

ГОУ ВПО «ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ»

ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Кафедра теории упругости и вычислительной математики

имени академика А.С. Космодамианского



УТВЕРЖДАЮ:

проректор по научно-методической
и учебной работе

Е.И. Скафа

22 апреля 2020 г.

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
«ПАРАЛЛЕЛЬНОЕ ПРОГРАММИРОВАНИЕ»**

Направление подготовки:	01.04.02 Прикладная математика и информатика
Магистерская программа:	Прикладная математика и информатика
Образовательная программа:	академическая магистратура
Квалификация:	магистр
Форма обучения:	<u>очная</u> , очно-заочная, заочная нужное подчеркнуть

Донецк 2020

УТВЕРЖДАЮ:

Декан факультета математики
и информационных технологий

И. А. Моисеенко

«16» апреля 2020 г.

МП



Программа составлена на основании Федерального государственного образовательного стандарта высшего образования (ФГОС ВО) направления подготовки 01.04.02 Прикладная математика и информатика, утвержденного приказом Министерства образования и науки Российской Федерации от «12» марта 2015 г. № 228; Порядка организации учебного процесса в образовательных организациях высшего профессионального образования Донецкой Народной Республики, утвержденного приказом Министерства образования и науки ДНР № 1171 от «10» ноября 2017 г.; учебного плана и основной образовательной программы Прикладная математика и информатика, направления подготовки 01.04.02 Прикладная математика и информатика, разработанных в ГОУ ВПО «Донецкий национальный университет».

Разработчик:

Доцент кафедры теории упругости и
вычислительной математики имени
академика А.С. Космодамианского

Р.Н. Нескородев

Программа учебной дисциплины утверждена на заседании кафедры теории упругости и вычислительной математики имени академика А.С. Космодамианского

Протокол № 11 от «9» апреля 2020 г.
Заведующий кафедрой

В.И. Сторожев

Программа учебной дисциплины одобрена учебно-методической комиссией факультета математики и информационных технологий

Протокол № 8 от «15» апреля 2020 г.

Председатель учебно-методической
комиссии факультета

Л.И. Селякова

1. ОБЛАСТЬ ПРИМЕНЕНИЯ И МЕСТО ДИСЦИПЛИНЫ В УЧЕБНОМ ПРОЦЕССЕ

Курс «Параллельное программирование» является дисциплиной вариативной части Блока 1 «Дисциплины (модули)» по направлению подготовки 01.04.02 Прикладная математика и информатика (магистерская программа: Прикладная математика и информатика).

Дисциплина реализуется на факультете математики и информационных технологий кафедрой теории упругости и вычислительной математики.

Этот курс, опираясь на математическую (математический анализ, алгебра, дискретная математика, численные методы и др.) и компьютерную (архитектура ЭВМ, языки и методы программирования, операционные системы) подготовку студентов, закладывает фундамент научно-методической подготовки будущих исследователей в области теории упругости и вычислительной математики.

Полученные знания могут быть использованы студентами во время выполнения научно-исследовательской работы при написании магистерской диссертации.

2. СТРУКТУРА ДИСЦИПЛИНЫ

<i>Характеристика учебной дисциплины</i>		
Направление подготовки	01.04.02 Прикладная математика и информатика	
Магистерская программа	Прикладная математика и информатика	
Образовательная программа	академическая магистратура	
Квалификация	магистр	
Количество содержательных модулей	1	
Дисциплина базовой / вариативной части образовательной программы	Дисциплина вариативной части	
Формы контроля (МК, экзамен, зачет)		
Показатели	очная форма обучения	заочная форма обучения
Количество зачетных единиц (кредитов)	3	
Год подготовки	2	
Семестр	3	
Количество часов	108	
- лекционных	18	
- практических, семинарских	-	
- лабораторных	36	
- самостоятельной работы	54	
в т.ч. индивидуальное задание	24	
Недельное количество часов,	6	
в т.ч. аудиторных	3	

3. ОПИСАНИЕ ДИСЦИПЛИНЫ

Цели и задачи

Целью изучения дисциплины «Параллельное программирование» является формирование у студентов практических навыков организации параллельных алгоритмов для многопроцессорных вычислительных систем, в приобретении студентами теоретических и практических знаний в области параллельного программирования.

Основными задачами изучения дисциплины являются:

- изучение общих положений и принципов программирования параллельных систем;
- изучение технологий и средств параллельного программирования средствами операционной системы и прикладного окружения;
- изучение методов программирования параллельных приближенных вычислений средствами технологии OpenMP.

Требования к результатам освоения дисциплины. Процесс изучения дисциплины “Параллельное программирование” направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО РФ по направлению подготовки 01.04.02 Прикладная математика и информатика и основной образовательной программы высшего профессионального образования направления подготовки 01.04.02 Прикладная математика и информатика (магистерская программа: прикладная математика и информатика):

а) общекультурных (ОК):

- способностью к абстрактному мышлению, анализу, синтезу (ОК-1);
- готовностью к саморазвитию, самореализации, использованию творческого потенциала (ОК-3);

б) общепрофессиональных (ОПК):

- способность самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности, расширять и углублять свое научное мировоззрение (ОПК-3);
- способностью использовать и применять углубленные знания в области прикладной математики и информатики (ОПК-4);

в) профессиональных (ПК):

научно-исследовательская деятельность:

- способность разрабатывать и применять математические методы, системное и прикладное программное обеспечение для решения задач научной и проектно-технологической деятельности (ПК-3);
- способность разрабатывать и анализировать концептуальные и теоретические модели решаемых задач проектной и производственно-технологической деятельности (ПК-4);

В результате изучения учебной дисциплины студент должен:

знать:

- принципы построения параллельных вычислительных систем;
- основы моделирования и анализа параллельных вычислений;
- основные операции передачи данных между процессами и производные типы данных;
- гибридные вычислительные системы с использованием различных технологий;
- основных команд библиотеки OpenMP для практической реализации на ПК параллельных алгоритмов.

уметь:

- разрабатывать параллельные алгоритмы и программы для решения разного класса задач на компьютерах с общей памятью;
- использовать библиотеки для параллельных программ;
- оценивать трудоемкость параллельных алгоритмов;
- ставить и решать задачи, возникающие в процессе конструирования параллельных программ.

владеть:

- теоретическими знаниями в области организации взаимодействующих процессов;
- навыками параллельного программирования на основе языка высокого уровня;
- средствами параллельного программирования OpenMP.

4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ И ФОРМЫ ОРГАНИЗАЦИИ УЧЕБНОГО ПРОЦЕССА

Порядковый номер и тема	Краткое содержание темы
Содержательный модуль 1	
Тема 1. <i>Принципы построения параллельных вычислительных систем.</i>	Параллельные компьютеры и суперЭВМ. Классификация параллельных вычислительных систем. Типы параллелизма. Параллелизм на уровне команд. Конвейер. Суперскалярность. SIMD команды. Параллелизм на уровне задач.
Тема 2. <i>Анализ вычислительной сложности параллельных алгоритмов</i>	Аналитические методы определения вычислительной сложности. Оценка вычислительной сложности для последовательных алгоритмов. Показатели для оценки параллельных алгоритмов: ускорение, эффективность и стоимость параллельных алгоритмов. Экспериментальные методы определения вычислительной сложности.
Тема 3. <i>Разработка параллельных программ</i>	Декомпозиция по функциям и по данным. Планирование параллельных программ. Зависимости данных. Способы синхронизации. Определение стоимости программы с учетом параллельного выполнения. Масштабируемость.
Тема 4. <i>Параллельное программирование при помощи технологии OpenMP</i>	Параллельные и последовательные области. Директива parallel. Распараллеливание оператора цикла. Использование параллельных секций. Синхронизация потоков. Параллельные методы матричного умножения. Вычисление определенного интеграла в многопроцессорной системе. Параллельные алгоритмы сортировки данных. Задача поиска кратчайших путей в графе.

Тематический план

Содержательный модуль 1												
Названия содержательных модулей и тем	Количество часов											
	Очная форма обучения						Заочная форма обучения					
	всего	в т.ч.					всего	в т.ч.				
		лекции	практические	лабораторные	самостоятельная работа	индивидуальная работа		лекции	практические	лабораторные	самостоятельная работа	индивидуальная работа
Тема 1. Принципы построения параллельных вычислительных систем.	24	4	-	6	8	6						
Тема 2. Анализ вычислительной сложности параллельных алгоритмов	25	6	-	6	7	6						
Тема 3. Разработка параллельных программ	30	6	-	12	6	6						
Тема 4. Параллельное программирование при помощи технологии OpenMP	29	2	-	12	9	6						
Итого по содержательному модулю 1	108	18	-	36	30	24						

5. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ПРОВЕДЕНИЯ ЛЕКЦИОННЫХ, ПРАКТИЧЕСКИХ И ЛАБОРАТОРНЫХ ЗАНЯТИЙ

Темы лекционных занятий

<i>№ п/п</i>	<i>Название темы</i>	<i>Количество часов</i>
1	Введение. Параллельные компьютеры и суперЭВМ. Классификация параллельных вычислительных систем. Типы параллелизма.	2
2	Параллелизм на уровне команд. Конвейер. Суперскалярность. SIMD команды. Параллелизм на уровне задач.	2
3	Аналитические методы определения вычислительной сложности. Оценка вычислительной сложности для последовательных алгоритмов.	2
4	Показатели для оценки параллельных алгоритмов: ускорение, эффективность и стоимость параллельных алгоритмов. Законы Амдала и Густафсона.	2
5	Экспериментальные методы определения вычислительной сложности. Средства профилирования.	2
6	Декомпозиция по функциям и по данным. Планирование параллельных программ.	2
7	Зависимости данных. Способы синхронизации и взаимного исключения потоков.	2
8	Определение стоимости программы с учетом параллельного выполнения.	2
9	Организация параллельных вычислений средствами технологии OpenMP.	2
	ВСЕГО	18

Темы лабораторных занятий

<i>№ п/п</i>	<i>Название темы</i>	<i>Количество часов</i>
1	Параллелизм на уровне команд. Конвейер. Суперскалярность.	2
2	Параллелизм на уровне данных. SIMD команды.	2
3	Параллелизм на уровне задач.	2
4	Оценка вычислительной сложности для последовательных алгоритмов.	2
5	Ускорение, эффективность и стоимость параллельных алгоритмов.	2
6	Экспериментальные методы определения вычислительной сложности.	2
7	Декомпозиция по функциям.	2
8	Декомпозиция по данным.	2
9	Зависимости данных. Типы зависимостей.	2
10	Способы синхронизации.	2
11	Определение стоимости программы с учетом параллельного выполнения.	2
12	Масштабируемость.	2
13	Параллельные и последовательные области. Директива parallel.	2
14	Распараллеливание оператора цикла. Синхронизация и редукция.	2
15	Параллельные алгоритмы матричного умножения.	2
16	Вычисление определенного интеграла в многопроцессорной системе.	2
17	Параллельные алгоритмы сортировки данных.	2
18	Задача поиска кратчайших путей в графе.	2
	ВСЕГО	36

6. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Организация самостоятельной работы студентов

№ n/n	Название темы	Количество часов
1	Ускорение, эффективность и стоимость параллельных алгоритмов.	4
2	Экспериментальные методы определения вычислительной сложности.	4
3	<i>Индивидуальное задание № 1</i>	6
4	Умножение матрицы на вектор.	3
5	Умножение матриц. Эффективность параллельного подхода..	4
6	<i>Индивидуальное задание № 2</i>	6
7	Интегрирование методом прямоугольников.	3
8	Вычисление определенного интеграла в многопроцессорной системе.	3
9	<i>Индивидуальное задание № 3</i>	6
10	Параллельные алгоритмы сортировки данных.	2
11	<i>Индивидуальное задание № 4</i>	5
12	Задача поиска кратчайших путей в графе.	3
13	<i>Индивидуальное задание № 5</i>	5
	ВСЕГО	54

7. ИНДИВИДУАЛЬНЫЕ ЗАДАНИЯ

ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИИ ПАРАЛЛЕЛЬНОГО ПРОГРАММИРОВАНИЯ OPENMP ПРИ РЕАЛИЗАЦИИ ПАРАЛЛЕЛЬНОГО УМНОЖЕНИЯ МАТРИЦ

Цель: овладение технологией OpenMP для реализации параллельных алгоритмов умножения матрицы на вектор, перемножения матриц, вычисления определенного интеграла, чет-нечетной сортировки, алгоритма Флойда поиска кратчайших путей в графе.

Задания:

1) Вычислить элементы двумерного квадратного массива $(a_{ij}) \ (i, j = \overline{0, N-1})$, используя для вычисления функцию $a_{ij} = f(i, j)$ при помощи последовательного и параллельного алгоритмов. Для различных значений размерности массива N и количества процессорных ядер определить экспериментально показатели эффективности для этих алгоритмов и сравнить их с ожидаемыми. Сделать соответствующие выводы.

2) а) Выполните реализацию параллельного алгоритма умножения матрицы на вектор, основанного на ленточном разбиении матрицы на горизонтальные полосы. Проведите вычислительные эксперименты с учетом параметров используемой вычислительной системы. Сравните результаты реальных экспериментов с теоретическими оценками.

б) Выполните параллельную реализацию матричного умножения $A \cdot A$, проведите соответствующие эксперименты и сделайте выводы.

3) Провести численные эксперименты для последовательной и двух параллельных версий двумерного интегрирования функции $f(x, y)$ по области $D = [0, a] \times [0, a]$ с

помощью формулы параллелепипедов в многоядерной вычислительной системе. Построить сравнительные диаграммы. Сделать соответствующие выводы.

4) Выполните реализацию последовательного и параллельного алгоритмов обобщенной пузырьковой сортировки (чет-нечетная перестановка). Массив сортируемых данных генерировать случайным образом. Проведите вычислительные эксперименты. Сравните полученные экспериментальные результаты с теоретическими оценками. Сделайте выводы.

5) Выполните реализацию параллельного алгоритма Флойда. Проведите вычислительные эксперименты. Постройте теоретические оценки с учетом параметров используемой вычислительной системы. Сравните полученные оценки с экспериментальными данными.

8. КОНТРОЛЬНЫЕ ВОПРОСЫ К ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

1. Предположим, что один землекоп может за час вырыть яму размером $1 \times 1 \times 1$ м³ и способен работать в таком режиме достаточно долго. За какое время бригада из 5, 10, 20 землекопов выроет яму размером 2×2 м² и глубиной 1 м.

2. Постройте график времени выполнения работы в зависимости от числа землекопов в бригаде.

3. Повторите задания п. 1, 2 для ямы 10×10 м², глубиной 1 м и бригады из 10, 100 землекопов.

4. Конвейерное устройство состоит из пяти устройств. Времена срабатываний ступеней равны 1, 1, 2, 1 и 3 такта соответственно. С какой максимальной частотой на выходе данного устройства будут появляться результаты, если на его вход аргументы поступают без перебоев?

5. За какое минимальное число тактов может быть выполнено 70 операций, если в распоряжении есть устройство, описанное в задаче 5.

6. В компьютере есть 7 параллельно работающих устройств, каждое из которых может выполнить операцию за 7 единиц времени. За какое минимальное время этот компьютер обработает 7 независимых операций?

7. Конвейерное устройство состоит из k ступеней, срабатывающих за n_1, n_2, \dots, n_k тактов соответственно. За какое минимальное число тактов может быть выполнено m операций на таком устройстве?

8. Назовите уровни параллелизма. К какому уровню относится конвейерная обработка команд?

9. Что такое временная сложность алгоритма?

10. Какие показатели используются для оценки параллельных алгоритмов?

11. Оцените 2 алгоритма, стоимости которых в последовательном и параллельном режиме одинаковы. Имеет ли смысл использовать параллельный алгоритм в этом случае?

12. Определите значения ускорения, эффективности и стоимости, если известны следующие временные характеристики: для 6 процессоров время выполнения последовательной части составляет 25 % и время выполнения параллельной части составляет 75 %.

13. Определите временную сложность для алгоритма вычисления суммы элементов одномерного массива в последовательном режиме. Проверьте экспериментально полученные соотношения.

14. Определите временную сложность для алгоритма сортировки методом простой вставки одномерного массива в последовательном режиме. Проверьте экспериментально полученные соотношения.

15. Определите временную сложность для алгоритма вычисления скалярного произведения двух одномерных векторов в последовательном режиме. Проверьте экспериментально полученные соотношения.

16. Определите временную сложность для алгоритма вычисления произведения матрицы на вектор в последовательном режиме. Проверьте экспериментально полученные соотношения.

17. Определите временную сложность для алгоритма вычисления произведения прямоугольных матриц в последовательном режиме. Проверьте экспериментально полученные соотношения.

18. Определите значения ускорения, эффективности и стоимости для алгоритма вычисления суммы элементов одномерного массива, если: а) число процессоров не ограничено; б) число процессоров задано ($p > 2$).

19. Что определяют условия Бернштейна?

20. Используя *GCD* тест, проверить наличие или отсутствие зависимости для цикла: `for(i=0; i<n; ++i) x[i]=x[2*i];`

21. Может ли ускорение быть больше, чем число ядер процессора, в каком случае это возможно?

22. Для каких операционных систем и языков программирования есть поддержка OpenMP?

23. Как программно проверить, что режим OpenMP включен?

24. В каких единицах возвращает время функция `omp_get_wtime()`? Как определить точность измерения времени? Напишите соответствующий код.

25. Пусть число потоков задано параметром `num_threads(2)`, параметром среды `OMP_NUM_THREAD(3)` и функцией `omp_set_num_threads(4)`. Сколько потоков будет создано для параллельной секции? Проверьте ответ соответствующим кодом программы.

26. Какие параметры надо передать потоковой функции для распараллеливания цикла с известным числом повторений?

27. Определите потоковую функцию для параллельного выполнения цикла сложения двух векторов с фиксированным размером. Напишите код для создания потоков и определите время выполнения суммирования без потоков и с использованием потоков. Определите значение ускорения в зависимости от размера векторов.

28. Составьте функции для вычисления скалярного произведения двух векторов в последовательном и параллельном режимах. Для накопления суммы в параллельном режиме используйте `reduction`. Сравните производительности функций.

29. Составьте функции для умножения матрицы на вектор в последовательном и параллельном режимах. Для накопления сумм в параллельном режиме используйте `reduction`. Сравните производительности функций.

30. Пусть параллельный участок программы содержит вывод элементов массива, сформированных в процессе вычислений. Как обеспечить вывод в естественном порядке, а не в порядке выполнения потоков? Напишите функции с выводом в произвольном и фиксированном порядке. Измерьте время выполнения функций для обоих вариантов. Сделайте выводы.

31. Пусть необходимо накапливать сумму элементов. Для этого можно использовать `reduction` или директиву `#pragma omp atomic`. Какой из этих способов более эффективный? Почему?

32. Как исключить возможность взаимной блокировки при использовании нескольких вложенных критических секций?

9. ОБРАЗЕЦ МОДУЛЬНОГО КОНТРОЛЯ

ГОУ ВПО «ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ»

Факультет математики и информационных технологий

Направление подготовки: **01.04.02 Прикладная математика и информатика**
 Магистерская программа: **Прикладная математика и информатика**
 Программа подготовки: **академическая магистратура**
 Семестр: **III**
 Учебная дисциплина: **Параллельное программирование**

МОДУЛЬНАЯ КОНТРОЛЬНАЯ РАБОТА

ВАРИАНТ №1

1. Назовите уровни параллелизма. К какому уровню относится конвейерная обработка команд?
2. Какие показатели используются для оценки параллельных алгоритмов?
3. Используя *GCD* тест, проверить наличие или отсутствие зависимости для цикла:
`for(i=0; i<n; ++i) x[i]=x[2*i];`
4. Составьте функции для вычисления скалярного произведения двух векторов в последовательном и параллельном режимах. Для накопления суммы в параллельном режиме используйте *reduction*. Сравните производительности функций.

Утверждено на заседании кафедрой теории упругости и вычислительной математики имени академика А.С. Космодамианского, протокол № ____ от “__” _____ 20__ г.

Зав. кафедрой
 Преподаватель

Критерии оценивания модульного контроля

<i>Номер задания</i>	<i>Количество баллов</i>
Задание 1	10
Задание 2	10
Задание 3	10
Задание 4	10
Всего	40

10. ОБРАЗЕЦ ЭКЗАМЕНАЦИОННОГО БИЛЕТА

(не предусмотрен программой)

11. ОБРАЗЕЦ ТЕСТОВОГО ЗАДАНИЯ (не предусмотрен программой)

12. КРИТЕРИИ ОЦЕНИВАНИЯ

По курсу предполагается проведение промежуточной аттестации в виде модульного контроля, выполнение индивидуальных заданий и зачет. Зачет студенты сдают с целью повышения рейтинга.

**Распределение баллов, которые могут получить студенты
в процессе изучения дисциплины**

Организационно учебная работа студента	СРС	
	Индивидуальные задания	Модульный контроль
max 10 баллов	max 50 баллов	max 40 баллов

Шкала соответствия баллов национальной шкале

Оценка по шкале ECTS	Оценка по 100-балльной шкале	Оценка по государственной шкале (экзамен, дифференцированный зачет)	Оценка по государственной шкале (зачет)
A	90-100	5 (отлично)	зачтено
B	80-89	4 (хорошо)	зачтено
C	75-79	4 (хорошо)	зачтено
D	70-74	3 (удовлетворительно)	зачтено
E	60-69	3 (удовлетворительно)	зачтено
FX	35-59	2 (неудовлетворительно) с возможностью повторной сдачи	не зачтено
F	0-34	2 (неудовлетворительно) с возможностью повторной сдачи при условии обязательного набора дополнительных баллов	не зачтено

13. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОГО ПРОЦЕССА

Лекционные занятия проводятся в аудитории, оснащенной доской. Лабораторные занятия проводятся в компьютерном классе, оборудованном компьютерами с лицензионным программным обеспечением, доступом к сети Интернет, столами, доской.

14. РЕКОМЕНДОВАННАЯ ЛИТЕРАТУРА

№ п/п	Наименование	Кол-во экземпляров в библиотеке ДонНУ	Наличие электронной версии в ЭБС
Основная литература			
1.	Нескородев, Р.Н. Параллельное программирование [Электронный ресурс]: учебное пособие / Р.Н. Нескородев; ГОУ ВПО “Донецкий национальный университет”. – Донецк: ДонНУ, 2019. – электронные данные (1 файл).	10	+
2.	Нескородев, Р.Н. Организация параллельных вычислений при помощи технологии OpenMP [Электронный ресурс]: учебно-методическое пособие для студентов направления подготовки 01.04.02 Прикладная математика и информатика / Р. Н. Нескородев; ГОУ ВПО “Донецкий национальный университет”. – Донецк: ДонНУ, 2019. – электронные данные (1 файл).	0	+

<i>Дополнительная литература</i>			
3.	Воеводин В.В. Параллельные вычисления: Учеб. пособие для вузов по направлению "Прикладная математика и информатика" / В.В. Воеводин, Вл.В. Воеводин. – СПб.: БХВ-Петербург, 2002. – 608 с.	1	-
4.	Качко Е.Г. Параллельное программирование / Е.Г. Качко. – Харьков: Форт, 2011. – 528 с.	6	-
5.	Барский А.Б. Параллельные информационные технологии / А.Б. Барский. – М.: Интернет-Ун-т информ. технологий: БИНОМ. Лаб. знаний, 2007. – 502 с.	2	-
6.	Гергель В.П. Теория и практика параллельных вычислений / В.П. Гергель. – М.: Бином. Лаборатория знаний, 2007. – 424 с.	2	-
7.	Калоеров С.А. Программирование на языке C++: учеб. пособие / С.А. Калоеров. – Донецк: Юго-Восток, 2009. – 298 с.	85	-
8.	Бахвалов, Н. С. Численные методы: Учеб. пособие для студентов физ.-мат. спец. вузов / Н. С. Бахвалов, Н. П. Жидков, Г. М. Кобельков. – 2-е изд. – М.: Лаб. Баз. Знаний; СПб.: Невский диалект, 2008. – 640 с.	7	-

15. ИНФОРМАЦИОННЫЕ РЕСУРСЫ

<http://parallel.ru> - Лаборатории Параллельных информационных технологий Научно-исследовательского вычислительного центра Московского государственного университета имени М.В.Ломоносова;

16. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Операционная система Microsoft Windows 7, среда разработки Microsoft Visual Studio.

Рабочая программа рассмотрена и переутверждена на заседании кафедры теории упругости и вычислительной математики имени академика А.С. Космодамианского с изменениями (без изменений) на 20____ год.

Протокол № ____ от “ ____ ” _____ 20__ г.

Заведующий. кафедрой

_____ В.И. Сторожев

Рабочая программа рассмотрена и переутверждена на заседании кафедры теории упругости и вычислительной математики имени академика А.С. Космодамианского с изменениями (без изменений) на 20____ год.

Протокол № ____ от “ ____ ” _____ 20__ г.

Заведующий. кафедрой

_____ В.И. Сторожев

Рабочая программа рассмотрена и переутверждена на заседании кафедры теории упругости и вычислительной математики имени академика А.С. Космодамианского с изменениями (без изменений) на 20____ год.

Протокол № ____ от “ ____ ” _____ 20__ г.

Заведующий. кафедрой

_____ В.И. Сторожев