

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«Томский государственный университет систем управления и радиоэлектроники»
(ТУСУР)

АЛГОРИТМЫ И АНАЛИЗ ИХ СЛОЖНОСТИ

Методические указания по самостоятельной и индивидуальной работе студентов всех форм обучения для специальности 010500
"Прикладная математика и информатика"

Томск-2010

Горитов А.Н.

Структуры и алгоритмы обработки данных в ЭВМ: методические указания по самостоятельной и индивидуальной работе студентов всех форм обучения по специальности "010500 – Прикладная математика и информатика" / А.Н. Горитов. – Томск: ТУСУР, 2010. – 8 с.

Методические указания разработаны в соответствии с решением кафедры автоматизированных систем управления

Составитель: д.т.н., профессор каф. АСУ А.Н. Горитов

Методические указания утверждены на заседании кафедры автоматизированных систем управлениям
28 августа 2010 г., протокол № 1

© ТУСУР, каф. АСУ, 2010

© Горитов А.Н. 2010

СОДЕРЖАНИЕ

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ	4
2 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ	4
2.1 Теоретический материал.....	4
2.2 Лабораторный практикум.....	5
2.3 Самостоятельная работа	6
2.4 Контрольные вопросы	6
3 УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ	7
3.1 Основная литература.....	7
3.2 Дополнительная литература.....	7
3.3 Учебно-методические пособия	8

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Дисциплина «Алгоритмы и анализ их сложности» читается в 9 семестре и предусматривает чтение лекций, проведение лабораторных занятий и получение различного рода консультаций.

Целью дисциплины «Алгоритмы и анализ их сложности» является изучение основных методов разработки алгоритмов обработки данных и анализ этих алгоритмов, взаимосвязь алгоритмов и структур данных.

Дисциплина «Алгоритмы и анализ их сложности» относится к профильному циклу вариативной части ООП. Успешное овладение дисциплиной предполагает предварительные знания по дисциплинам: «Математический анализ», «Вычислительная математика», «Дискретная математика» в объеме, предусмотренном специальностью «Прикладная математика и информатика», а также навыки программирования на языках высокого уровня. Знания и навыки, полученные при изучении этой дисциплины, используются в последующих дисциплинах: «Методы и алгоритмы параллельного программирования», «Обработка изображения/Распознавание образов и сцен», «Объектно-ориентированные языки и системы программирования» и при подготовке выпускной квалификационной работы.

В результате изучения дисциплины студент должен:

- иметь представление о некоторых методах разработки алгоритмов, математических методах анализа алгоритмов, классификации алгоритмических задач по сложности, сводимости алгоритмических задач к известным задачам определенного класса сложности.
- знать и использовать основные алгоритмы решения классических задач информатики;
- иметь опыт работы с алгоритмическими языками программирования.

2 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

2.1 Теоретический материал

Тема 1. Алгоритмы и их сложность

Алгоритмы и их классификация. Вычислительная сложность алгоритмов.

Тема 2. NP-полные и труднорешаемые задачи

Массовая и индивидуальная задачи. Сложность алгоритма. Полиномиальные алгоритмы и класс P. Недетерминированные алгоритмы и класс NP. Полиномиальная преобразуемость задач. NP-трудные и NP-полные задачи.

Тема 3. Методы разработки алгоритмов

Метод декомпозиции. Пример.

Поиск с возвратом. Общий алгоритм. Пример.

Метод ветвей и границ. Применение метода при решении задачи о рюкзаке, задаче о назначениях и задаче коммивояжера.

Динамическое программирование. Пример. Нисходящее и восходящее динамическое программирование. Задача определения наиболее длинной общей подпоследовательности. Задача построения оптимального бинарного дерева поиска.

Тема 4. Поиск подстрок

Задача поиска подстрок. Простейший алгоритм. Алгоритм Кнута-Морриса-Пратта. Алгоритм Бойера-Мура. Алгоритм Рабина-Карпа.

Тема 5. Алгоритмы на графах

Графы: определения и примеры. Представления графов: матрица инцидентностей, матрица смежности, список пар, списки смежности.

Фундаментальные алгоритмы на графах: Поиск в графе. Поиск в ширину. Поиск в глубину.

Остовные деревья графа. Связные компоненты. Построение и свойства остовных деревьев при поиске в глубину и в ширину.

Эйлеров путь в графе. Алгоритм построения Эйлера пути.

Циклы: Фундаментальное множество циклов графа. Алгоритм отыскания фундаментального множества циклов в графе.

Минимальное остовное дерево. Алгоритм Прима. Алгоритм Крускала.

Гамильтонов путь в графе. Нахождение Гамильтонова пути в графе с помощью алгоритма с возвратом.

Тема 6. Кратчайшие пути в графе

Кратчайшие пути в графе. Кратчайшие пути от фиксированной вершины. Алгоритм Форда-Беллмана. Случай неотрицательных весов: алгоритм Дейкстры. Кратчайшие пути в бесконтурном графе. Топологическая сортировка. Кратчайшие пути между всеми парами вершин. Матрица смежности, матрица достижимости и транзитивное замыкание отношения, алгоритм Уоршалла. Алгоритм Флойда-Уоршалла вычисления расстояний между всеми парами вершин, одновременное построение путей.

Тема 7. Задачи о потоках

Задача о максимальном потоке. Введение. Транспортные сети и потоки. Алгоритм Форда-Фалкерсона. Алгоритм Эдмондса-Карпа. Алгоритм Диница.

Задача о потоке минимальной стоимости.

Тема 8. Двудольные графы

Паросочетание. Алгоритм определения максимального паросочетания. Задача о полном паросочетании.

Тема 9. Алгоритмы с возвратом

Гамильтонов путь. Клики графа.

2.2 Лабораторный практикум

В результате выполнения лабораторных работ студенты учатся использовать на практике изученный теоретический материал.

Темы лабораторных работ:

1. Метод ветвей и границ.

2. Динамическое программирование.
3. Поиск подстрок.
4. Алгоритмы на графах.
5. Задачи о потоках.

2.3 Самостоятельная работа

Самостоятельная работа рассматривается как вид учебного труда студента, позволяющего целенаправленно формировать и развивать его самостоятельность как личное качество при выполнении лабораторных работ и проработке дополнительного учебного материала.

Самостоятельная работа организуется в двух формах:

- аудиторная работа: на лабораторных занятиях при выполнении лабораторных работ;
- внеаудиторная работа: проработка лекций с целью освоения всех разделов дисциплины, подготовка к лабораторным занятиям, оформление отчетов по лабораторным работам, самостоятельное изучение дополнительных тем дисциплины.

Темы для самостоятельного изучения:

- 1) Стеки и очереди.
- 2) Списки.
- 3) Деревья.
- 4) Бинарные деревья поиска.

2.4 Контрольные вопросы

Стеки, очереди

1. Что такое стековый принцип сохранения элементов?
2. Какие основные операции реализуются для стеков?
3. Какие шаги выполняются при добавлении элемента в стек-массив?
4. Какие шаги выполняются при удалении элемента из стека-массива?
5. Какие особые ситуации возможны при реализации стека с помощью массива?

Списки

1. В чем состоит отличие списковых структур от стека и очереди?
2. Что включает в себя стандартный набор операций со списком?
3. В чем состоит простейший способ реализации списка с помощью массива?
4. Как выполняется вставка элемента при простейшей реализации списка на базе массива?
5. Как выполняется удаление элемента при простейшей реализации списка на базе массива?

Деревья

1. В чем состоит основное отличие древовидных структур от списковых?
2. Что такое строго бинарное дерево?
3. Что такое полное бинарное дерево?
4. Что такое почти полное бинарное дерево?
5. Основные операции, выполняемые над бинарным деревом?

Двоичные деревья поиска

1. Какое дерево называется деревом поиска?
2. В чем состоит практическая важность использования деревьев поиска?
3. Какие преимущества имеет использование деревьев поиска для хранения упорядоченных данных по сравнению с массивами и списками?
4. Почему наивысшая эффективность поиска достигается у идеально сбалансированных деревьев?
5. Как находится максимально возможное число шагов при поиске в идеально сбалансированном дереве?

Сбалансированные деревья поиска

1. Какие проблемы возникают при использовании деревьев поиска?
2. Как влияет на структуру дерева поиска разный порядок поступления одних и тех же входных ключей?
3. Почему при построении дерева поиска важно управлять его структурой?
4. Какие деревья называются AVL-сбалансированными?
5. Как связаны понятия “идеально сбалансированное дерево” и “AVL-сбалансированное дерево”?

Графы

1. Какие существуют разновидности графов?
2. Какие способы можно использовать для представления графов как структур данных?
3. Что такое матрица смежности и как она описывается?
4. Какие структуры данных необходимы для реализации списков смежности?
5. Какие основные задачи возникают при использовании графов?

3 УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

3.1 Основная литература

1. Горитов А.Н. Основы структур и алгоритмов обработки данных: Учебное пособие. – Томск: ТУСУР, 2007. – 229 с. (50 экз.)

3.2 Дополнительная литература

2. Макконелл Дж. Основы современных алгоритмов. 2-е дополненное издание. – Москва: Техносфера, 2004. – 368 с. (13 экз.)
3. Иванов Б.Н. Дискретная математика. Алгоритмы и программы: Учебное пособие. – М.: Лаборатория базовых знаний, 2003. – 288 с. (50 экз.)
4. Окулов С. М. Программирование в алгоритмах. – 2-е изд., доп. – М.: БИНОМ. Лаборатория знаний, 2006. – 384 с. (30 экз.)
5. Новиков Ф.А. Дискретная математика для программистов: Учебник для вузов. – СПб: Питер, 2002. – 302 с. (19 экз.)
6. Андерсон Д.А. Дискретная математика и комбинаторика. – М.: Издательский дом "Вильямс", 2004. – 960 с. (10 экз.)
7. Липский В. Комбинаторика для программистов. – М.: Мир, 1989. – 213 с. (9 экз.)
8. Вирт Н. Алгоритмы и структуры данных. – М.: Мир, 1989. – 360 с. (50 экз.)
9. Гагарина Л.Г., Колдаев В.Д. Алгоритмы и структуры данных. – М: Финансы и статистика; ИНФРА-М, 2009. – 304 с.

3.3 Учебно-методические пособия

10. Горитов А.Н. Основы структур и алгоритмов обработки данных: лабораторный практикум по дисциплине "Структуры и алгоритмы обработки данных" для студентов специальности 230105 Программное обеспечение вычислительной техники и автоматизированных систем. – Томск: ТУСУР, 2007. – 41 с. (48 экз.).