

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное  
учреждение высшего образования

«Южно-Уральский государственный университет  
(национальный исследовательский университет)»

Высшая школа электроники и компьютерных наук

Кафедра системного программирования

**Структуры и алгоритмы обработки данных**

Методические указания по выполнению практических работ студентов  
всех форм обучения для направлений  
подготовки 09.03.04 «Программная инженерия», профиль  
«Инженерия информационных и интеллектуальных систем»

## Содержание

|  |    |
|--|----|
| 1. Цели и задачи дисциплины .....                                    | 3  |
| 2. Методы и формы организации обучения .....                         | 3  |
| 3. Место дисциплины в структуре ООП .....                            | 3  |
| 4. Практические работы.....  | 4  |
| 5. Учебно-методическое и информационное обеспечение дисциплины ..... | 11 |

## **1. Цели и задачи дисциплины**

Целью изучения дисциплины «Структуры и алгоритмы обработки данных» является изучение применяемых в программировании (и информатике) структур данных, их спецификации и реализации, а также алгоритмов обработки данных и анализ этих алгоритмов, взаимосвязь алгоритмов и структур.

В результате изучения дисциплины студент должен:

- а) иметь представление об основных тенденциях в создании структур данных, методах оптимального использования памяти и времени для обработки структур данных и управления процессами обработки данных;
- б) знать и использовать различные (динамические и статистические) структуры данных в соответствии с запросами алгоритмов;
- в) создавать списковые и древообразные структуры и управлять организацией этих структур (изменение списков и деревьев посредством включения исключения, замены элементов структур) знать, использовать оптимальные методы поиска и сортировки данных;
- г) знать и использовать основные алгоритмы решения классических задач информатики;
- д) иметь представление о математических методах анализа алгоритмов; классификации алгоритмических задач по сложности, сводимости алгоритмических задач к известным задачам определенного класса сложности;
- е) иметь опыт работы с алгоритмическими языками программирования.

## **2. Методы и формы организации обучения**

Для успешного освоения дисциплины применяются различные образовательные технологии, которые обеспечивают достижение планируемых результатов обучения согласно основной образовательной программе, с учетом требований к объему занятий в интерактивной форме.

Для контроля освоения компетенций используются следующие формы контроля: защита практических заданий, опрос по изучаемым разделам дисциплины, тесты.

## **3. Место дисциплины в структуре ООП**

Дисциплина «Структуры и алгоритмы обработки данных» относится к вариативной части профессионального цикла ООП. Успешное овладение дисциплиной предполагает предварительные знания таких дисциплин как «Математический анализ», «Дискретная математика», «Программирование», «Математическая логика и теория алгоритмов» в объеме, предусмотренном направлением 09.03.01 – «Информатика и вычислительная техника».

## **4. Практические работы**

Практические работы дисциплины "Структуры и алгоритмы обработки данных" позволяет получить практические навыки использования изучаемых структур данных и эффективных алгоритмов решения различных задач.

Результаты выполнения практических работ представляются в виде отчета, который состоит из следующих пунктов:

1. Тема
2. Цель работы
3. Задание
4. Алгоритм решения задачи
5. Текст программы
6. Результаты работы программы
7. Выводы

### **Порядок выполнения практических работ**

- 1) изучить теоретический материал по теме практической работы;
- 2) составить программу на одном из алгоритмических языков программирования для заданного варианта задания;
- 3) выполнить отладку составленной программы и показать преподавателю;
- 4) составить и защитить отчет по практической работе.

### **4.1 Множества**

Цель практической работы – изучение основных способов использования множеств.

#### **Теория**

Для выполнения практической работы необходимо ознакомиться со следующими разделами дисциплины:

Множества. Операции над множествами. Представление в памяти.

#### **Методические рекомендации**

При выполнении практической работы рекомендуется следующая последовательность:

1. Изучить теорию по теме «Множества».
2. Получить задание на выполнение практической работы.
3. Используя один из алгоритмических языков выполнить задание на практическую работу.
4. Подготовить отчет по практической работе.
5. Защитить выполненную практическую работу.

#### **Контрольные вопросы**

1. Как задается тип данных «Множество»?

2. Как множества представляются в памяти ЭВМ.
3. Операции, выполняемые над множествами.

#### **4.2 Стеки, очереди**

Целью практической работы является изучение способов представления стеков и очередей в оперативной памяти ЭВМ и реализация основных операций работы с ними.

##### **Теория**

Для выполнения практической работы необходимо ознакомиться со следующими разделами дисциплины:

Стек, очередь и дек. Стек, очередь и дек как абстрактные типы данных. Представление и реализация.

Примеры алгоритмов, использующих стек, очередь, дек.

##### **Методические рекомендации**

При выполнении практической работы необходимо дать описание абстрактного типа данных «Стек» или «Очередь». Реализация основных операции АДТ выполняется в виде процедур или функций. Все данные в процедуры или функции передаются через формальные параметры.

Перед завершением выполнения программы необходимо убедиться, что вся динамическая память, использованная для создания динамических структур данных, освобождена. В противном случае обеспечить ее полное освобождение.

##### **Контрольные вопросы**

1. Понятие стека. Операции, выполняемые над стеком.
2. Понятие очереди. Операции, выполняемые над очередью.
3. Понятие дека. Операции, выполняемые над деком.
4. Представление стека с помощью массива. Выполнение основных операций.
5. Представление очереди с помощью массива. Выполнение основных операций.

#### **4.3 Связанные списки**

Целью практической работы является изучение основных способов представления связанных списков в оперативной памяти ЭВМ и реализация основных операций работы со списками.

##### **Теория**

Для выполнения практической работы необходимо ознакомиться со следующими разделами дисциплины:

Односвязные, двусвязные, кольцевые списки и операции над ними. Представление и реализация (непрерывная, ссылочная в связанной памяти).

#### Методические рекомендации

При выполнении практической работы необходимо дать описание абстрактного типа данных «Список». Реализация основных операции АТД выполняется в виде процедур или функций. Все данные в процедуры или функции передаются через формальные параметры.

Перед завершением выполнения программы необходимо убедиться, что вся динамическая память, использованная для создания динамических структур данных, освобождена. В противном случае обеспечить ее полное освобождение.

#### Контрольные вопросы

1. Отличие динамических структур от статических и полустатических.
2. Основные операции, выполняемые над списками.
3. Что такое кольцевой список.

### **4.4 Бинарные деревья**

Цель практической работы – изучение основных способов представления бинарных деревьев в оперативной памяти ЭВМ и практическая реализация алгоритма работы с деревьями.

#### Теория

Для выполнения практической работы необходимо ознакомиться со следующими разделами дисциплины:

Обходы бинарных деревьев: рекурсивные и не рекурсивные алгоритмы. Обходы дерева или леса.

Представления и реализации бинарных деревьев: ссылочная реализация в связанной памяти, ссылочная реализация ограниченного бинарного дерева на базе вектора.

#### Методические рекомендации

Необходимо дать описание «дерева» как абстрактного типа данных. Затем реализовать основные операции АТД «Дерево» в виде процедур или функций, при этом данные в процедуры или функции должны передаваться через формальные параметры.

Во всех заданиях предполагается вывод сформированного дерева на экран.

Перед завершением выполнения программы необходимо убедиться, что вся динамическая память, использованная для создания динамических

структур данных, освобождена. В противном случае обеспечить ее полное освобождение.

#### Контрольные вопросы

1. Что такое «полное бинарное дерево»?
2. Что такое «почти полное бинарное дерево»?
3. Сколько узлов в полном бинарном дереве высоты  $h$ ?
4. Какова минимальная высота бинарного дерева с  $n$  узлами?
5. Какие операции определены на АТД «Бинарное дерево поиска»?

### 4.5 Сортировка

Цель практической работы – изучение основных способов сортировки данных.

#### Теория

Для выполнения практической работы необходимо ознакомиться со следующими разделами дисциплины:

Стратегии внутренней сортировки. Сортировка выборкой: метод простого выбора, турнирная сортировка, пирамидальная сортировка, анализ сложности алгоритмов. Сортировка включением: метод простых вставок, метод вставки с убывающим шагом, анализ сложности алгоритмов. Обменные сортировки: сортировка пузырьком, быстрая сортировка, анализ сложности алгоритмов. Сортировка распределением: двоичная быстрая сортировка, цифровая распределяющая сортировка, блочная сортировка, сортировка подсчетом, анализ сложности алгоритмов. Сортировка слиянием.

Нижняя граница сложности задачи сортировки.

#### Методические рекомендации

При выполнении практической работы рекомендуется следующая последовательность:

1. Изучить теорию по теме «Сортировка».
2. Получить задание на выполнение практической работы.
3. Используя один из алгоритмических языков выполнить задание на практическую работу.
4. Подготовить отчет по практической работе.
5. Защитить выполненную практическую работу.

#### Контрольные вопросы

1. В чем заключается сортировка методом вставки с убывающим шагом?
2. В чем заключается быстрая сортировка?
3. В чем заключается двоичная быстрая сортировка?
4. В чем заключается цифровая распределяющая сортировка?

## 5. В чем заключается сортировка слиянием?

### **4.6 Поиск подстрок**

Цель практической работы – изучение эффективных алгоритмов поиска подстрок.

#### Теория

Для выполнения практической работы необходимо ознакомиться со следующими разделами дисциплины:

Задача поиска подстроки. Алгоритм Кнута-Морриса-Пратта. Алгоритм Боуера-Мура. Алгоритм Рабина-Карпа.

#### Методические рекомендации

При выполнении практической работы рекомендуется следующая последовательность:

1. Изучить теоретический материал по алгоритмам поиска подстрок.
2. Разобрать работу этих алгоритмов на примерах.
3. Получить задание на выполнение практической работы.
4. Используя один из алгоритмических языков выполнить задание на практическую работу.
5. Подготовить отчет по практической работе.
6. Защитить выполненную практическую работу.

#### Контрольные вопросы

1. Что такое собственный суффикс и собственный префикс образца?
2. Как строится префикс-функция в алгоритме Кнута-Морриса-Пратта?
3. Что такое «эвристика безопасного суффикса»?
4. Что такое «эвристика стоп-символа»?
5. За счет чего в алгоритме Рабина-Карпа удается получить высокое быстроедействие?

### **4.7 Фундаментальные алгоритмы на графах**

Цель практической работы – изучение основных способов представления графов в оперативной памяти ЭВМ и практическая реализация алгоритма работы с графами.

#### Теория

Для выполнения практической работы необходимо ознакомиться со следующими разделами теории графов:

Графы: определения и примеры. Упорядоченный граф. Представления графов: матрица инцидентий, матрица смежности, список пар, списки смежности.

Поиск в графе: Основные методы обработки графов. Поиск в ширину. Поиск в глубину.



Связные компоненты: Определение компонент связности. Топологическая сортировка.

Двусвязность: Точки сочленения и их свойства. Алгоритм выделения компонент двусвязности графа.

Эйлеров путь в графе. Алгоритм построения Эйлерова пути.

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

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

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

Остовные деревья минимального веса: Минимальное остовное дерево. Алгоритм Прима. Алгоритм Крускала.

#### Методические рекомендации

Необходимо дать описание «Графа» как абстрактного типа данных. Затем реализовать основные операции АТД «Граф» в виде процедур или функций, при этом данные в процедуры или функции должны передаваться через формальные параметры.

Перед завершением выполнения программы необходимо убедиться, что вся динамическая память, использованная для создания динамических структур данных, освобождена. В противном случае обеспечить ее полное освобождение.

#### Контрольные вопросы

1. Опишите, распространенные способы представления графов в памяти ЭВМ.
2. Что такое эйлеров путь.
3. Необходимое и достаточное условие существования эйлерова пути.
4. Что такое гамильтонов путь.
5. Что такое стягивающее дерево.

### **4.8 Кратчайшие пути в графе**

Цель практической работы – получение навыком при решении задач нахождения кратчайшего пути в графе.

#### Теория

Для выполнения практической работы необходимо ознакомиться со следующими разделами теории графов:

Кратчайшие пути в графе от фиксированной вершины. Алгоритм Форда-Беллмана. Алгоритм Дейкстры. Кратчайшие пути в бесконтурном графе.

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.

### Методические рекомендации

Необходимо ввести абстрактный тип данных «Граф». Изучить методы нахождения кратчайших путей в графе. Реализовать основные операции АТД «Граф» в виде процедур или функций, при этом данные в процедуры или функции должны передаваться через формальные параметры.

Перед завершением выполнения программы необходимо убедиться, что вся динамическая память, использованная для создания динамических структур данных, освобождена. В противном случае обеспечить ее полное освобождение.

### Контрольные вопросы

1. Поясните, в чем состоит алгоритм Форда-Беллмана.
2. Поясните, в чем состоит алгоритм Дейкстры.
3. На каком принципе строится алгоритм нахождения кратчайших путей в бесконтурном графе?
4. Что такое «транзитивное замыкание».
5. Поясните, в чем состоит алгоритм Уоршалла.

## **5. Учебно-методическое и информационное обеспечение дисциплины**

### **5.1 Основная литература**

1. Гагарина Л.Г., Колдаев В.Д. Алгоритмы и структуры данных: Учебное пособие. – М.: Финансы и статистика; ИНФРА-М, 2009. – 304 с.

### **5.2 Дополнительная литература**

2. Макконелл Дж. Основы современных алгоритмов. 2-е дополненное издание. – Москва: Техносфера, 2004. – 368 с.
3. Иванов Б.Н. Дискретная математика. Алгоритмы и программы: Учебное пособие. – М.: Лаборатория базовых знаний, 2003. – 288 с.
4. Окулов С. М. Программирование в алгоритмах. – 2-е изд., доп. – М.: БИНОМ. Лаборатория знаний, 2006. – 384 с.
5. Новиков Ф.А. Дискретная математика для программистов: Учебник для вузов. – СПб: Питер, 2002. – 302 с.
6. Андерсон Д.А. Дискретная математика и комбинаторика. – М.: Издательский дом "Вильямс", 2004. – 960 с.
7. Вирт Н. Алгоритмы и структуры данных. – М.: Мир, 1989. – 360 с.
8. Бежанова М.М. Практическое программирование. Структуры данных и алгоритмы. – М.: Логос, 2001. – 224 с.