



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.В.01.02 Высокопроизводительные вычисления и распределенные
вычислительные системы

Направление подготовки: 01.04.02 Прикладная математика и информатика

Направленность (профиль): Информационное обеспечение экономической деятельности

Квалификация (степень): магистр

Форма обучения: очная

Институт: математики, естествознания и техники

Кафедра: математического моделирования и компьютерных технологий

	очная форма	очно-заочная форма	заочная форма
Курс	2		
Семестр/триместр	3, 4		

Лекции	21		
Лабораторные занятия			
Практические (семинарские) занятия	21		
Консультации			
Форма(ы) промежуточной аттестации	Экзамен - 0,3		
Контроль	9		
Иные формы работы			
Самостоятельная работа	92,7		

Всего часов: 144

Трудоемкость: 4 зачетных единиц.

I. ОРГАНИЗАЦИОННО-МЕТОДИЧЕСКИЙ РАЗДЕЛ

Цель изучения дисциплины: является научная и практическая подготовка магистров по направлениям применения дискретных и вероятностных математических моделей в технических и экономических системах.

Задачи изучения дисциплины:

- сформировать навыки построения моделей с использованием параллельного программирования;
- знать параллельные вычисления для задач вычислительной математики (матричные вычисления, решение систем линейных уравнений, сортировка, обработка графов, уравнения в частных производных, многоэкстремальная оптимизация);
- научить использовать в своей практической деятельности математические методы и модели;
- привить умение самостоятельно изучать и использовать литературу по математическому моделированию.

Место дисциплины в структуре ОПОП: реализуется в рамках части, формируемой участниками образовательных отношений.

Планируемые результаты обучения по дисциплине, соотнесенные с индикаторами достижения компетенций:

Код компетенции	Индикаторы достижения компетенции	Планируемые результаты обучения по дисциплине
УК-3	Знать: - правила командной работы; - необходимые условия для эффективной командной работы.	Знает: - основные подходы и необходимые условия для разработки параллельных программ.
	Уметь: - планировать командную работу, распределять поручения и делегировать полномочия членам команды; - организовывать обсуждение разных идей и мнений; - предвидеть результаты (последствия) как личных, так и коллективных действий.	Умеет: - предвидеть основные результаты, параметры получаемых параллельных программ, таких как ускорение, эффективность и масштабируемость.
	Владеть: - организацией и управлением командным взаимодействием	Владеет: - организацией и управлением командным взаимодействием в решении поставленных

	<p>действием в решении поставленных целей;</p> <ul style="list-style-type: none"> - навыками создания команды для выполнения практических задач; - навыками разработки стратегии командной работы; - навыками преодоления возникающих в команде разногласий, споров и конфликтов на основе учета интересов всех сторон. 	<p>целей при разработки параллельных программ для многоядерных систем.</p>
ПКС-1	<p>Знать:</p> <ul style="list-style-type: none"> – методы и приемы организации процесса разработки баз данных ИС; – методологии разработки баз данных ИС и технологии программирования; – языки формализации функциональных спецификаций; – методы и приемы формализации задач; – методы и средства проектирования программного обеспечения, программных интерфейсов и баз данных. 	<p>Знает:</p> <ul style="list-style-type: none"> - методы параллельных вычислений для задач вычислительной математики (матричные вычисления, решение систем линейных уравнений, сортировка, обработка графов, уравнения в частных производных, многоэкстремальная оптимизация).
	<p>Уметь:</p> <ul style="list-style-type: none"> – использовать существующие типовые решения проектирования баз данных ИС; – применять методы и средства проектирования структур данных, баз данных, программных интерфейсов; – осуществлять коммуникации с заинтересованными сторонами; – выбирать средства реализации требований к базам данных ИС; – вырабатывать варианты реализации баз данных ИС и требований к нему; – проводить анализ исполнения требований. 	<p>Умеет:</p> <ul style="list-style-type: none"> - строить модель выполнения параллельных программ; - оценивать эффективности параллельных вычислений; - анализировать сложность вычислений и возможность распараллеливания разрабатываемых алгоритмов; - применять общие схемы разработки параллельных программ для реализаций собственных алгоритмов.
	<p>Владеть:</p>	<p>Владеет:</p>

	<ul style="list-style-type: none"> – методологией организации процесса разработки баз данных ИС; – методологией и технологиями проектирования программных интерфейсов, структур и баз данных ИС в соответствии с установленными требованиями; – действиями по разработке и согласованию технических спецификаций на программные компоненты; – действиями по согласованию требований к базам данных ИС с заинтересованными сторонами, распределению заданий между программистами в соответствии с техническими спецификациями, осуществлению контроля выполнения заданий, формированию отчетности в соответствии с установленными регламентами; 	<ul style="list-style-type: none"> - основами разработки параллельных программ для многоядерных систем с применением технологии OpenMP.
--	--	--

II. СОДЕРЖАНИЕ И ОБЪЕМ ДИСЦИПЛИНЫ

с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу

Очная форма обучения

№ п/п	Наименование разделов и тем	Всего	Аудиторные занятия			Сам. раб.
			ЛК	ПЗ	ЛБ	
	Раздел 1. Цели и задачи введения параллельной обработки данных	16	2	2		12
1.	Тема 1. Методы и средства параллельной обработки информации	16	2	2		12
	Раздел 2. Принципы построения параллельных вычислительных систем.	16	2	2		12
2.	Тема 2. Обзор современных параллельных вычислительных систем.	16	2	2		12

	Раздел 3 Модели вычислений и методы анализа эффективности.	22	4	4		14
3	Тема 3. Показатели эффективности параллельных вычислений: ускорение, эффективность, масштабируемость.	22	4	4		14
	Раздел 4. Основы параллельного программирования для многоядерных систем.	18	2	2		14
4	Тема 4. Основные понятия: взаимодействие и взаимоисключение потоков, синхронизация потоков.	18	2	2		14
	<i>Зачет</i>					
	<i>Итого за 3 семестр</i>	72	10	10		52
	Раздел 5. Технология разработки параллельных программ для многоядерных систем с общей памятью OpenMP.	32,7	6	6		20,7
5	Тема 5. Основы технологии OpenMP.	32,7	6	6		20,7
	Раздел 6. Параллельные численные алгоритмы для решения типовых задач вычислительной математики.	30	5	5		20
6	Тема 6. Матричные вычисления (умножение матриц, транспонирование, решение систем линейных уравнений).	30	5	5		20
	<i>Контроль</i>	9				
	<i>Консультация</i>					
	<i>Экзамен</i>	0,3				
	<i>Итого за 4 семестр</i>	62,7	11		11	40,7
	ИТОГО:	144	21		21	92,7

Очно-заочная форма обучения (не реализуется)

Заочная форма обучения (не реализуется)

III. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕЙ И

ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Текущая аттестация проводится в форме теста.

Типовой вариант теста

№	Задание (вопрос)	Ответ	Балл
1	К какому классу в соответствии с классификацией Флинна относится большинство современных многопроцессорных вычислительных систем?	<input type="checkbox"/> SISD (Single Instruction, Single Data) <input type="checkbox"/> SIMD (Single Instruction, Multiple Data) <input type="checkbox"/> MISD (Multiple Instruction, Single Data) <input checked="" type="checkbox"/> MIMD (Multiple Instruction, Multiple Data stream)	1
2	Укажите все классы многопроцессорных вычислительных систем с общей оперативной памятью.	<input checked="" type="checkbox"/> SMP-системы (симметричные мультипроцессорные системы) <input checked="" type="checkbox"/> NUMA-системы (системы с неоднородным доступом к памяти) <input type="checkbox"/> MPP-системы (процессоры с массовым параллелизмом) <input type="checkbox"/> COW-системы (кластеры рабочих станций)	1
3	Укажите все классы многопроцессорных вычислительных систем с распределенной оперативной памятью.	<input type="checkbox"/> SMP-системы (симметричные мультипроцессорные системы) <input type="checkbox"/> NUMA-системы (системы с неоднородным доступом к памяти) <input checked="" type="checkbox"/> MPP-системы (процессоры с массовым параллелизмом) <input checked="" type="checkbox"/> COW-системы (кластеры рабочих станций)	1
4	Какие модели параллельного программирования реализованы в стандарте MPI?	<input checked="" type="checkbox"/> SPMD (Single Program Multiple Data) <input checked="" type="checkbox"/> MPMD (Multiple Program Multiple Data) <input checked="" type="checkbox"/> Передача сообщений <input type="checkbox"/> Общая память <input type="checkbox"/> Параллелизм "fork-join"	1
5	Для каких языков программирования штатно предназначен стандарт MPI?	<input checked="" type="checkbox"/> C <input checked="" type="checkbox"/> C++ <input checked="" type="checkbox"/> FORTRAN <input type="checkbox"/> Pascal <input type="checkbox"/> Object Pascal (Delphi) <input type="checkbox"/> Java	1
6	Укажите верные утверждения.	<input checked="" type="checkbox"/> Если среда программирования не настроена для поддержки стандарта MPI, то запуск программы, использующей данный стандарт, невозможен ввиду синтаксической ошибки в программе. <input type="checkbox"/> Если среда программирования не настроена для поддержки стандарта MPI, то запуск программы, использующей данный стандарт, невозможен ввиду ошибки времени выполнения программы. <input type="checkbox"/> Если среда программирования не настроена для поддержки стандарта MPI, то это не влияет на возможность запуска программы, использующей данный стандарт.	1

№	Задание (вопрос)	Ответ	Балл												
7	Укажите порядок, в котором данные конструкции должны быть представлены в MPI-программе.	<table border="1"> <tr> <td>3</td> <td>Функции обменов сообщениями (MPI_Send, MPI_Recv и др.)</td> </tr> <tr> <td>4</td> <td>Функция MPI_Finalize</td> </tr> <tr> <td>1</td> <td>#include "mpi.h"</td> </tr> <tr> <td>2</td> <td>Функция MPI_Init</td> </tr> </table>	3	Функции обменов сообщениями (MPI_Send, MPI_Recv и др.)	4	Функция MPI_Finalize	1	#include "mpi.h"	2	Функция MPI_Init	1				
3	Функции обменов сообщениями (MPI_Send, MPI_Recv и др.)														
4	Функция MPI_Finalize														
1	#include "mpi.h"														
2	Функция MPI_Init														
8	<p>Пусть обмены между MPI-процессами 0 и 1 организованы по следующей схеме:</p> <table border="1"> <thead> <tr> <th>Процесс 0</th> <th>Процесс 1</th> </tr> </thead> <tbody> <tr> <td>1. MPI_Recv от процесса 1</td> <td>1. MPI_Recv от процесса 0</td> </tr> <tr> <td>2. MPI_Send процессу 1</td> <td>2. MPI_Send процессу 0</td> </tr> </tbody> </table> <p>Какое утверждение о данных процессах является верным?</p>	Процесс 0	Процесс 1	1. MPI_Recv от процесса 1	1. MPI_Recv от процесса 0	2. MPI_Send процессу 1	2. MPI_Send процессу 0	<table border="1"> <tr> <td><input checked="" type="checkbox"/></td> <td>Безусловно, возникает тупиковая ситуация</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Безусловно, тупиковая ситуация не возникает</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Возможно возникновение тупиковой ситуации</td> </tr> </table>	<input checked="" type="checkbox"/>	Безусловно, возникает тупиковая ситуация	<input type="checkbox"/>	Безусловно, тупиковая ситуация не возникает	<input type="checkbox"/>	Возможно возникновение тупиковой ситуации	1
Процесс 0	Процесс 1														
1. MPI_Recv от процесса 1	1. MPI_Recv от процесса 0														
2. MPI_Send процессу 1	2. MPI_Send процессу 0														
<input checked="" type="checkbox"/>	Безусловно, возникает тупиковая ситуация														
<input type="checkbox"/>	Безусловно, тупиковая ситуация не возникает														
<input type="checkbox"/>	Возможно возникновение тупиковой ситуации														
9	<p>Пусть обмены между MPI-процессами 0 и 1 организованы по следующей схеме:</p> <table border="1"> <thead> <tr> <th>Процесс 0</th> <th>Процесс 1</th> </tr> </thead> <tbody> <tr> <td>1. MPI_Send процессу 1</td> <td>1. MPI_Send процессу 0</td> </tr> <tr> <td>2. MPI_Recv от процесса 1</td> <td>2. MPI_Recv от процесса 0</td> </tr> </tbody> </table> <p>Какое утверждение о данных процессах является верным?</p>	Процесс 0	Процесс 1	1. MPI_Send процессу 1	1. MPI_Send процессу 0	2. MPI_Recv от процесса 1	2. MPI_Recv от процесса 0	<table border="1"> <tr> <td><input type="checkbox"/></td> <td>Безусловно, возникает тупиковая ситуация</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Безусловно, тупиковая ситуация не возникает</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>Возможно возникновение тупиковой ситуации</td> </tr> </table>	<input type="checkbox"/>	Безусловно, возникает тупиковая ситуация	<input type="checkbox"/>	Безусловно, тупиковая ситуация не возникает	<input checked="" type="checkbox"/>	Возможно возникновение тупиковой ситуации	1
Процесс 0	Процесс 1														
1. MPI_Send процессу 1	1. MPI_Send процессу 0														
2. MPI_Recv от процесса 1	2. MPI_Recv от процесса 0														
<input type="checkbox"/>	Безусловно, возникает тупиковая ситуация														
<input type="checkbox"/>	Безусловно, тупиковая ситуация не возникает														
<input checked="" type="checkbox"/>	Возможно возникновение тупиковой ситуации														
10	<p>Пусть обмены между MPI-процессами 0 и 1 организованы по следующей схеме:</p> <table border="1"> <thead> <tr> <th>Процесс 0</th> <th>Процесс 1</th> </tr> </thead> <tbody> <tr> <td>1. MPI_Send процессу 1</td> <td>1. MPI_Recv от процесса 0</td> </tr> <tr> <td>2. MPI_Recv от процесса 1</td> <td>2. MPI_Send процессу 0</td> </tr> </tbody> </table> <p>Какое утверждение о данных процессах является верным?</p>	Процесс 0	Процесс 1	1. MPI_Send процессу 1	1. MPI_Recv от процесса 0	2. MPI_Recv от процесса 1	2. MPI_Send процессу 0	<table border="1"> <tr> <td><input type="checkbox"/></td> <td>Безусловно, возникает тупиковая ситуация</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>Безусловно, тупиковая ситуация не возникает</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Возможно возникновение тупиковой ситуации</td> </tr> </table>	<input type="checkbox"/>	Безусловно, возникает тупиковая ситуация	<input checked="" type="checkbox"/>	Безусловно, тупиковая ситуация не возникает	<input type="checkbox"/>	Возможно возникновение тупиковой ситуации	1
Процесс 0	Процесс 1														
1. MPI_Send процессу 1	1. MPI_Recv от процесса 0														
2. MPI_Recv от процесса 1	2. MPI_Send процессу 0														
<input type="checkbox"/>	Безусловно, возникает тупиковая ситуация														
<input checked="" type="checkbox"/>	Безусловно, тупиковая ситуация не возникает														
<input type="checkbox"/>	Возможно возникновение тупиковой ситуации														
11	<p>Пусть обмены между MPI-процессами 0 и 1 организованы по следующей схеме:</p> <table border="1"> <thead> <tr> <th>Процесс 0</th> <th>Процесс 1</th> </tr> </thead> <tbody> <tr> <td>1. MPI_Irecv от процесса 1</td> <td>1. MPI_Irecv от процесса 0</td> </tr> <tr> <td>2. MPI_Isend процессу 1</td> <td>2. MPI_Isend процессу 0</td> </tr> </tbody> </table> <p>Какое утверждение о данных процессах является верным?</p>	Процесс 0	Процесс 1	1. MPI_Irecv от процесса 1	1. MPI_Irecv от процесса 0	2. MPI_Isend процессу 1	2. MPI_Isend процессу 0	<table border="1"> <tr> <td><input type="checkbox"/></td> <td>Безусловно, возникает тупиковая ситуация</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>Безусловно, тупиковая ситуация не возникает</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Возможно возникновение тупиковой ситуации</td> </tr> </table>	<input type="checkbox"/>	Безусловно, возникает тупиковая ситуация	<input checked="" type="checkbox"/>	Безусловно, тупиковая ситуация не возникает	<input type="checkbox"/>	Возможно возникновение тупиковой ситуации	1
Процесс 0	Процесс 1														
1. MPI_Irecv от процесса 1	1. MPI_Irecv от процесса 0														
2. MPI_Isend процессу 1	2. MPI_Isend процессу 0														
<input type="checkbox"/>	Безусловно, возникает тупиковая ситуация														
<input checked="" type="checkbox"/>	Безусловно, тупиковая ситуация не возникает														
<input type="checkbox"/>	Возможно возникновение тупиковой ситуации														

№	Задание (вопрос)	Ответ	Балл
12	Какие функции стандарта MPI используются для обменов сообщениями между процессами и <i>приостанавливают</i> выполнение процесса до завершения операции обмена?	<input checked="" type="checkbox"/> MPI_Send <input checked="" type="checkbox"/> MPI_Ssend <input checked="" type="checkbox"/> MPI_Bsend <input checked="" type="checkbox"/> MPI_Rsend <input checked="" type="checkbox"/> MPI_Recv <input type="checkbox"/> MPI_Isend <input type="checkbox"/> MPI_Issend <input type="checkbox"/> MPI_Ibsend <input type="checkbox"/> MPI_Irsend <input type="checkbox"/> MPI_Irecv	1
13	Какие функции стандарта MPI используются для обменов сообщениями между процессами и <i>не приостанавливают</i> выполнение процесса до завершения операции обмена?	<input type="checkbox"/> MPI_Send <input type="checkbox"/> MPI_Ssend <input type="checkbox"/> MPI_Bsend <input type="checkbox"/> MPI_Rsend <input type="checkbox"/> MPI_Recv <input checked="" type="checkbox"/> MPI_Isend <input checked="" type="checkbox"/> MPI_Issend <input checked="" type="checkbox"/> MPI_Ibsend <input checked="" type="checkbox"/> MPI_Irsend <input checked="" type="checkbox"/> MPI_Irecv	1
14	Какие модели параллельного программирования реализованы в стандарте OpenMP?	<input type="checkbox"/> SPMD (Single Program Multiple Data) <input type="checkbox"/> MPMD (Multiple Program Multiple Data) <input type="checkbox"/> Передача сообщений <input checked="" type="checkbox"/> Общая память <input checked="" type="checkbox"/> Параллелизм "fork-join"	1
15	Для каких языков программирования штатно предназначен стандарт OpenMP?	<input checked="" type="checkbox"/> C <input checked="" type="checkbox"/> C++ <input checked="" type="checkbox"/> FORTRAN <input type="checkbox"/> Pascal <input type="checkbox"/> Object Pascal (Delphi) <input type="checkbox"/> Java	1
16	Укажите верные утверждения.	<input type="checkbox"/> Если среда программирования не настроена для поддержки стандарта OpenMP, то запуск программы, использующей данный стандарт, невозможен ввиду синтаксической ошибки в программе. <input type="checkbox"/> Если среда программирования не настроена для поддержки стандарта OpenMP, то запуск программы, использующей данный стандарт, невозможен ввиду ошибки времени выполнения программы. <input checked="" type="checkbox"/> Если среда программирования не настроена для поддержки стандарта OpenMP, то это не влияет на возможность запуска программы, использующей данный стандарт.	1

№	Задание (вопрос)	Ответ	Балл
17	<p>Дан фрагмент OpenMP-программы:</p> <pre>void main() { int a,b,c,d; #pragma omp parallel private (c) shared (d,e) { int e,f; ... } ... }</pre> <p>Укажите все переменные, которые являются общими.</p>	<input type="checkbox"/> a, b, c, d <input type="checkbox"/> e, f <input checked="" type="checkbox"/> a, b, d, e <input type="checkbox"/> a, b, d <input type="checkbox"/> c, e, f <input type="checkbox"/> c, f <input type="checkbox"/> c, d <input type="checkbox"/> Общие переменные отсутствуют	1
18	<p>Дан фрагмент OpenMP-программы:</p> <pre>void main() { int a,b,c,d; #pragma omp parallel private (c) shared (d,e) { int e,f; ... } ... }</pre> <p>Укажите все переменные, которые являются частными.</p>	<input type="checkbox"/> a, b, c, d <input type="checkbox"/> e, f <input type="checkbox"/> a, b, d, e <input type="checkbox"/> a, b, d <input type="checkbox"/> c, e, f <input checked="" type="checkbox"/> c, f <input type="checkbox"/> c, d <input type="checkbox"/> Частные переменные отсутствуют	1
19	<p>Дана OpenMP-программа:</p> <pre>#include "omp.h" #include <stdio.h> void main() { int r; #pragma omp parallel { r=omp_get_thread_num(); printf("%d\n", r); } }</pre> <p>Что будет выдано на экран в результате работы данной программы?</p>	<input type="checkbox"/> Числа 0, 1, ..., N-1 без повторов, где N – количество создаваемых нитей. <input type="checkbox"/> Одно число из диапазона 0, 1, ..., N-1, где N – количество создаваемых нитей. <input checked="" type="checkbox"/> Числа (возможно, повторяющиеся) из диапазона 0, 1, ..., N-1, где N – количество создаваемых нитей.	1
20	<p>Дана OpenMP-программа:</p> <pre>#include "omp.h" #include <stdio.h> void main() { int r; #pragma omp parallel private(r) { r=omp_get_thread_num(); printf("%d\n", r); } }</pre> <p>Что будет выдано на экран в результате работы данной программы?</p>	<input checked="" type="checkbox"/> Числа 0, 1, ..., N-1 без повторов, где N – количество создаваемых нитей. <input type="checkbox"/> Одно число из диапазона 0, 1, ..., N-1, где N – количество создаваемых нитей. <input type="checkbox"/> Числа (возможно, повторяющиеся) из диапазона 0, 1, ..., N-1, где N – количество создаваемых нитей.	1

№	Задание (вопрос)	Ответ	Балл
21	<p>Дан фрагмент OpenMP-программы:</p> <pre>#pragma omp директива { операторы }</pre> <p>Укажите все директивы, создающие параллельный регион.</p>	<input type="checkbox"/> barrier <input type="checkbox"/> master <input checked="" type="checkbox"/> parallel <input type="checkbox"/> single <input checked="" type="checkbox"/> parallel sections <input type="checkbox"/> critical	1
22	<p>Дан фрагмент OpenMP-программы:</p> <pre>#pragma omp директива { операторы }</pre> <p>Укажите все директивы, в соответствии с которыми операторы параллельного региона не будут выполняться одновременно двумя или более нитями.</p>	<input type="checkbox"/> barrier <input type="checkbox"/> master <input checked="" type="checkbox"/> critical <input type="checkbox"/> single	1
23	<p>Дан фрагмент OpenMP-программы:</p> <pre>#pragma omp директива { операторы }</pre> <p>Укажите все директивы, в соответствии с которыми операторы параллельного региона будут выполняться только одной нитью.</p>	<input type="checkbox"/> barrier <input checked="" type="checkbox"/> master <input type="checkbox"/> critical <input checked="" type="checkbox"/> single	1
Итого			23

Количество баллов	Оценка
18-23	отлично
15-17	хорошо
12-14	удовлетворительно
11 и менее	неудовлетворительно

Шкала
Оценка _____

80%

65% Подпись экзаменатора

50%

Промежуточная аттестация обучающихся осуществляется в форме зачета, экзамена с использованием следующих оценочных материалов: перечень вопросов к зачету, экзамену.

Вопросы к зачету (3 семестр, очная форма обучения)

1. Отображение с минимальным временем ожидания: план с минимальным временем ожидания, задача целочисленного программирования для определения минимального времени ожидания при фиксированном числе процессорных элементов, план с минимальным временем ожидания при фиксированном назначении узлов.
2. Свойства: синхронность, модульность и регулярность, пространственная и временная локальность, конвейеризуемость.

3. Требования к систолическим структурам: простота и регулярность схемы, параллелизм и локальность связей, сбалансированность вычислений с вводом-выводом, схема разводки синхросигналов.
4. Непосредственное отображение ГЗ в систолическую схему. Процедура систолизации сечением.
5. Процедура ресинхронизации сечением: масштабирование времени, перемещение задержки, процедура систолизации.
6. Критерии оптимальности, структуры с минимальным временем вычислений, схемы с минимальным конвейерным тактом.
7. Оптимизация на этапах разработки ГЗ, ГПС и систолизации.
8. Процедура минимизации числа элементов задержки. Многоскоростная систолическая схема.
9. Повышение эффективности использования процессорных элементов: конвейер с чередованием, разделение процессора, требование одинаковых времен
10. Понятие вычислительных волновых фронтов.
11. Отображение ГЗ на волновые процессоры: граф потока данных (ГПД), отображение ГЗ на ГПД, получение ГПД из ГПС.
12. Временной анализ и оптимальное распределение очередей.
13. Оптимизация производительности ГПД: оптимальный конвейерный такт, задача минимального распределения очередей.
14. Временной анализ регулярных волновых процессоров.
15. Базовая архитектура и концепция. Внутренняя архитектура транспьютеров.
16. Тенденции развития методов параллельных вычислений и параллельного программирования.

Вопросы к экзамену (4 семестр, очная форма обучения)

1. Важность проблематики параллельных вычислений.
2. Обзор современных параллельных вычислительных систем.
3. Классификация и оценка производительности.
4. Понятие многоядерных и многопроцессорных вычислительных систем с общей и распределенной памятью.
5. Показатели эффективности параллельных вычислений: ускорение, эффективность, масштабируемость.
6. Модель вычислений в виде графа "операции-операнды".
7. Анализ модели: определение времени выполнения параллельного метода, оценка максимально достижимого распараллеливания.
8. Анализ модели: выбор вариантов распределения вычислительной нагрузки.
9. Методы оценки масштабируемости параллельных алгоритмов.
10. Агрегация модели вычислений.
11. Взаимодействие и взаимоисключение потоков, синхронизация потоков.
12. Проблемы взаимоблокировки потоков.
13. Примеры классических задач синхронизации: "Производители-Потребители", "Читатели-Писатели", "Обедающие философы" и "Спящий парикмахер".

14. Обзор методов повышения эффективности параллельных программ.
15. Основы технологии OpenMP.
16. Распределение вычислительной нагрузки между потоками (распараллеливание по данным для циклов).
17. Управление данными для параллельно-выполняемых потоков.
18. Организация взаимного исключения при использовании общих переменных.
19. Распределение вычислительной нагрузки между потоками.
20. Расширенные возможности OpenMP (обзор библиотечных функций).

IV. ПЕРЕЧЕНЬ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1. Кареева, Е.Д. Основы многопоточного и параллельного программирования : учебное пособие / Е.Д. Кареева ; Сибирский федеральный университет, Институт вычислительного моделирования Сибирского отделения Российской академии наук, Сибирский научно-образовательный центр суперкомпьютерных технологий. – Красноярск : Сибирский федеральный университет (СФУ), 2016. – 355 с. : ил. – URL: <http://biblioclub.ru/index.php?page=book&id=497217> .
2. Николаев, Е. И. Параллельные вычисления : учебное пособие / Е. И. Николаев ; Северо-Кавказский федеральный университет. – Ставрополь : Северо-Кавказский Федеральный университет (СКФУ), 2016. – 185 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=459124>
2. Афанасьев, К.Е. Основы высокопроизводительных вычислений : учебное пособие : [16+] / К.Е. Афанасьев, И.В. Григорьева, Т.С. Рейн. – Кемерово : Кемеровский государственный университет, 2012. – Т. 3. Параллельные вычислительные алгоритмы. – 185 с. – URL: <http://biblioclub.ru/index.php?page=book&id=232205>.
3. Основы высокопроизводительных вычислений : учебное пособие / К.Е. Афанасьев, С.В. Стуколов, В.В. Малышенко и др. – Кемерово : Кемеровский государственный университет, 2012. – Т. 2. Технологии параллельного программирования. – 412 с. – URL: <http://biblioclub.ru/index.php?page=book&id=232204>.

V. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

№ пп	Ссылка на информационный ресурс	Наименование разработки в электронной форме	Доступность
1.	https://infourok.ru/	Инфоурок: образовательный интернет-проект России. Включает: конспекты уроков, презентации, тесты, видеоуроки и другие материалы по предметам школьной программы.	Свободный доступ

2.	http://edu.ru/	Российское образование: Федеральный портал. Включает ссылки на порталы и сайты образовательных учреждений; государственные образовательные стандарты; нормативные документы; каталог экскурсий и обучающих программ.	Свободный доступ
----	---	---	------------------

VI. СОВРЕМЕННЫЕ ПРОФЕССИОНАЛЬНЫЕ БАЗЫ ДАННЫХ И ИНФОРМАЦИОННЫЕ СПРАВОЧНЫЕ СИСТЕМЫ

1.	http://www.biblioclub.ru	Электронно-библиотечная система (ЭБС) Университетская библиотека онлайн	Регистрация через любой университетский компьютер. В дальнейшем предоставляется неограниченный индивидуальный доступ из любой точки, в которой имеется доступ к сети Интернет
2.	www.garant.ru	Информационно-правовой портал	Свободный доступ
3.	www.elibrary.ru	Российский информационный портал в области науки, технологии, медицины и образования	Свободный доступ
4.	www.consultant.ru	Российская компьютерная справочно-правовая система	Свободный доступ

VII. ЛИЦЕНЗИОННОЕ И СВОБОДНО РАСПРОСТРАНЯЕМОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

При реализации учебной дисциплины применяется следующее лицензионное и свободно распространяемое программное обеспечение:

- Microsoft Windows;
- Microsoft Office;
- LibreOffice и др.

VIII. ОБОРУДОВАНИЕ И ТЕХНИЧЕСКИЕ СРЕДСТВА ОБУЧЕНИЯ, НЕОБХОДИМЫЕ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Учебные занятия проводятся в аудиториях, укомплектованных специализированной мебелью, в том числе стационарными или переносными техническими средствами обучения (проектор, экран, компьютер/ноутбук).

Самостоятельная работа проводится в кабинетах, оснащенных компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

