

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

**Языки и технологии программирования
Programming language and Software Engineering**

**Направление подготовки
МАТЕМАТИКА**

УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС

Для специальности:
Направление
010400 «Прикладная математика и информатика»

г. _____ – 2013 г.

Утверждено на заседании кафедры вычислительной математики механико-математического факультета МГУ, __.__.2013. Протокол № __.

Составитель:

_____ кафедры вычислительной математики механико-математического факультета МГУ,
к.ф.-м.н. А.С.Козицын

Учебно-методический комплекс «Языки и технологии программирования» составлен в соответствии с требованиями Государственного образовательного стандарта высшего профессионального образования по специальности 010400 «Прикладная математика и информатика».

Дисциплина входит в региональный компонент цикла общепрофессиональных дисциплин (ОПД.Р.03) и является обязательной для изучения.

1. Цели и задачи освоения дисциплины.

Целями освоения дисциплины «Языки и технологии программирования» являются подготовка в области применения современной вычислительной техники для решения практических задач обработки данных, математического моделирования, информатики, получение высшего профессионального (на уровне бакалавра) образования, позволяющего выпускнику успешно работать в избранной сфере деятельности с применением современных компьютерных технологий.

Задачами освоения дисциплины «Языки и технологии программирования» являются:

- 1) изучение классических алгоритмов обработки данных;
- 2) освоение языка C;
- 3) освоение одного из математических пакетов (Maple, Mathematica);
- 4) изучение базовых принципов построения информационных систем;
- 5) получение навыков работы на javascript, shell, SQL, XPath.

2. Место дисциплины в структуре ООП ВПО.

Дисциплина входит в вариативную часть цикла естественно-научных дисциплин. Для изучения и освоения дисциплины нужны первоначальные знания из курсов математического анализа и алгебры. Знания и умения, приобретенные студентами в результате изучения дисциплины, будут использоваться при изучении курсов «Численные методы», «Архитектура ЭВМ и системное программирование», «Базы данных», а также при выполнении курсовых и дипломных работ, связанных с математическим моделированием и обработкой наборов данных.

3. Компетенции обучающегося, формируемые в результате освоения дисциплины: ОК-1, ОК-6, ОК-7, ОК-8, ОК-10, ОК-12, ОК-15, ОК-17, ОК-18, ПК-2, ПК-3, ПК-12, ПК-17, ПК-19, ПК-20, ПК-21

В результате освоения данной дисциплины обучающийся должен:

- 1) Знать: методы и технологии программирования, базовые алгоритмы обработки данных, иметь представление о современных языках программирования, о структуре вычислительных систем и способах сетевого взаимодействия.
- 2) Уметь: разрабатывать алгоритмы, реализовывать алгоритмы на языке программирования высокого уровня, реализовывать методы анализа и обработки данных, работать в средах программирования;
- 3) Владеть: методами и технологиями разработки алгоритмов, программирования на языке высокого уровня, работы в различных средах программирования.

4. Структура и содержание дисциплины

4.1 Содержание разделов дисциплины.

4.1.1 Базовые конструкции языка C, типы данных, структура программы.

Структура программы, основные блоки, функции, локальные и глобальные переменные, указатели, циклы, ветвления, рекурсия, массивы, выделение памяти.

4.1.2 Алгоритмы сортировки.

Сортировка выбором, пузырьком, шейкерная сортировка, простыми вставками, Шелла, пирамидальная сортировка, быстрая сортировка, сортировка слиянием, блочная сортировка, поразрядная сортировка, сортировка подсчетом.

4.1.3 Алгоритмы сжатия данных.

Сжатие с потерей и без потери данных. Алгоритмы RLE, Хаффмана, Лемпеля-Зива, арифметическое кодирование, алгоритм Лемпеля-Зива-Велча, использование буфера, двухступенчатое сжатие.

4.1.4 Алгоритмы шифрования данных.

Психологически проблемы безопасности, определение проблемы обмена для двух закрытых

контуров, классификация возможности нападающего, задачи и методы защиты, типы шифров, шифр Цезаря, статистический метод взлома Цезаря, шифр равного ключа (блокнота), RSA, проблемы центров и передачи ключа, использование дискретного логарифма, протокол Шамира, подпись подлинности, пример подделки подписи, шифры с закрытым ключом, хэш функции, блочные шифры, Гост, комбинирование открытого и закрытого ключа.

4.1.5 Алгоритмы кодирования

Коды Грэя, коды Хэмминга, коды Фибоначчи.

4.1.6 Технологии поиска. Типы индексов

Двоичные деревья, хэш-индекс, В-деревья, В+-деревья, R-деревья, красно-черные деревья, битовые карты.

4.1.7 Работа с матрицами, быстрое умножение.

Метод Карацубы, метод Шенхыге-Штрассена, алгоритм Штрассена.

4.1.8 Методы динамического программирования.

Пример вычислений чисел Фибоначчи, принцип динамического программирования, Алгоритм Флойда — Уоршелла, алгоритм Дейкстры, алгоритм Джонсона, алгоритм Беллмана — Форда.

4.1.9 Теория игр, критерии принятия решений

Примеры, понятие стратегии, классификация игрового дерева позиций, цена игры, игры с нулевой суммой. Статистические критерии, Критерий минимакса, критерий Вальда, критерий Гурвица, критерий Байеса - Лапласа, критерий Ходжа-Лемана, BL (MM) критерий, критерий произведений, критерий Сэвиджа.

4.1.10 Специализированные пакеты для математических расчетов.

Знакомство с Maple и Mathematica. Основные типы данных, язык пакетной обработки, основные библиотеки.

4.1.11 Грамматики

Аналитические и формальные грамматики. Неограниченные, контекстнозависимые, контекстнонезависимые и регулярные грамматики. Процедуры вывода. Регулярные грамматики левые и правые. Деревья разбора. Семантика. Анализ сверху вниз LL-анализатор, снизу вверх LR и GLR-анализатор. Коррекция ошибок, компиляторы. Однозначные и неоднозначные (существенно и несущественно) грамматики.

4.1.12 Методы визуализации данных (деревьев, многомерных пространств). Задачи о рюкзаке и о графах.

Кластеризация, самоорганизующиеся карты, деревья, деление плоскости. Классы задач, машина Тьюринга, самораспознающийся автомат. NP-задачи: логическая формула, обобщенные пятнашки, коммивояжер, минимальная сеть в графе, хроматическое число графа, минимальное вершинное покрытие, минимальное покрытие множества, максимальный полный подграф, максимальный независимый подграф, сапер и тетрис. Приближенные методы (Метод Монтекарло). Применение в шифровании. Доказательство с нулевым знанием.

4.1.13 Методы работы с полуструктурированными данными.

Языки XML, XPath, XQuery.

4.1.14 Язык Javascript.

Работа с памятью, понятие виртуальной машины. Основные языковые конструкции. Примеры программ.

4.1.15 Язык VBA.

Типы данных, основные операторы, примеры программ.

4.1.16 Скрипты в shell и cmd.

Основные функции операционной системы. Система команд. Метод пакетной работы в операционной системе. Основные операторы. Примеры программ.

4.1.17 Архитектуры систем 1, 2, 3 звенные. Распределенные объекты. Технологии промежуточного слоя. Методы обеспечения отказоустойчивости, перераспределения нагрузки.

Достоинства и недостатки основных архитектур построения клиент-серверных систем. Методы дублирования данных, распределения данных, резервирования данных, дублирования критичных узлов системы. Методы распределения нагрузки. Основные

виды репликаций баз данных, достоинства и недостатки.

4.1.18 Экспертные системы, генетические алгоритмы, нейронный сети, нечеткие множества.

Правила вывода, нечеткая логика. Понятие мутации и эволюции. Понятие нейрона, способы построения и обучения нейронной сети.

4.1.19 Обзор сетевых технологий и протоколов

Основные уровни сетевых протоколов. Понятие мак-адреса и IP-адреса, ARP-запросы. Способы маршрутизации, DNS, DHCP, Gateway, маски сети, классы сети. Проблемы оптимизации передачи в TCP. Обзор протоколов верхнего уровня.

4.1.20 Обработка текстовых данных

Морфологический, синтаксический и семантический анализ. Классификация и кластеризация, тематический анализ, анализ тональности. Аннотирование и реферирование. Ранжирование результатов поисковых запросов.

4.2 Структура дисциплины

Общая трудоемкость дисциплины составляет 10 зачетных единиц.

Программа дисциплины

	Раздел дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)					Кон тр.	Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации по семестрам
				Лек.	Сем.	Лаб.	Сам.	Контр.		
1	Базовые конструкции языка C, типы данных, структура программы.	1	1-4		4	12	28			
2	Алгоритмы сортировки	1	5-6		2	6	14	2	Контрольная работа	
3	Алгоритмы сжатия данных	1	7-8		2	6	14		Прием лабораторных работ	
4	Алгоритмы шифрования данных	1	9-11		6	6	21	2	Прием лабораторных работ	
5	Алгоритмы кодирования	1	12		2	2	7		Прием лабораторных работ	
	Технологии поиска. Типы индексов	1	13	2	2		7			
	Работа с матрицами, быстрое умножение.	1	14	2	2		7		Контрольная работа	
	Методы динамического программирования.	1	15	2	2		7			
6	Теория игр, критерии принятия решений	1	16-17	2	2	4	14		Зачет	
	Специализированные пакеты для математических расчетов	2	1-4		4	12	28			
7	Грамматики	2	5		4		7			
8	Методы визуализации данных (деревьев, многомерных пространств). Задачи о рюкзаке и о	2	6	2	2		7			

	графах.								
9	Методы работы с полуструктурированными данными	2	7-8	4	4		14	2	Контрольная работа
	Язык Javascript	2	9-10		8		14		
	Язык VBA	2	11		4		7		
	Скрипты в shell и cmd	2	12		4		7		Контрольная работа
	Архитектуры систем 1, 2, 3 звенные Распределенные объекты. Технологии промежуточного слоя. Методы обеспечения отказоустойчивости, перераспределения нагрузки.	2	13	2	2		6		
	Экспертные системы, генетические алгоритмы, нейронный сети, нечеткие множества	2	14	2	2		6		
	Обзор сетевых технологий и протоколов	2	15	2	2		7		
11	Обработка текстовых данных	2	16	2	2		6		Зачет
				22	60	46	228		Всего 360 часов

Примеры контрольного задания в 1 семестре

- Реализовать алгоритм сортировки со сложностью $n \log(n)$.
- Реализовать алгоритм сжатия RLE.
- Построить закрытый и открытый ключ RSA на основе двух заданных простых чисел.
- Подсчитать сумму четных чисел от -1000 до 1000 имеющих в записи цифру 5 или 7 и не делящихся на 7 и на 11

Пример зачетных вопросов в 1 семестре

- Почему алгоритм сжатия Хаффмана плохо сжимает зашифрованные тексты?
- Какие задачи решаются при помощи цифровой подписи?
- Что называет «естественным поведением» для сортировки?
- Какие индексы используют в OLAP и OLTP? Почему?
- Какая стратегия наиболее выгодна при игре в орлянку?

Примеры контрольного задания во 2 семестре

- Найти решение системы уравнений $x^3 - y^2 = 5$, $x + y = 3$ по модулю 7.
- Посчитать асимптотическое приближение функции $(x+5)/(x^3 - \sin(x))$ при x стремящимся к бесконечности.
- Подсчитать функцию Эйлера от 2500.
- Разложить на множители полином $x^6 + 2x^5 + 7x^4 - 9x^3 + x^2 - 2x$ по модулю 11
- Решить уравнение $\exp(3x) + \exp(x) = 7$
- Построить случайную матрицу размерности 30 с Гаусовым распределением элементов(1,5) и подсчитать ее характеристики.
- Найти решение уравнения $D[1,1](y)(x) + 2D(y)(x) = \sin(x) + 1$
- макросы в VBA. Сделать макрос, который во всех файла в поддиректории де-

- лает жирными все параграфы содержащие слово "безопасность".
- cmd (под windows). Вывести список файлов, которые есть в текущей директории и поддиректории сору.
- JavaScript. Сделать страницу со списком случайных чисел.
- JavaScript. Сделать калькулятор: два поля для ввода чисел и четыре кнопки для действий.
- JavaScript. Сделать окно с полем и кнопкой. В поле вводится число, наживается кнопка и в открывшемся новом окне начинается обратный отсчет с введенного числа.
- При достижении 0 окно закрывается.

Пример зачетных вопросов во 2 семестре

- Maple. Преобразование и упрощение символьных выражений. Работа с полиномами.
- Подходит ли Javascript для реализации менеджера файловой системы?
- Как влияет на стоимость сервера выбор между «толстым» и «тонким» клиентом для разработки КИС?
- Написать формулы расчета принятия решения для фразы «Если чай горячий и человек не очень хочет пить, то он ждет когда чай остынет».
- Почему на домашнем компьютере в домашней сети не следует ставить DHCP сервер?

5. Образовательные технологии:

Сочетание традиционных образовательных технологий в форме лекции с интерактивными семинарскими занятиями и компьютерными автоматизированными информационными технологиями при выполнении лабораторных работ и проведении контрольных мероприятий (зачетов и промежуточного тестирования).

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

Контроль качества подготовки осуществляется путем проверки теоретических знаний и практических навыков посредством:

- 1) промежуточных контрольных работ
- 2) зачетов в конце семестра
- 3) проверки и приема текущих семестровых заданий и лабораторных работ.

7. Учебно-методическое и информационное обеспечение дисциплины:

а) основная литература

Д.Кнут, Искусство программирования для ЭВМ в 3 т. М.: Мир, 1976

Шень А. Программирование: теоремы и задачи. — М.: МЦНМО, 1995

Б. Страуструп. Язык программирования C++, М.: "Бином", 1999

Г. Каца W3C XML:XPath от экспертов. Руководство по языку запросов

Дунаев В. Самоучитель JavaScript. 2-е изд.

Дьяконов В.П. Maple 9. 5/10 в математике, физике и образовании

б) дополнительная литература:

Мартынов Н. Н. Информатика: С для начинающих.
Дейтел Х.М., Дейтел П.Дж. Как программировать на С. Изд.4
Васильев А Н Maple 8. Самоучитель
Санжей Мишра, Алан Бьюли Секреты Oracle SQL
В.В. Корнеев, А.Ф. Гареев, С.В. Васютин, В.В. Райх. Базы данных. Интеллектуальная обработка информации.

в) программное обеспечение и Интернет-ресурсы

<http://citforum.ru/>

<http://www.emanual.ru/>

<http://www.intuit.ru/department/se/prmalgs/>

<http://www.citforum.ru/programming/theory/>

<http://www.compmodel.ru/>

8. Материально-техническое обеспечение дисциплины

При освоении дисциплины для выполнения лабораторных работ необходимы классы персональных компьютеров с набором базового программного обеспечения разработчика - системы программирования на языках С/С++, с возможностью многопользовательской работы, централизованного администрирования и доступа к информационным ресурсам.