

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Ярославский государственный университет им. П.Г. Демидова
Факультет информатики и вычислительной техники

УТВЕРЖДАЮ

Проректор по развитию образования

_____ Е.В. Сапир

" ____ " _____ 2012 г.

**Рабочая программа дисциплины
послевузовского профессионального образования
(аспирантура)**

**Математическое и программное обеспечение вычислительных машин,
комплексов и компьютерных сетей**

по специальности научных работников

**05.13.11 «Математическое и программное обеспечение вычислительных
машин, комплексов и компьютерных сетей»**

Ярославль 2012

1. Цели освоения дисциплины

Целями освоения дисциплины «Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей» в соответствии с общими целями основной профессиональной образовательной программы послевузовского профессионального образования (аспирантура) (далее ОПП – образовательная программа послевузовского профессионального образования) являются получение обучающимися знаний в области, включающими в себя задачи развития теории программирования, создания и сопровождения программных средств различного назначения. Значение решения указанных проблем состоит в повышении эффективности и надежности процессов обработки и передачи данных и знаний в вычислительных машинах, комплексах и компьютерных сетях.

2. Место дисциплины в структуре ООП послевузовского профессионального образования (аспирантура)

Данная дисциплина относится к разделу обязательные дисциплины (подраздел специальные дисциплины отрасли науки и научной специальности ОД.А.03.) образовательной составляющей образовательной программы послевузовского профессионального образования по специальности научных работников 05.13.11 Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей.

Дисциплина «Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей» необходима для решения задач создания научных основ современных информационных технологий на базе использования средств вычислительной техники и ускорения на этой основе научно-технического прогресса.

Для изучения данной дисциплины необходимы «входные» знания, умения, полученные в процессе обучения по программам специалитета или бакалавриата-магистратуры. Дисциплина «Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей» опирается на дисциплины, входящие в план подготовки студентов по соответствующим программам специалитета, бакалавриата и магистратуры. От аспиранта требуется наличие логического мышления, образованность, организованность и трудолюбие, самостоятельность, настойчивость в достижении цели, а также знания, полученные при обучении по указанным выше программам.

3. Требования к результатам освоения содержания дисциплины

В результате освоения дисциплины обучающийся должен:

Знать:

1. Языки программирования и системы программирования.
2. Операционные системы.
3. Системы управления базами данных и знаний.
4. Программные системы символьных вычислений.
5. Языки и инструментальные средства параллельного программирования.

Уметь:

- 1) создавать модели, методы и алгоритмы проектирования и анализа программ и программных систем, их эквивалентных преобразований, верификации и тестирования;

- 2) разрабатывать модели, методы, алгоритмы, языки и программные инструменты для организации взаимодействия программ и программных систем;
- 3) разрабатывать модели и методы создания программ и программных систем для параллельной и распределенной обработки данных,

Владеть:

- 1) методами оценки качества, стандартизация и сопровождение программных систем; методами анализа информационных и программных систем;
- 2) методами, алгоритмами и программными средствами для организации глобально распределенной обработки данных;
- 3) методами организации человеко-машинного интерфейса;
- 4) методами, алгоритмами и программными средствами машинной графики, визуализации, обработки изображений, систем виртуальной реальности, мультимедийного общения.

4. Структура и содержание дисциплины «Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей»

Общая трудоемкость дисциплины составляет 4 зачетные единицы, 144 часа.

№ п/п	Раздел Дисциплины	Курс	Неделя	Виды учебной работы, включая самостоятельную работу обучающихся, и трудоемкость (в часах) Форма обуч.: очная/заочная					Формы текущего контроля успеваемости (по неделям) Форма промежуточной аттестации
				Лекций	Лабораторных	Практических	Сам. работа	Контроль сам. работы	
1	Раздел 1. Математические основы программирования	1	1-2	2/1			24/ 25		Индивидуальное задание № 1 (срок 2 неделя)
2	Раздел 2. Вычислительные машины, системы и сети	1	3-4	1			19		Индивидуальное задание № 2 (срок 4 неделя)
3	Раздел 3. Языки и системы программирования. Технология разработки программного обеспечения	1	5-6	1			19		
4	Раздел 4. Операционные системы	1	7-8	2			24		Индивидуальное задание № 3 (срок 8 неделя)
5	Раздел 5. Методы хранения данных и доступа к ним. Организация баз данных и знаний	1	9-10	2			24		
6	Раздел 6. Защита данных и программных систем	1	11-12	2/1			24/ 25		Индивидуальное задание № 4 (срок 12 неделя)
	Всего	1	12	10/8			134/ 136		зачет

Содержание дисциплины

Раздел 1. Математические основы программирования

Понятие алгоритма и его уточнения: машины Тьюринга, нормальные алгоритмы Маркова, рекурсивные функции. Эквивалентность данных формальных моделей алгоритмов. Понятие об алгоритмической неразрешимости. Понятие сложности алгоритмов. Классы P и NP. Полиномиальная сводимость задач. Теорема Кука об NP-полноте задачи выполнимости булевой формулы. Точные и приближенные комбинаторные алгоритмы.

Автоматы. Эксперименты с автоматами. Алгебры регулярных выражений. Теорема Клини о регулярных языках.

Алгебра логики. Булевы функции, канонические формы задания булевых функций. Понятие полной системы. Критерий полноты Поста. Минимизация булевых функций в классах нормальных форм.

Исчисление предикатов первого порядка. Понятие интерпретации. Выполнимость и общезначимость формулы первого порядка. Понятие модели. Теорема о полноте исчисления предикатов первого порядка.

Отношения и функции. Отношение эквивалентности и разбиения. Фактор множества. Отношения частичного порядка. Теоретико-множественное и алгебраическое определения решетки, их эквивалентность. Свойства решеток. Булевы решетки. Полные решетки.

Формальные языки и способы их описания. Классификация формальных грамматик. Их использование в лексическом и синтаксическом анализе.

“мю”-исчисление, правила редукции, единственность нормальной формы и правила ее достижения, представление рекурсивных функций.

Основы комбинаторного анализа. Метод производящих функций, метод включений и исключений. Коды с исправлением ошибок. Алфавитное кодирование. Методы сжатия информации.

Основы криптографии. Задачи обеспечения конфиденциальности и целостности информации. Теоретико-информационный и теоретико-сложностный подходы к определению криптографической стойкости. Американский стандарт шифрования DES и российский стандарт шифрования данных ГОСТ 28147-89. Системы шифрования с открытым ключом (RSA). Цифровая подпись. Методы генерации и распределения ключей.

Раздел 2. Вычислительные машины, системы и сети

Архитектура современных компьютеров. Организации памяти и архитектура процессора современных вычислительных машин. Страничная и сегментная организация виртуальной памяти. Кэш-память. Командный и арифметический конвейеры, параллельное выполнение независимых команд, векторные команды. Специализированные процессоры. Машины, обеспечивающие выполнение вычислений, управляемых потоком данных. Организация ввода-вывода, каналы и процессоры ввода-вывода, устройства сопряжения с объектами.

Классификация вычислительных систем (ВС) по способу организации параллельной обработки. Многопроцессорные и многомашинные комплексы. Вычислительные кластеры. Проблемно-ориентированные параллельные структуры: матричные ВС, систолические структуры, нейросети.

Назначение, архитектура и принципы построения информационно – вычислительных сетей (ИВС). Локальные и глобальные ИВС, технические и программные средства объединения различных сетей.

Методы и средства передачи данных в ИВС, протоколы передачи данных. Особенности архитектуры локальных сетей (Ethernet, Token Ring, FDDI).

Сеть Internet, доменная организация, семейство протоколов TCP/IP. Информационно-вычислительные сети и распределенная обработка информации.

Раздел 3. Языки и системы программирования. Технология разработки программного обеспечения

Процедурные языки программирования (Фортран, Си). Функциональные языки программирования (Лисп), логическое программирование (Пролог), объектно-ориентированные языки программирования (Ява).

Процедурные языки программирования. Основные управляющие конструкции, структура программы. Работа с данными: переменные и константы, типы данных (булевский, целочисленные, плавающие, символьные, типы диапазона и перечисления, указатели), структуры данных (массивы и записи). Процедуры (функции): вызов процедур, передача параметров (по ссылке, по значению, по результату), локализация переменных, побочные эффекты. Обработка исключительных ситуаций. Библиотеки процедур и их использование.

Объектно-ориентированное программирование. Классы и объекты, наследование, интерфейсы. Понятие об объектном окружении. Рефлексия. Библиотеки классов. Средства обработки объектов (контейнеры и итераторы).

Распределенное программирование. Процессы и их синхронизация. Семафоры, мониторы Хоара. Объектно-ориентированное распределенное программирование. CORBA. Параллельное программирование над общей памятью. Нити. Стандартный интерфейс Open MP. Распараллеливание последовательных программ. Параллельное программирование над распределенной памятью. Парадигмы SPMD и MIMD. Стандартный интерфейс MPI.

Основы построения трансляторов. Структура оптимизирующего транслятора. Промежуточные представления программы: последовательность символов, последовательность лексем, синтаксическое дерево, абстрактное синтаксическое дерево. Уровни промежуточного представления: высокий, средний, низкий. Формы промежуточного представления.

Анализ исходной программы в компиляторе. Автоматные (регулярные) грамматики и сканирование, контекстно свободные грамматики и синтаксический анализ, организация таблицы символов программы, имеющей блочную структуру, хеш-функции. Нисходящие (LL(1)-грамматики) и восходящие (LR(1)-грамматики) методы синтаксического анализа. Атрибутные грамматики и семантические программы, построение абстрактного синтаксического дерева. Автоматическое построение лексических и синтаксических анализаторов по формальным описаниям грамматик. Системы lex и yacc. Система Gentle.

Оптимизация программ при их компиляции. Оптимизация базовых блоков, чистка циклов. Анализ графов потока управления и потока данных. Отношение доминирования и его свойства, построение границы области доминирования вершины, выделение сильно связанных компонент графа.

Построение графа зависимостей. Перевод программы в SSA-представление и обратно. Глобальная и межпроцедурная оптимизация.

Генерация объектного кода в компиляторах. Перенастраиваемые (retargetable) компиляторы, gcc (набор компиляторов Gnu). Переработка термов (term rewriting). Применение оптимизационных эвристик (целочисленное программирование, динамическое программирование) для автоматической генерации генераторов объектного кода (системы BEG, Iburg и др.). Машинно-ориентированные языки, язык ассемблера. Представление машинных команд и констант. Команды транслятору. Их типы, принципы реализации. Макросредства, макровыводы, языки макроопределений, условная макрогенерация, принципы реализации.

Системы программирования (СП), типовые компоненты СП: языки, трансляторы, редакторы связей, отладчики, текстовые редакторы. Модульное программирование. Типы модулей. Связывание модулей по управлению и данным.

Пакеты прикладных программ (ППП). Системная часть и наполнение. Языки общения с ППП. Машинная графика. Средства поддержки машинной графики. Графические пакеты.

Технология разработки и сопровождения программ. Жизненный цикл программы. Этапы разработки, инкапсуляции, задания типов. Модули, взаимодействие между модулями, иерархические структуры программ.

Отладка, тестирование, верификация и оценивание сложности программ. Генерация тестов. Системы генерации тестов. Срезы программ (slice, chop) и их применение при отладке программ и для генерации тестов.

Методы спецификации программ. Методы проверки спецификации. Схемное, структурное, визуальное программирование. Разработка пользовательского интерфейса, стандарт CUA, мультимедийные среды интерфейсного взаимодействия.

Раздел 4. Операционные системы

Режимы функционирования вычислительных систем, структура и функции операционных систем. Основные блоки и модули. Основные средства аппаратной поддержки функций операционных систем (ОС): система прерываний, защита памяти, механизмы преобразования адресов в системах виртуальной памяти, управление каналами и периферийными устройствами.

Виды процессов и управления ими в современных ОС. Представление процессов, их контексты, иерархии порождения, состояния и взаимодействие. Многозадачный (многопрограммный) режим работы. Команды управления процессами. Средства взаимодействия процессов. Модель клиент-сервер и ее реализация в современных ОС.

Параллельные процессы, схемы порождения и управления. Организация взаимодействия между параллельными и асинхронными процессами: обмен сообщениями, организация почтовых ящиков. Критические участки, примитивы взаимоисключения процессов, семафоры Дейкстры и их расширения. Проблема тупиков при асинхронном выполнении процессов, алгоритмы обнаружения и предотвращения тупиков.

Операционные средства управления процессами при их реализации на параллельных и распределенных вычислительных системах и сетях: стандарты и программные средства PVM, MPI, OpenMP, POSIX .

Одноуровневые и многоуровневые дисциплины циклического обслуживания процессов на центральном процессоре, выбор кванта.

Управление доступом к данным. Файловая система, организация, распределение дисковой памяти. Управление обменом данными между дисковой и оперативной памятью. Рабочее множество страниц (сегментов) программы, алгоритмы его определения.

Управление внешними устройствами.

Оптимизация многозадачной работы компьютеров. Операционные системы Windows, Unix, Linux. Особенности организации, предоставляемые услуги пользовательского взаимодействия.

Операционные средства управления сетями. Эталонная модель взаимодействия открытых систем ISO/OSI. Маршрутизация и управление потоками данных в сети. Локальные и глобальные сети. Сетевые ОС, модель клиент-сервер, средства управления сетями в ОС UNIX, Windows NT. Семейство протоколов TCP/IP, структура и типы IP-адресов, доменная адресация в Internet. Транспортные протоколы TCP, UDP .

Удаленный доступ к ресурсам сети. Организация электронной почты, телеконференций. Протоколы передачи файлов FTP и HTTP, язык разметки гипертекста HTML, разработка WEB-страниц, WWW-серверы.

Раздел 5. Методы хранения данных и доступа к ним. Организация баз данных и знаний

Концепция типа данных. Абстрактные типы данных. Объекты (основные свойства и отличительные признаки).

Основные структуры данных, алгоритмы обработки и поиска. Сравнительная характеристика методов хранения и поиска данных.

Основные понятия реляционной и объектной моделей данных.

Теоретические основы реляционной модели данных (РДМ). Реляционная алгебра, реляционное исчисление. Функциональные зависимости и нормализация отношений.

CASE-средства и их использование при проектировании базы данных (БД).

Организация и проектирование физического уровня БД. Методы индексирования.

Обобщенная архитектура, состав и функции системы управления базой данных (СУБД). Характеристика современных технологий БД.

Основные принципы управления транзакциями, журнализацией и восстановлением.

Язык баз данных SQL. Средства определения и изменения схемы БД, определения ограничений целостности. Контроль доступа. Средства манипулирования данными.

Стандарты языков SQL. Интерактивный, встроенный, динамический SQL.

Основные понятия технологии клиент-сервер. Характеристика SQL-сервера и клиента. Сетевое взаимодействие клиента и сервера.

Информационно-поисковые системы. Классификация. Методы реализации и ускорения поиска.

Методы представления знаний: процедурные представления, логические представления, семантические сети, фреймы, системы продукций. Интегрированные методы представления знаний. Языки представления знаний. Базы знаний.

Экспертные системы (ЭС). Области применения ЭС. Архитектура ЭС. Механизмы вывода, подсистемы объяснения, общения, приобретения знаний ЭС. Жизненный цикл экспертной системы. Примеры конкретных ЭС.

Раздел 6. Защита данных и программных систем

Аппаратные и программные методы защиты данных и программ. Защита данных и программ с помощью шифрования.

Защита от несанкционированного доступа в ОС Windows NT. Система безопасности и разграничения доступа к ресурсам в Windows NT. Файловая система NTFS и сервисы Windows NT.

Защита от несанкционированного копирования. Методы простановки не копируемых меток, настройка устанавливаемой программы на конкретный компьютер, настройка на конфигурацию оборудования.

Защита от разрушающих программных воздействий. Вредоносные программы и их классификация. Загрузочные и файловые вирусы, программы-закладки. Методы обнаружения и удаления вирусов, восстановления программного обеспечения.

Защита информации в вычислительных сетях Novell Netware, Windows NT и др.

5. Образовательные технологии

В основу образовательной технологии по дисциплине «Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей» помимо традиционных форм занятий в виде лекций положена также форма, состоящая в выполнении аспирантом индивидуальных заданий по темам дисциплины. Имеются четыре индивидуальных задания по дисциплине. Задания должны быть решены письменно или в электронном виде с последующей устной защитой. Аспиранты, сдавшие все индивидуальные задания в установленные сроки, после успешного ответа на ряд

дополнительных вопросов, закрывающих оставшиеся темы, получают отметку о сдаче зачета досрочно. Такой подход стимулирует постоянную работу аспирантов и активизирует усвоение материала. Эта технология позволяет проводить индивидуальное обучение аспирантов и дает хорошие результаты. Она дополняется обсуждением общих (типичных) ошибок на лекционных занятиях.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы обучающихся

Текущий контроль успеваемости аспирантов организован в виде индивидуальных заданий, которые должен выполнить каждый обучающийся. В предыдущем разделе описана технология индивидуального обучения аспирантов при помощи таких заданий. Промежуточная аттестация по итогам освоения дисциплины проводится в виде зачета.

Примеры индивидуальных заданий

1. Назовите известные Вам предметные области, в которых широко используются информационные и телекоммуникационные технологии. Расскажите о специфике аппаратного и программного обеспечения, применяемого в этих областях.

2. В настоящее время в сети Интернет применяются различные браузеры. Чем обусловлено это разнообразие. Опишите основные достоинства и недостатки предпочитаемого Вами браузера.

3. Функционирование компьютеров любого типа немислимо без использования операционных систем. Какие операционные системы Вы предпочитаете использовать на своих десктопах, ноутбуках, нетбуках, коммуникаторах, серверах? Чем обуславливается Ваш выбор?

4. Рассмотрите основные критерии качества программного обеспечения. Какими, на ваш взгляд, методами и техническими приемами можно обеспечить поддержку различных критериев качества программного обеспечения?

5. Определите противоречия, возникающие при одновременной поддержке различных критериев качества программного обеспечения. Какие подходы и приемы, на Ваш взгляд, могут обеспечить сглаживание этих противоречий. Приведите примеры для различных комбинаций критериев.

6. Сопоставьте критерии качества программ с различными прикладными областями. Какие из критериев имеют наибольшее значение. Распределите известные Вам критерии по важности для выбранной предметной области.

7. Для выбранной предметной области опишите используемые в ней специализированные модели. Выделите формальные и неформальные модели. Обоснуйте независимость или зависимость представленных моделей от методов реализации в виде соответствующего программного обеспечения.

8. Приведите особенности различных моделей исполнителей. Опишите, как эти особенности влияют на реализацию моделей предметной области. Увяжите между собой выбранные модели предметной области и исполнителя. Опишите особенности семантического разрыва, возникающего между этими моделями.

9. На примере заданной вычислительной задачи покажите, каким образом изменяются принципы написания программы в зависимости от архитектуры используемой вычислительной системы.

10. Проведите сравнительный анализ двух заданных архитектур параллельных вычислительных систем покажите, по каким параметрам каждая из систем превосходит другую. По каким причинам любая из рассмотренных архитектур не может превзойти другую по всем параметрам?

11. Приведите примеры существующих параллельных вычислительных систем, в которых совмещены несколько архитектурных решений. Каковы достоинства и недостатки таких гибридных организаций?
12. Выделите информационный граф заданного алгоритма. Рассмотрите, какие стратегии управления вычислениями на него можно наложить.
13. На примере заданной последовательной программы выделите ее информационный граф. На информационном графе постройте управляющий граф, соответствующий последовательному выполнению программы.
14. Для заданного информационного графа программы постройте несколько управляющих графов, реализующих различные стратегии вычислений.
15. Какие ресурсные ограничения возникают при использовании переменных, располагающихся в общей памяти? Как эти ограничения могут влиять на параллелизм алгоритма?
16. Какие ресурсные ограничения возникают при организации цикла? Как они влияют на параллелизм задач? Приведите примеры.
17. Выделите агрегаты и обобщения на примерах различных предметов и природных явлений. Рассмотрите выделенные понятия с точки зрения процедурного и объектно-ориентированного подхода.
18. Процедурный подход обладает большой гибкостью. Он позволяет смоделировать объектно-ориентированный стиль программирования. Приведите примеры реализации объектно-ориентированных конструкций с применением процедурного подхода. Можно ли при этом достигнуть того же уровня эволюционного расширения программы, который обеспечивает объектно-ориентированный стиль?
19. В существующих языках программирования агрегаты и обобщения формируются различными способами. На примере одного из языков программирования покажите, каким путем в нем создаются агрегаты и обобщения. Оцените возможности эволюционного расширения в данном языке рассмотренных конструкций.
20. Проанализируйте исходные тексты программы, написанные на одном из объектно-ориентированных языков. Оцените чистоту объектно-ориентированного стиля. Существуют ли в рассматриваемой программе фрагменты кода, которые оформлены в процедурном стиле?

Вопросы к зачету.

1. Математические основы программирования

Понятие алгоритма и его уточнения.

Понятие сложности алгоритмов. Классы P и NP.

Автоматы.

Алгебра логики.

Отношения и функции. Отношения частичного порядка.

Формальные языки и способы их описания.

Коды с исправлением ошибок. Методы сжатия информации.

Основы криптографии.

2. Вычислительные машины, системы и сети

Архитектура современных компьютеров.

Классификация вычислительных систем.

Назначение, архитектура и принципы построения информационно-вычислительных сетей.

Протоколы передачи данных.

Особенности архитектуры локальных сетей (Ethernet, Token Ring, FDDI).

Сеть Internet, доменная организация, семейство протоколов TCP/IP.

3. Языки и системы программирования. Технология разработки программного обеспечения

Языки программирования.
 Распределенное программирование.
 Параллельное программирование над общей памятью.
 Параллельное программирование над распределенной памятью.
 Основы построения трансляторов.
 Пакеты прикладных программ.
 Отладка, тестирование, верификация и оценивание сложности программ.
 Методы спецификации программ.

4. Операционные системы

Виды процессов и управления ими в современных ОС.
 Параллельные процессы, схемы порождения и управления.
 Управление доступом к данным.
 Управление внешними устройствами.
 Оптимизация многозадачной работы компьютеров.
 Операционные средства управления сетями.
 Удаленный доступ к ресурсам сети.

5. Методы хранения данных и доступа к ним. Организация баз данных и знаний

Концепция типа данных.
 Основные структуры данных, алгоритмы обработки и поиска.
 Основные понятия реляционной и объектной моделей данных.
 Характеристика современных технологий БД.
 Язык баз данных SQL.
 Информационно-поисковые системы.
 Методы представления знаний.
 Экспертные системы.

6. Защита данных и программных систем

Аппаратные и программные методы защиты данных и программ.
 Защита от несанкционированного копирования.
 Защита от разрушающих программных воздействий.
 Защита информации в вычислительных сетях.

7. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

1. Ахо, Сети Р., Ульман Дж. Компиляторы: принципы, техника реализации и инструменты. М., 2001.
2. Введение в криптографию / Под ред. В.В. Яценко. СПб.: МЦНМО, 2001.
3. Дейт К.Дж. Введение в системы баз данных. М.: Вильямс, 1999.
4. Дейтел Г. Введение в операционные системы. М.: Мир, 1987.
5. Кнут Д. Искусство программирования. Т. 1 – 3. М., СПб., Киев: ИД «Вильямс», 2000.
6. Когаловский М.Р. Энциклопедия технологий баз данных. М.: Финансы и статистика, 2002.
7. Компьютерные сети. Учебный курс Microsoft Corporation, 1997.
8. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы, построение и анализ. М.: МЦНМО, 2000.
9. Котов В.Е., Сабельфельд В.К. Теория схем программ. М.: Наука, 1991.
10. Матфик С. Механизмы защиты в сетях ЭВМ. М.: Мир, 1993.
11. Мельников В.В. Защита информации в компьютерных системах. М.: Финансы и статистика, 1997.
12. Яблонский С.В. Введение в дискретную математику. М.: Наука, 2001.

б) дополнительная литература:

1. Керниган Б., Пайк П. UNIX – универсальная среда программирования. М.: Финансы и статистика, 1992.
2. Корнеев В.В. Параллельные вычислительные системы. М.: Нолидж, 1999.
3. Королёв Л.Н. Структуры ЭВМ и их математическое обеспечение. М.: Наука, 1980.
4. Соломон Д., Руссинович М. Внутреннее устройство Microsoft Windows 2000. СПб.: Питер, 2001.

в) программное обеспечение и Интернет-ресурсы:

1. SMV. Symbolic Model Verifier. Carnegie Mellon University.
<http://www.cs.cmu.edu/~modelcheck/smv.html>
2. SPIN. <http://spinroot.com/spin/whatispin.html>
3. UPPAAL. <http://www.uppaal.com>

8. Материально-техническое обеспечение дисциплины

Компьютер, мультимедийный проектор, набор электронных презентаций и схем.

Программа составлена в соответствии с федеральными государственными требованиями к структуре основной профессиональной образовательной программы послевузовского профессионального образования (аспирантура) (приказ Минобрнауки от 16.03.2011 г. № 1365) с учетом рекомендаций, изложенных в письме Минобрнауки от 22.06.2011 г. № ИБ – 733/12.

Программа одобрена на заседании кафедры теоретической информатики.

17.10.2012 г. (протокол № 11)

Заведующий кафедрой

В.А. Соколов, д.ф.-м.н., проф.

Автор

В.А. Соколов, д.ф.-м.н., проф.