компетенций, формируемых в ходе выполнения и защиты лабораторных работ в виде балльной оценки, осуществляется в соответствии с таблицей 7.1 раздела 7. Оценивание компетенций, сформированных в ходе лекционных занятий, производится с помощью рубежного теста по основному разделу курса. Описание тестовых материалов приведено выше. Каждый правильный ответ на вопрос теста оценивается в 0,2 балла. За тест студент может получить 10 баллов при правильном ответе на все вопросы.

Оценка знаний на экзамене осуществляется путем ответов на вопросы билета

В приложении Б приведены списки вопросов, каждый из которых оценен определенным числом баллов. Максимальное число баллов, которые можно набрать за экзамен - 36.

Регламент, определяющий процедуры оценивания знаний, умений и навыков определен положением ЮЗГУ П02.016-2012 «О балльно-рейтинговой оценке качества освоения образовательных программ».

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

Таблица 6.2 Показатели и критерии определения уровня сформированности компетенций (частей компетенций)

№	Код компе-	Уровни сформированности компетенции
---	------------	-------------------------------------

п/п	тенции (или ее части)	Пороговый (удовлетворительный)	Продвинутый (хорошо)	Высокий (отличный)
1	ОПК-5	<p>Знать:</p> <ul style="list-style-type: none"> <li>- базовые положения параллельного программирования.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- применять основные методы построения параллельных программ.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- основами нормативных документов и стандартов в области разработки параллельных программ.</li> </ul>	<p>Знать:</p> <ul style="list-style-type: none"> <li>- базовые положения параллельного программирования;</li> <li>- области применения специализированных параллельных программ.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- применять основные методы построения параллельных программ,</li> <li>- применять методы анализа и синтеза параллельных программ.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- основами нормативных документов и стандартов в области разработки параллельных программ</li> </ul>	<p>Знать:</p> <ul style="list-style-type: none"> <li>- базовые положения параллельного программирования;</li> <li>- области применения специализированных параллельных программ;</li> <li>- принципы и основные структурные преобразования программ на различных уровнях оптимизации.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- применять основные методы построения параллельных программ;</li> <li>- применять методы анализа и синтеза параллельных программ;</li> <li>- проектировать, отлаживать и сопровождать специализированные параллельные программы.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- основами нормативных документов и стандартов в области разработки параллельных программ.</li> </ul>
2	ПК-3	<p>Знать:</p> <ul style="list-style-type: none"> <li>- Этапы, технологию разработки и состав технического задания на выполнение проектных работ в сфере параллельных вычислительных архитектур.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- Составлять технические задания на разработку и проектирование параллельных программ.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- Методикой проектирования и</li> </ul>	<p>Знать:</p> <ul style="list-style-type: none"> <li>- Этапы, технологию разработки и состав технического задания на выполнение проектных работ в сфере параллельных вычислительных архитектур,</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- методы анализа эффективности реализации специализированных процессоров.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- Составлять технические задания на разработку и</li> </ul>	<p>Знать:</p> <ul style="list-style-type: none"> <li>- Этапы, технологию разработки и состав технического задания на выполнение проектных работ в сфере параллельных вычислительных архитектур,</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- методы анализа эффективности реализации параллельных программ,</li> <li>- принципы оптимизации программных средств с последовательной и параллельной архитектурой,</li> </ul>

		<p>программирования параллельных вычислительных систем.</p>	<p>проектирование параллельных программ</p> <ul style="list-style-type: none"> <li>- выполнять проектирование и оценку качества параллельных программ.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- Методикой проектирования и программирования параллельных вычислительных систем.</li> </ul>	<ul style="list-style-type: none"> <li>- основы моделирования и расчетов эффективности последовательных и параллельных программ.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- Составлять технические задания на разработку и проектирование параллельных программ;</li> <li>- выполнять проектирование ПО с параллельной архитектурой;</li> <li>- выполнять компьютерное параллельных программ.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- Методикой проектирования и программирования параллельных вычислительных систем.</li> </ul>
3	ПСК-2	<p>Знать:</p> <ul style="list-style-type: none"> <li>- Местонахождение информации по параллельному программированию;</li> <li>- Теоретические основы организации параллельных вычислительных архитектур.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- Анализировать документы с изложением технологий и стандартов параллельного программирования.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- Навыками работы с технической литературой в области параллельного программирования.</li> </ul>	<p>Знать:</p> <ul style="list-style-type: none"> <li>- Расположение информации по параллельному программированию в сети Интернет;</li> <li>- Этапы, технологию разработки и состав технического задания на выполнение проектных работ в сфере параллельных вычислительных архитектур.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- Искать и анализировать наиболее актуальную информацию по параллельному программированию в сети Интернет.</li> <li>- Осуществлять организацию и техническое оснащение рабочих</li> </ul>	<p>Знать:</p> <ul style="list-style-type: none"> <li>- Основные категории и понятия производственного менеджмента;</li> <li>- Методы проектирования и настройки компонентов параллельных вычислительных систем;</li> <li>- Архитектуру, протоколы и технические средства систем параллельного программирования.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- Моделировать компоненты параллельных программ.</li> <li>- Составлять технические задания на разработку и проектирование параллельных программ.</li> <li>- Разрабатывать логические, электрические и топологические схемы</li> </ul>

			<p>мест.</p> <ul style="list-style-type: none"> <li>- Настраивать основные сервисы локальных сетей, конфигурировать параллельные программы.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- Навыками работы со справочными и обучающими системами, доступными в компьютерных сетях.</li> <li>- Навыками составления спецификаций оборудования систем параллельного программирования.</li> <li>- Навыками конфигурирования параллельных вычислительных средств.</li> </ul>	<p>систем параллельного программирования.</p> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- Навыками составления спецификаций оборудования аппаратного обеспечения параллельной архитектурой.</li> <li>- Методикой проектирования и программирования параллельных вычислительных систем.</li> <li>- Методикой моделирования параллельных вычислительных архитектур.</li> </ul>	с	и
--	--	--	---	---	---	---

Таблица 6.3 Паспорт комплекта оценочных средств

№	Раздел дисциплины (тема)	Код контролируемой компетенции (или ее части)	Технология формирования	Оценочные средства		Описание шкал оценивания
				наименование	№ заданий	
1.	<b>Программирование симметричных многопроцессорных систем.</b> Использование директив компилятора для подсказок к распараллеливанию (5 семестр)	ОПК-5	Лабораторная работа, СРС	тесты	1-11	согласно табл. 7.1
2.	<b>Программирование кластерных систем.</b> Обзор инструментария MPI (6 семестр)	ПК-3	Лабораторная работа, СРС	контрольные вопросы	1-5	согласно табл. 6.4, табл. 7.1



3.	<b>Программирование систем с массовым параллелизмом.</b> Обзор средств технологий CUDA, STREAM, OpenCL (7 семестр)	ПСК-2	Лабораторная работа, СРС	контрольные вопросы	1-9	согласно табл. 6.4, табл. 7.1
----	---	-------	--------------------------	---------------------	-----	-------------------------------

**Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования компетенций:**

- Положение П02.016-2015 «О балльно-рейтинговой системе оценки качества освоения образовательных программ»;
- Список методических указаний, используемых в образовательном процессе, представлен в п.8.2;

Оценочные средства представлены в учебно-методическом комплексе дисциплины.

**6.3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования компетенций в процессе освоения образовательной программы:**

1. Инструментарий CUDA применяется для

- \* программирования под кластеры и суперкомпьютеры
- \* программирования под GPU +
- \* программирования под грид-системы
- \* программирования векторных расширений
- \* многопоточного программирования

2. Какой из типов оптимизации кода программы дает максимальный выигрыш?

- \* высокоуровневая
- \* алгоритмическая +
- \* микроархитектурная

3. Вынос инвариантов цикла является примером оптимизации следующего типа

- \* высокоуровневая +
- \* алгоритмическая
- \* микроархитектурная

4. Раскрутка циклов является примером оптимизации следующего типа

- \* высокоуровневая +
- \* алгоритмическая
- \* микроархитектурная

5. Изменение алгоритмов работы программы является примером оптимизации следующего типа

- \* высокоуровневая
- \* алгоритмическая +
- \* микроархитектурная

6. Приспособление программы к особенностям конкретного процессора является примером оптимизации следующего типа

- \* высокоуровневая
- \* алгоритмическая
- \* микроархитектурная +

7. Согласно закону Амдала выигрыш от распараллеливания программы ограничивается

- \* ее последовательной частью +
- \* числом процессоров
- \* числом потоков
- \* особенностями микроархитектуры процессора

8. Какая из перечисленных технологий не относится к векторным расширениям системы команд процессора?

- \* MMX
- \* SSE
- \* AVX
- \* AES +

9. Код с плавающей точкой какого компилятора характеризуется наихудшим качеством оптимизации?

- \* Borland C++ Builder +
- \* Intel C++ Compiler
- \* Microsoft Visual C++
- \* GCC

10. Какие из перечисленных оптимизаций не может выполнять компилятор?

- \* раскрутка циклов
- \* ликвидация хвостовой рекурсии
- \* удаление константных параметров подпрограмм
- \* распараллеливание кода под MPI +
- \* векторизация кода под AVX

## 7. Рейтинговый контроль изучения дисциплины

Таблица 7.1 контроль изучения дисциплины

Форма контроля	Минимальный балл		Максимальный балл	
	балл	примечание	балл	примечание
1	2	3	4	5
Многопоточное программирование с использованием WinAPI. Обзор инструментария WinAPI-функций и его использование для организации параллельных вычислений	4	Выполнил, но не защитил	7	Выполнил и защитил
Программирование симметричных многопроцессорных систем. Использование директив компилятора для подсказок к	4	Выполнил, но не защитил	7	Выполнил и защитил

распараллеливанию				
Программирование кластерных систем. Обзор инструментария MPI	4	Выполнил, но не защитил	7	Выполнил и защитил
Программирование систем с массовым параллелизмом. Обзор средств технологий CUDA, STREAM, OpenCL	4	Выполнил, но не защитил	7	Выполнил и защитил
Творческая компонента	0	Не участвовал	15	За участие в конференциях, публикации, задания повышенной сложности индивидуальные научные исследования и т.д.
Зачет	0	Не ответил ни на один вопрос	36	Ответил на все вопросы
Итого	16		100	

## 8. Учебно-методическое и информационное обеспечение учебной дисциплины

### 8.1 Основная литература и дополнительная литература.

#### а) Основная литература

1. Сальников А.М., Ярошенко Е.А., Гребенник О.С., Спиридонов С.В. Введение в параллельные вычисления. Основы программирования на языке СИ с использованием интерфейса MPI. - М.: ИПУ РАН, 2009 - 123 с. <http://window.edu.ru/resource/338/66338>
2. Гергель В.П. Высокопроизводительные вычисления для многоядерных многопроцессорных систем: Учебное пособие. - Нижний Новгород; Изд-во ННГУ им. Н.И. Лобачевского, 2010. - 420 с. <http://window.edu.ru/resource/858/79858>

#### б) Дополнительная литература

3. Антонов А.С. Параллельное программирование с использованием технологии OpenMP. М.: Изд-во МГУ, 2009. 76 с. (электронный ресурс <http://window.edu.ru/resource/291/61291>)
4. Антонов А.С. Параллельное программирование с использованием технологии MPI. М.: Изд-во МГУ, 2004. 71 с. (электронный ресурс)
5. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. СПб.: БХВ-Петербург, 2002. 608 с.

### 8.2 Перечень методических указаний

Разработка программ с поддержкой технологии CUDA с использованием компилятора командной строки / Ватутин Э.И., ЮЗГУ, 2016.

Определение параметров видеокарты с поддержкой технологии CUDA в среде Microsoft Visual Studio / Ватутин Э.И., ЮЗГУ, 2016.

Измерение пропускной способности памяти видеокарт с поддержкой технологии CUDA / Ватутин Э.И., ЮЗГУ, 2016.

Оптимизация работы с глобальной памятью для видеокарт с поддержкой технологии CUDA / Ватутин Э.И., ЮЗГУ, 2016.

### **8.3 Перечень ресурсов информационно-телекоммуникационной сети Интернет.**

Обязательные Интернет ресурсы отсутствуют.

### **8.4. Перечень информационных технологий.**

1. Среда разработки ПО RAD Studio, Visual Studio.
2. Операционная система Windows XP.

### **8.5 Другие учебно-методические материалы.**

Отсутствуют.

## **9. Материально-техническое обеспечение учебной дисциплины.**

Стандартно оборудованные лекционные аудитории. Для проведения отдельных занятий (по заявке) - выделение компьютерного класса, а также аудитории для проведения интерактивных лекций: видеопроектор, экран настенный, др. оборудование.

Рабочие места студентов должны быть оснащены оборудованием не ниже: Pentium III-800/ОЗУ-256 Мб / Video с поддержкой GPGPU / Sound card – 16bit /Headphones / HDD 80 Гб / CD-ROM – 48x / Network adapter – 10/100/ Мбс / SVGA – 19”.



МИНОБРАЗОВАНИЯ РОССИИ  
Юго-Западный государственный университет

УТВЕРЖДАЮ:

И.о. декана факультета  
фундаментальной и прикладной  
информатики

 Т.А. Ширабакина

« 31 » 08 2016 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Программирование распределенных и массивно-параллельных систем

*(наименование дисциплины)*

направление подготовки бакалавров 09.03.01

*цифр согласно ФГОС*

Информатика и вычислительная техника

*наименование направления подготовки (специальности)*

Вычислительные машины, комплексы, системы и сети

*наименование профиля*

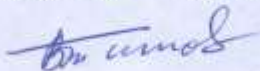
форма обучения очная



Рабочая программа составлена в соответствии с Федеральным государственным образовательным стандартом высшего образования направления подготовки 09.03.01 Информатика и вычислительная техника и на основании учебного плана направления подготовки 09.03.01 Информатика и вычислительная техника, одобренного Ученым советом университета протокол №7 «29» февраля 2016 г.

Рабочая программа обсуждена и рекомендована к применению в образовательном процессе для обучения студентов по направлению подготовки 09.03.01 Информатика и вычислительная техника на заседании кафедры вычислительной техники протокол № 4 от «30» 08 2016 г.

Зав. кафедрой ВТ



В.С. Титов

Разработчик программы

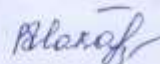
к.т.н.



Э.И. Ватутин

Согласовано:

Директор научной библиотеки



В.Г. Макаровская

Рабочая программа пересмотрена, обсуждена и рекомендована к применению в образовательном процессе на основании учебного плана направления подготовки 09.03.01 Информатика и вычислительная техника, одобренного Ученым советом университета протокол № 7 от «29» 02 2016 г. на заседании кафедры вычислительной техники протокол № 1 «29» 06 2017 г.

Зав. кафедрой ВТ



В.С. Титов

Рабочая программа пересмотрена, обсуждена и рекомендована к применению в образовательном процессе на основании учебного плана направления подготовки 09.03.01 Информатика и вычислительная техника, одобренного Ученым советом университета протокол № 7 от «29» 02 2016 г. на заседании кафедры вычислительной техники протокол № 1 «29» 06 2018 г.

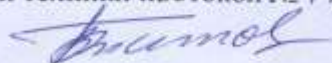
Зав. кафедрой ВТ



В.С. Титов

Рабочая программа пересмотрена, обсуждена и рекомендована к применению в образовательном процессе на основании учебного плана направления подготовки 09.03.01 Информатика и вычислительная техника, одобренного Ученым советом университета протокол № 7 от «29» 02 2016 г. на заседании кафедры вычислительной техники протокол № 18 «27» 06 2019 г.

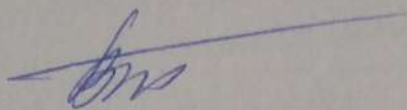
Зав. кафедрой ВТ



В.С. Титов

Рабочая программа пересмотрена, обсуждена и рекомендована к применению в образовательном процессе на основании учебного плана направления подготовки 09.03.01 «Информатика и вычислительная техника», утвержденного Ученым советом университета протокол № 9 «26» 03 2018 г. на заседании кафедры вычислительной техники, протокол № 17 от 02.07.2020 г.

Зав. кафедрой



В.С. Титов

Рабочая программа пересмотрена, обсуждена и рекомендована к применению в образовательном процессе на основании учебного плана направления подготовки 09.03.01 «Информатика и вычислительная техника», утвержденного Ученым советом университета протокол № \_\_ «\_\_» \_\_ 20\_\_ г. на заседании кафедры вычислительной техники, протокол № \_\_ от \_\_\_\_\_ г.

Зав. кафедрой

В.С. Титов

Рабочая программа пересмотрена, обсуждена и рекомендована к применению в образовательном процессе на основании учебного плана направления подготовки 09.03.01 «Информатика и вычислительная техника», утвержденного Ученым советом университета протокол № \_\_ «\_\_» \_\_ 20\_\_ г. на заседании кафедры вычислительной техники, протокол № \_\_ от \_\_\_\_\_ г.

Зав. кафедрой

В.С. Титов

## **1. Планируемые результаты обучения, соотнесенные с планируемыми результатами освоения ОП**

### **1.1 Цель преподавания дисциплины**

Целью изучения дисциплины «Программирование распределенных и массивно-параллельных систем» является формирование у студентов систематизированных специализированных знаний основных принципов организации параллельных вычислений с использованием метакомпьютинга и массивно-параллельных аппаратных платформ, инструментальных средств и технологий разработки ПО для них.

### **1.2 Задачи изучения дисциплины**

К задачам изучения дисциплины относятся:

- ознакомление студентов с теоретическими основами программирования параллельных вычислительных архитектур;
- ознакомление студентов с распространенными стандартами и протоколами, используемыми в параллельном программировании

### **1.3 Компетенции, формируемые в результате освоения дисциплины**

В результате освоения дисциплины студент формирует и демонстрирует следующие компетенции:

- способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности (ОПК-5);
- способностью обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности (ПК-3);
- способностью осваивать современные и перспективные направления развития вычислительных комплексов и систем (ПСК-2).

## **2. Место дисциплины в структуре образовательной программы**

Для изучения дисциплины Б1.В.ДВ.14.2 «Программирование распределенных и массивно-параллельных систем» необходимо предварительно усвоить следующие дисциплины, предусмотренные учебным планом:

- информатика;
- программирование;
- философия;
- дискретная математика;
- вычислительная математика;
- организация ЭВМ и систем;
- системное программное обеспечение;
- параллельное программирование.

Дисциплина преподается в 5–7 семестрах на 3–4 курсах.

## **3. Содержание учебной дисциплины**

### **3.1. Содержание дисциплины и лекционных занятий**

Общая трудоемкость (объем) дисциплины составляет 3 зачетных единицы (3 зе), 108 часов.

Таблица 3.1 - Объем дисциплины по видам учебных занятий

Объем дисциплины	Всего, часов
Общая трудоемкость дисциплины	108
Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)	54,2
в том числе:	
лекции	0
лабораторные занятия	54
практические занятия	0
экзамен	не предусмотрен
зачет	0,6
курсовая работа (проект)	не предусмотрена
расчетно-графическая (контрольная) работа	не предусмотрена
Аудиторная работа (всего):	54
в том числе:	
лекции	0
лабораторные занятия	54
практические занятия	0
Самостоятельная работа обучающихся (всего)	54
Контроль/экзамен (подготовка к зачету)	0

Таблица 3.1 Содержание дисциплины и ее методическое обеспечение.

№ п/п	Раздел, темы дисциплины	Виды деятельности			Учебно-методические материалы	Формы текущего контроля успеваемости	Компетенции
		лек	лаб	пр.			
1	2	3	4	5	6	7	8
1.	<b>Симметричная многопроцессорность.</b> Организация и принципы программирования (5 семестр)		18		У1,МУ1	С, КП	ОПК-5
2.	<b>Программирование кластерных систем.</b> Организация и принципы программирования (6 семестр)		18		У1	С, КО	ПК-3
3.	<b>Системы с массовым параллелизмом.</b> Организация и принципы программирования (7 семестр)		18		У1	С, КО	ПСК-2
	Итого		54				

Примечание:

У – учебное пособие, учебник;

МУ – методические указания;

С – собеседование;  
 КО – контрольный опрос;  
 МК – машинный контроль.

Таблица 3.2 - Краткое содержание лекционного курса

№п.п	Раздел (тема) дисциплины	Содержание
1.	Симметричная многопроцессорность	Организация параллельного исполнения кода в операционной системе. Потoki и процессы. Примитивы синхронизации. Семафоры, мьютексы, критические секции. Проблема блокировок.
2.	Программирование кластерных систем	Принципы организации и программирования кластерных систем. Интерфейс MPI. Блокирующие и неблокирующие обмены. Примитивы синхронизации. Топологии обмена данными. Виды обменов.
3.	Системы с массовым параллелизмом	Концепция GPGPU. Организация GPU на аппаратном уровне. Организация обмена данными через шину PCI Express. Сетка, блоки и нити. Группы потоков (WARP'ы). Барьерная синхронизация. Типы памяти GPU.

### 3.2 Лабораторные и (или) практические занятия

#### 3.2.1. Лабораторные работы

Таблица 3.2 Лабораторные и практические занятия

№ п/п	Наименование лабораторной работы	Объем, час.
1.	<b>Программирование симметричных многопроцессорных систем.</b> Использование директив компилятора для подсказок к распараллеливанию (5 семестр)	18
2.	<b>Программирование кластерных систем.</b> Обзор инструментария MPI (6 семестр)	18
3.	<b>Программирование систем с массовым параллелизмом.</b> Обзор средств технологий CUDA, STREAM, OpenCL (7 семестр)	18
Итого:		54

### 3.3 Самостоятельная работа студентов (СРС)

Таблица 3.3 Самостоятельная работа студентов (СРС)

№	Наименование раздела дисциплины	Срок выполнения	Время, затрачиваемое на выполнение СРС, час
3	Введение в программирование векторных	4-11	18

	расширений		
9	Многопоточное программирование	12-18	18
6-10	Программирование GPU	19-35	18
Итого:			54

#### 4. Учебно-методическое обеспечение для самостоятельной работы

Для выполнения самостоятельной работы достаточно ресурсов кафедры, включая:

4.1 Методические указания к выполнению лабораторных работ, которые содержатся в отдельно изданных методических указаниях и в кафедральной базе данных.

4.2 Тексты рубежных тестов кафедральной базы тестов (приложение А).

4.3 Вопросы к экзаменам, находящиеся в открытом доступе кафедральной базы данных (приложение Б).

#### 5. Образовательные технологии

Структурная составляющая компетенции **знания** формируется путем чтения лекций и выполнения части самостоятельной работы, ориентированной на приобретение знаний. Источником знаний кроме конспекта лекций являются соответствующие учебники, учебные пособия, статьи в профессиональных журналах и сведения, получаемые с помощью интернет технологий. Приобретение **умений** и **навыков** обеспечивается в ходе выполнения лабораторных работ и самостоятельной работы студентов.

Расчетный удельный вес занятий, проводимых в интерактивной форме не менее 10 часов.

Таблица 5.1 Интерактивные образовательные технологии, используемые при проведении аудиторных занятий.

№ п/п	Наименование раздела (лекции и лабораторные занятия)	Используемые интерактивные образовательные технологии	Объем в часах
1.	Грид-системы. Организация и принципы программирования	Кейс технология	2
2.	Многопоточное программирование с использованием WinAPI-функций	Диалог с аудиторией	2
3.	Симметричная многопроцессорность. Организация и принципы программирования	Кейс технология	2
4.	Программирование кластерных систем. Организация и принципы программирования	Разбор ситуаций	2
5.	Системы с массовым параллелизмом. Организация и принципы программирования	Кейс технология	2
Итого:		В часах	12

#### 6. Фонд оценочных тестов для проведения промежуточной аттестации.

Оценка успешности образовательного процесса в соответствии с принятой в университете концепцией балльно-рейтинговых оценок формируется следующим образом.

Для контроля знаний студентов в течение семестра (до экзаменационной сессии) организуется текущий контроль, в ходе которого оценивается качество усвоения студентами теоретических разделов дисциплины, знаний, умений и навыков, полученных во время



лабораторных работ, а так же в ходе выполнения курсового проекта (если предусмотрено) и самостоятельной работы.

Формой промежуточного контроля по дисциплине является зачет.

Перечень вопросов к зачету:

1. Запуск двух вычислительно-интенсивных потоков на двух физических ядрах процессора в условиях отсутствия прочих конфликтов способен дать выигрыш в реальной производительности
  - \* около 2 раз +
  - \* 10-20%
  - \* выигрыша не будет
  
2. Запуск двух вычислительно-интенсивных потоков на двух логических ядрах процессора в условиях отсутствия прочих конфликтов способен дать выигрыш в реальной производительности
  - \* около 2 раз
  - \* 10-20% +
  - \* выигрыша не будет
  
3. Запуск двух вычислительно-интенсивных потоков на одноядерном процессоре в условиях отсутствия прочих конфликтов способен дать выигрыш в реальной производительности
  - \* около 2 раз
  - \* 10-20% +
  - \* выигрыша не будет +
  
4. Запуск двух потоков с большим числом обращений в память на двух физических ядрах процессора в условиях отсутствия прочих конфликтов способен дать выигрыш в реальной производительности
  - \* около 2 раз
  - \* 10-20%
  - \* выигрыша не будет +
  
5. Какие из задач наиболее неудобны для параллельных вычислений?
  - \* набор независимых подзадач
  - \* слабосвязанные
  - \* сильносвязанные +
  
6. Какой из типов оптимизации кода программы не относится к программной оптимизации?
  - \* высокоуровневая
  - \* алгоритмическая
  - \* параллельная +
  - \* микроархитектурная
  
7. Какие задачи возможно эффективно решать с использованием грид-систем?
  - \* слабосвязанные +
  - \* сильсвязанные
  - \* все вышеперечисленные
  
8. Какие задачи возможно эффективно решать с использованием суперкомпьютеров?
  - \* слабосвязанные
  - \* сильсвязанные
  - \* все вышеперечисленные +

9. Программный интерфейс MPI применяется для

- \* программирования под кластеры и суперкомпьютеры +
- \* программирования под GPU
- \* программирования под грид-системы
- \* программирования векторных расширений

10. Программный интерфейс OpenMP применяется для

- \* программирования под кластеры и суперкомпьютеры
- \* программирования под GPU
- \* программирования под грид-системы
- \* программирования векторных расширений
- \* многопоточного программирования +

## 6.1 Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

Таблица 6.1 Этапы формирования компетенции

Код компетенции, содержание компетенции	Дисциплины (модули), при изучении которых формируется данная компетенция
способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности (ОПК-5)	математика, инженерная и компьютерная графика, сети и телекоммуникации, защита информации, введение в направление подготовки, алгебра и геометрия, микропроцессорные системы, параллельное программирование, теоретические основы организации многопроцессорных комплексов и систем, методы оптимизации, программирование, математические основы теории бифуркаций электронных схем, основы комбинаторной оптимизации, теория принятия решений, теория нечёткой логики и множеств, моделирование, математические основы теории динамических систем, радиоприёмные устройства, комплексы бортового оборудования, основы построения систем цифровой обработки сигналов, информационные технологии проектирования авионики, организация систем искусственного интеллекта, системное проектирование радиоэлектронной аппаратуры, теория проектирования приборных комплексов, основы комплексной автоматизации проектирования ЭВМ, антенно-фидерные устройства систем цифровой обработки сигналов, конструирование и стандартизация, организация и методология научных исследований, программирование распределённых и массивно-параллельных систем
способностью обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности (ПК-3)	Математика, электротехника, электроника и схемотехника, сети и телекоммуникации, защита информации, метрология, стандартизация и сертификация, введение в направление подготовки, алгебра и геометрия, параллельное программирование, теоретические основы

	<p>организации многопроцессорных комплексов и систем, методы оптимизации, математические основы теории бифуркаций электронных схем, основы комбинаторной оптимизации, теория принятия решений, теория нечёткой логики и множеств, моделирование, математические основы теории динамических систем, радиоприёмные устройства, комплексы бортового оборудования, основы построения систем цифровой обработки сигналов, информационные технологии проектирования авионики, системное проектирование радиоэлектронной аппаратуры, теория проектирования приборных комплексов, основы теории цепей и сигналов, вычислительные системы повышенной надёжности, антенно-фидерные устройства систем цифровой обработки сигналов, конструирование и стандартизация, программирование распределённых и массивно-параллельных систем</p>
<p>способностью осваивать современные и перспективные направления развития вычислительных комплексов и систем (ПСК-2)</p>	<p>ЭВМ и периферийные устройства. Защита информации. Введение в направление подготовки. Теоретические основы организации многопроцессорных комплексов и систем. Специальные процессоры, машины и сети. Микропроцессорные системы для автоматизации технологических процессов. Микропроцессорные системы в системах передачи и обработки данных. Технические и программные средства комплексного моделирования и стендовой отладки бортовых систем. Информационные технологии проектирования авионики. Организация систем искусственного интеллекта. Организация и методология научных исследований. Программирование распределённых и массивно-параллельных систем.</p>

Оценивание компетенций, формируемых в ходе выполнения и защиты лабораторных работ в виде балльной оценки, осуществляется в соответствии с таблицей 7.1 раздела 7. Оценивание компетенций, сформированных в ходе лекционных занятий, производится с помощью рубежного теста по основному разделу курса. Описание тестовых материалов приведено выше. Каждый правильный ответ на вопрос теста оценивается в 0,2 балла. За тест студент может получить 10 баллов при правильном ответе на все вопросы.

Оценка знаний на экзамене осуществляется путем ответов на вопросы билета

В приложении Б приведены списки вопросов, каждый из которых оценен определенным числом баллов. Максимальное число баллов, которые можно набрать за экзамен - 36.

Регламент, определяющий процедуры оценивания знаний, умений и навыков определен положением ЮЗГУ П02.016-2012 «О балльно-рейтинговой оценке качества освоения образовательных программ».

## **6.2 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания**

Таблица 6.2 Показатели и критерии определения уровня сформированности компетенций (частей компетенций)

№ п/п	Код компетенции (или ее части)	Уровни сформированности компетенции		
		Пороговый (удовлетворительный)	Продвинутый (хорошо)	Высокий (отличный)
1	ОПК-5	<p>Знать:</p> <ul style="list-style-type: none"> <li>- базовые положения параллельного программирования.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- применять основные методы построения параллельных программ.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- основами нормативных документов и стандартов в области разработки параллельных программ.</li> </ul>	<p>Знать:</p> <ul style="list-style-type: none"> <li>- базовые положения параллельного программирования;</li> <li>- области применения специализированных параллельных программ.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- применять основные методы построения параллельных программ,</li> <li>- применять методы анализа и синтеза параллельных программ.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- основами нормативных документов и стандартов в области разработки параллельных программ</li> </ul>	<p>Знать:</p> <ul style="list-style-type: none"> <li>- базовые положения параллельного программирования;</li> <li>- области применения специализированных параллельных программ;</li> <li>- принципы и основные структурные преобразования программ на различных уровнях оптимизации.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- применять основные методы построения параллельных программ;</li> <li>- применять методы анализа и синтеза параллельных программ;</li> <li>- проектировать, отлаживать и сопровождать специализированные параллельные программы.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- основами нормативных документов и стандартов в области разработки параллельных программ.</li> </ul>
2	ПК-3	<p>Знать:</p> <ul style="list-style-type: none"> <li>- Этапы, технологию разработки и состав технического задания на выполнение проектных работ в сфере параллельных вычислительных архитектур.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- Составлять технические задания на разработку и проектирование параллельных</li> </ul>	<p>Знать:</p> <ul style="list-style-type: none"> <li>- Этапы, технологию разработки и состав технического задания на выполнение проектных работ в сфере параллельных вычислительных архитектур,</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- методы анализа эффективности реализации специализированных процессоров.</li> </ul>	<p>Знать:</p> <ul style="list-style-type: none"> <li>- Этапы, технологию разработки и состав технического задания на выполнение проектных работ в сфере параллельных вычислительных архитектур,</li> <li>- методы анализа эффективности реализации параллельных программ,</li> <li>- принципы оптимизации</li> </ul>

		<p>программ. Владеть:</p> <ul style="list-style-type: none"> <li>- Методикой проектирования и программирования параллельных вычислительных систем.</li> </ul>	<p>Уметь:</p> <ul style="list-style-type: none"> <li>- Составлять технические задания на разработку и проектирование параллельных программ</li> <li>- выполнять проектирование и оценку качества параллельных программ.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- Методикой проектирования и программирования параллельных вычислительных систем.</li> </ul>	<p>программных средств с последовательной и параллельной архитектурой,</p> <ul style="list-style-type: none"> <li>- основы моделирования и расчетов эффективности последовательных и параллельных программ.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- Составлять технические задания на разработку и проектирование параллельных программ;</li> <li>- выполнять проектирование ПО с параллельной архитектурой;</li> <li>- выполнять компьютерное параллельных программ.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- Методикой проектирования и программирования параллельных вычислительных систем.</li> </ul>
3	ПСК-2	<p>Знать:</p> <ul style="list-style-type: none"> <li>- Местонахождение информации по параллельному программированию;</li> <li>- Теоретические основы организации параллельных вычислительных архитектур.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- Анализировать документы с изложением технологий и стандартов параллельного программирования.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- Навыками работы с технической литературой в области параллельного программирования.</li> </ul>	<p>Знать:</p> <ul style="list-style-type: none"> <li>- Расположение информации по параллельному программированию в сети Интернет;</li> <li>- Этапы, технологию разработки и состав технического задания на выполнение проектных работ в сфере параллельных вычислительных архитектур.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- Искать и анализировать наиболее актуальную информацию по параллельному программированию в сети Интернет.</li> </ul>	<p>Знать:</p> <ul style="list-style-type: none"> <li>- Основные категории и понятия производственного менеджмента;</li> <li>- Методы проектирования и настройки компонентов параллельных вычислительных систем;</li> <li>- Архитектуру, протоколы и технические средства систем параллельного программирования.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>- Моделировать компоненты параллельных программ.</li> <li>- Составлять технические задания на разработку и проектирование параллельных программ.</li> </ul>

			<ul style="list-style-type: none"> <li>- Осуществлять организацию и техническое оснащение рабочих мест.</li> <li>- Настраивать основные сервисы локальных сетей, конфигурировать параллельные программы.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- Навыками работы со справочными и обучающими системами, доступными в компьютерных сетях.</li> <li>- Навыками составления спецификаций оборудования систем параллельного программирования.</li> <li>- Навыками конфигурирования параллельных вычислительных средств.</li> </ul>	<ul style="list-style-type: none"> <li>- Разрабатывать логические, электрические и топологические схемы систем параллельного программирования.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>- Навыками составления спецификаций оборудования аппаратного обеспечения параллельной архитектурой.</li> <li>- Методикой проектирования и программирования параллельных вычислительных систем.</li> <li>- Методикой моделирования параллельных вычислительных архитектур.</li> </ul>
--	--	--	--	---

Таблица 6.3 Паспорт комплекта оценочных средств

№	Раздел дисциплины (тема)	Код контролируемой компетенции (или ее части)	Технология формирования	Оценочные средства		Описание шкал оценивания
				наименование	№ заданий	
1.	<b>Программирование симметричных многопроцессорных систем.</b> Использование директив компилятора для распараллеливанию (5 семестр)	ОПК-5	Лабораторная работа, СРС	тесты	1-11	согласно табл. 7.1
2.	<b>Программирование кластерных</b>	ПК-3	Лабораторная работа, СРС	контрольные вопросы	1-5	согласно табл. 6.4,

	<b>систем. Обзор инструментария MPI (6 семестр)</b>					табл. 7.1
3.	<b>Программирование систем с массовым параллелизмом. Обзор средств технологий CUDA, STREAM, OpenCL (7 семестр)</b>	ПСК-2	Лабораторная работа, СРС	контрольные вопросы	1-9	согласно табл. 6.4, табл. 7.1

**Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования компетенций:**

- Положение П02.016-2015 «О балльно-рейтинговой системе оценки качества освоения образовательных программ»;

- Список методических указаний, используемых в образовательном процессе, представлен в п.8.2;

Оценочные средства представлены в учебно-методическом комплексе дисциплины.

**6.3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования компетенций в процессе освоения образовательной программы:**

1. Инструментарий CUDA применяется для

\* программирования под кластеры и суперкомпьютеры

\* программирования под GPU +

\* программирования под грид-системы

\* программирования векторных расширений

\* многопоточного программирования

2. Какой из типов оптимизации кода программы дает максимальный выигрыш?

\* высокоуровневая

\* алгоритмическая +

\* микроархитектурная

3. Вынос инвариантов цикла является примером оптимизации следующего типа

\* высокоуровневая +

\* алгоритмическая

\* микроархитектурная

4. Раскрутка циклов является примером оптимизации следующего типа

\* высокоуровневая +

\* алгоритмическая

\* микроархитектурная

5. Изменение алгоритмов работы программы является примером оптимизации следующего типа

\* высокоуровневая

\* алгоритмическая +

\* микроархитектурная

6. Приспособление программы к особенностям конкретного процессора является примером оптимизации следующего типа

- \* высокоуровневая
- \* алгоритмическая
- \* микроархитектурная +

7. Согласно закону Амдала выигрыш от распараллеливания программы ограничивается

- \* ее последовательной частью +
- \* числом процессоров
- \* числом потоков
- \* особенностями микроархитектуры процессора

8. Какая из перечисленных технологий не относится к векторным расширениям системы команд процессора?

- \* MMX
- \* SSE
- \* AVX
- \* AES +

9. Код с плавающей точкой какого компилятора характеризуется наихудшим качеством оптимизации?

- \* Borland C++ Builder +
- \* Intel C++ Compiler
- \* Microsoft Visual C++
- \* GCC

10. Какие из перечисленных оптимизаций не может выполнять компилятор?

- \* раскрутка циклов
- \* ликвидация хвостовой рекурсии
- \* удаление константных параметров подпрограмм
- \* распараллеливание кода под MPI +
- \* векторизация кода под AVX

## 7. Рейтинговый контроль изучения дисциплины

Таблица 7.1 контроль изучения дисциплины

Форма контроля	Минимальный балл		Максимальный балл	
	балл	примечание	балл	примечание
1	2	3	4	5
Многопоточное программирование с использованием WinAPI. Обзор инструментария WinAPI-функций и его использование для организации параллельных вычислений	4	Выполнил, но не защитил	7	Выполнил и защитил
Программирование симметричных многопроцессорных систем.	4	Выполнил, но не защитил	7	Выполнил и защитил



Использование директив компилятора для подсказок к распараллеливанию				
Программирование кластерных систем. Обзор инструментария MPI	4	Выполнил, но не защитил	7	Выполнил и защитил
Программирование систем с массовым параллелизмом. Обзор средств технологий CUDA, STREAM, OpenCL	4	Выполнил, но не защитил	7	Выполнил и защитил
Творческая компонента	0	Не участвовал	15	За участие в конференциях, публикации, задания повышенной сложности индивидуальные научные исследования и т.д.
Зачет	0	Не ответил ни на один вопрос	36	Ответил на все вопросы
Итого	16		100	

## 8. Учебно-методическое и информационное обеспечение учебной дисциплины

### 8.1 Основная литература и дополнительная литература.

#### а) Основная литература

1. Сальников А.М., Ярошенко Е.А., Гребенник О.С., Спиридонов С.В. Введение в параллельные вычисления. Основы программирования на языке СИ с использованием интерфейса MPI. - М.: ИПУ РАН, 2009 - 123 с. <http://window.edu.ru/resource/338/66338>
2. Гергель В.П. Высокопроизводительные вычисления для многоядерных многопроцессорных систем: Учебное пособие. - Нижний Новгород; Изд-во ННГУ им. Н.И. Лобачевского, 2010. - 420 с. <http://window.edu.ru/resource/858/79858>

#### б) Дополнительная литература

3. Антонов А.С. Параллельное программирование с использованием технологии OpenMP. М.: Изд-во МГУ, 2009. 76 с. (электронный ресурс <http://window.edu.ru/resource/291/61291>)
4. Антонов А.С. Параллельное программирование с использованием технологии MPI. М.: Изд-во МГУ, 2004. 71 с. (электронный ресурс)
5. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. СПб.: БХВ-Петербург, 2002. 608 с.

### 8.2 Перечень методических указаний

Разработка программ с поддержкой технологии CUDA с использованием компилятора командной строки / Ватутин Э.И., ЮЗГУ, 2016.

Определение параметров видеокарты с поддержкой технологии CUDA в среде Microsoft Visual Studio / Ватутин Э.И., ЮЗГУ, 2016.

Измерение пропускной способности памяти видеокарт с поддержкой технологии CUDA / Ватутин Э.И., ЮЗГУ, 2016.

Оптимизация работы с глобальной памятью для видеокарт с поддержкой технологии CUDA / Ватутин Э.И., ЮЗГУ, 2016.

### **8.3 Перечень ресурсов информационно-телекоммуникационной сети Интернет.**

Обязательные Интернет ресурсы отсутствуют.

### **8.4. Перечень информационных технологий.**

1. Среды разработки ПО RAD Studio, Visual Studio.
2. Операционная система Windows XP.

### **8.5 Другие учебно-методические материалы.**

Отсутствуют.

## **9. Материально-техническое обеспечение учебной дисциплины.**

Стандартно оборудованные лекционные аудитории. Для проведения отдельных занятий (по заявке) - выделение компьютерного класса, а также аудитории для проведения интерактивных лекций: видеопроектор, экран настенный, др. оборудование.

Рабочие места студентов должны быть оснащены оборудованием не ниже: Pentium III-800/ОЗУ-256 Мб / Video с поддержкой GPGPU / Sound card – 16bit /Headphones / HDD 80 Гб / CD-ROM – 48x / Network adapter – 10/100/ Мбс / SVGA – 19”.

