

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Галунин Сергей Александрович  
Должность: проректор по учебной работе  
Дата подписания: 23.12.2025 13:42:26  
Уникальный программный ключ:  
08ef34338325bdb0ac5a47baa5472ce36cc3fc3b

Приложение к ОПОП  
«Разработка программно-  
информационных систем»



**СПбГЭТУ «ЛЭТИ»**  
ПЕРВЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ

МИНОБРНАУКИ РОССИИ

федеральное государственное автономное образовательное учреждение высшего образования  
**«Санкт-Петербургский государственный электротехнический университет  
«ЛЭТИ» им. В.И.Ульянова (Ленина)»  
(СПбГЭТУ «ЛЭТИ»)**

---

**РАБОЧАЯ ПРОГРАММА**

дисциплины

**«АЛГОРИТМЫ И СТРУКТУРЫ ДАННЫХ»**

для подготовки бакалавров

по направлению

09.03.04 «Программная инженерия»

по профилю

**«Разработка программно-информационных систем»**

Санкт-Петербург

2025

## **ЛИСТ СОГЛАСОВАНИЯ**

Разработчики:

доцент, к.т.н., доцент Заславский М.М.

ассистент Иванов Д.В.

Рабочая программа рассмотрена и одобрена на заседании кафедры МОЭВМ  
20.01.2025, протокол № 1

Рабочая программа рассмотрена и одобрена учебно-методической комиссией  
ФКТИ, 28.01.2025, протокол № 1

Согласовано в ИС ИОТ

Начальник ОМОЛА Загороднюк О.В.

## 1 СТРУКТУРА ДИСЦИПЛИНЫ

Обеспечивающий факультет	ФКТИ
Обеспечивающая кафедра	МОЭВМ

Общая трудоемкость (ЗЕТ)	4
Курс	2
Семестр	3

### **Виды занятий**

Лекции (академ. часов)	34
Лабораторные занятия (академ. часов)	34
Иная контактная работа (академ. часов)	3
Все контактные часы (академ. часов)	71
Самостоятельная работа, включая часы на контроль (академ. часов)	73
Всего (академ. часов)	144

### **Вид промежуточной аттестации**

Экзамен (курс)	2
Курсовая работа (курс)	2

## **2 АННОТАЦИЯ ДИСЦИПЛИНЫ**

### **«АЛГОРИТМЫ И СТРУКТУРЫ ДАННЫХ»**

Дисциплина предназначена для изучения и освоения базовых понятий, методов и приёмов разработки алгоритмов и программ с использованием структур данных (с реализацией на языке программирования C++ в рамках парадигм процедурного, модульного и объектно-ориентированного программирования) и охватывает следующие основные темы. Рекурсия как метод разработки алгоритмов, программирование рекурсивных алгоритмов. Абстрактный тип данных: спецификация, представление, реализация. Линейные структуры данных: стек, очередь, дек. Нелинейные структуры данных: иерархические списки, деревья и леса, бинарные деревья. Обходы деревьев. Быстрый поиск: бинарный поиск, хеширование; бинарные деревья поиска (БДП), случайные БДП, оптимальные БДП, сбалансированные по высоте (АВЛ) и рандомизированные БДП (случайные БДП и пирамиды поиска). Задачи сортировки; внутренняя и внешняя сортировки; алгоритмы сортировки; оптимальная сортировка; порядковые статистики; анализ сложности и эффективности алгоритмов поиска и сортировки.

## **SUBJECT SUMMARY**

### **«ALGORITHMS AND DATA STRUCTURES»**

Discipline is dedicated to the mastering of base conceptions and methods of algorithms and software developing with using of data structures (with implementation on programming language C++ within the scope of procedural, modular, and object-oriented programming) and includes following basic subjects. Recursion as algorithms developing method, programming of recursive algorithms. Abstract data types: specification, announcement, implementation. Linear data structures: stack, queue, deque. Non-linear data structures: hierarchical lists, trees and forests, binary

trees. Trees traversals. Quick search: binary search, hashing; binary search trees (BSTs), random BSTs, optimal BSTs, height-balanced (AVL) and randomized BSTs (random BSTs and search heaps). Sorting tasks; internal and external sorting; sorting algorithms; optimal sorting; order statistics; complexity and efficiency analysis of search and sorting algorithms.

## **3 ОБЩИЕ ПОЛОЖЕНИЯ**

### **3.1 Цели и задачи дисциплины**

1. Целью дисциплины является изучение применяемых в информатике и программировании структур данных, их спецификации и реализации, алгоритмов обработки данных и анализа этих алгоритмов, взаимосвязи алгоритмов и структур данных, и получение практических умений и навыков их применения к решению задач профессиональной деятельности.
2. Задачи изучения дисциплины: научиться самостоятельно составлять, программировать и тестировать рекурсивные алгоритмы; реализовывать типовые алгоритмы, структуры данных и их модификации на рабочем языке программирования.
3. Приобретение знаний базовых понятий, лежащих в основе процесса разработки алгоритмов и структур данных; приемов программирования рекурсивных алгоритмов, модели абстрактного типа данных (спецификации, представления, реализации), как основы конструирования и использования сложных (динамических) структур данных; конструкций и приёмов программирования на языке C++, позволяющего реализовать знание алгоритмов и структур данных (в различных парадигмах программирования: в процедурном, модульном, объектно-ориентированном программировании); различных моделей, предполагающих знание и реализацию динамических структур данных (списков, стеков и очередей, деревьев и лесов); основных видов алгоритмов быстрого поиска и упорядочения данных, используемых в них структурах данных и общих схем решения задач.
4. Формирование умений разрабатывать алгоритмы и программы решения задач, используя изложенные в курсе общие схемы, методы и приемы построения алгоритмов, выбирая подходящие структуры данных для представления инфор-

мационных объектов; умения определять основные характеристики сложности разработанных алгоритмов и проводить компьютерные испытания для исследования их эффективности.

5. Освоение: навыков использования средств рабочего языка программирования в объеме, необходимом для реализации изучаемых алгоритмов и структур данных; навыков постановки задач, разработки алгоритмов и программ, их тестирования и отладки, экспериментального (компьютерного) испытания их эффективности.

### **3.2 Место дисциплины в структуре ОПОП**

Дисциплина изучается на основе ранее освоенных дисциплин учебного плана:

1. «Информатика»
2. «Программирование»
3. «Информационные технологии»

и обеспечивает изучение последующих дисциплин:

1. «Математическая логика и теория алгоритмов»
2. «Низкоуровневое программирование»
3. «Операционные системы»
4. «Построение и анализ алгоритмов»
5. «Тестирование программного обеспечения»
6. «Web-технологии»
7. «Базы данных»
8. «Компьютерная графика»
9. «Основы бэкенд-разработки на Python ч.1»
10. «Основы фронтенд-разработки ч.1»

11. «Параллельные алгоритмы»
12. «Вариационное исчисление»
13. «Введение в нереляционные системы управления базами данных»
14. «Основы промышленной разработки программного обеспечения»
15. «Основы фронтенд-разработки ч.2»
16. «Распределенные алгоритмы»
17. «Основы компьютерного зрения»
18. «Проектирование человеко-машинного интерфейса»
19. «Производственная практика (научно-исследовательская работа)»
20. «Инженерный документооборот»
21. «Интеллектуальные системы»
22. «Качество и метрология программного обеспечения»
23. «Разработка приложений для мобильных платформ»
24. «Теория принятия решений»
25. «Цифровая обработка изображений»



### 3.3 Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

В результате освоения образовательной программы обучающийся должен достичь следующие результаты обучения по дисциплине:

Код компетенции/ индикатора компетенции	Наименование компетенции/индикатора компетенции
ПК-3	Способен овладевать навыками моделирования, анализа и использования формальных методов конструирования программного обеспечения
<i>ПК-3.1</i>	<i>Знает основы моделирования и формальные методы конструирования программного обеспечения</i>
<i>ПК-3.2</i>	<i>Умеет использовать формальные методы конструирования программного обеспечения</i>
<i>ПК-3.3</i>	<i>Владеет методами формализации и моделирования программного обеспечения</i>

## 4 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

### 4.1 Содержание разделов дисциплины

#### 4.1.1 Наименование тем и часы на все виды нагрузки

№ п/п	Наименование темы дисциплины	Лек, ач	Лаб, ач	ИКР, ач	СР, ач
1	Введение	1			
2	Тема 1. Программирование рекурсивных алгоритмов	6	6		6
3	Тема 2. Рекурсивная обработка иерархических списков	6	6		6
4	Тема 3. Линейные структуры данных	6	6		6
5	Тема 4. Деревья и леса	6	6		6
6	Тема 5. Сортировка	4	6		6
7	Тема 6. Быстрый поиск	4	4		8
8	Заключение	1		3	35
	Итого, ач	34	34	3	73
	Из них ач на контроль	0	0	0	35
	Общая трудоемкость освоения, ач/зе	144/4			

#### 4.1.2 Содержание

№ п/п	Наименование темы дисциплины	Содержание
1	Введение	Место дисциплины в ООП. Рабочая программа дисциплины. Материал, выносимый на аттестацию. Формы проведения аттестации
2	Тема 1. Программирование рекурсивных алгоритмов	Рекурсивные определения и рекурсивные функции. Рекурсивные алгоритмы. Выполнение рекурсивных алгоритмов: подготовка трассы, стек (магазин). Анализ рекурсивных алгоритмов. Рекуррентные уравнения. Соотношение время-память для рекурсивных алгоритмов. Рекурсивные функции в языке C++. Приемы рекурсивного программирования (нисходящая и восходящая рекурсия, накапливающие параметры). Примеры: простая рекурсия, программы с несколькими рекурсивными вызовами, косвенная рекурсия (взаимно-рекурсивные подпрограммы). Рекурсивный алгоритм и рекурсивный вычислительный процесс. Соотношение рекурсивных и итеративных программ.

№ п/п	Наименование темы дисциплины	Содержание
3	Тема 2. Рекурсивная обработка иерархических списков	<p>Структуры данных и алгоритмы. Линейные списки как абстрактный тип данных (обзор материала, изученного ранее в предшествующем курсе программирования). Рекурсивное определение и функциональная спецификация линейных списков. Рекурсивное определение и функциональная спецификация иерархических (нелинейных) списков. Базовые функции (индикаторы, селекторы, конструкторы). Размеченное объединение множеств и реализация в языках программирования высокого уровня: записи с вариантами (Паскаль) и объединения (C++). Представление списков и реализация базовых функций на языке C++. Элементы функционального стиля при программировании рекурсивной обработки списков на языке C++. Примеры использования нелинейных списков: дифференцирование символических выражений, действия с полиномами многих переменных.</p>
4	Тема 3. Линейные структуры данных	<p>Разреженные матрицы и ортогональные списки. Аналогии: Плотное представление полинома массивом и разреженное представление линейным списком. Представление матриц различной (регулярной) структуры. Представление разреженных матриц ортогональными списками. Ссылочное представление в динамической памяти. Ссылочное представление разреженной матрицы на базе вектора (метод Д.Кнута). Примеры операций над разреженными матрицами. Стек, очередь и дек как последовательности с ограниченными наборами операций (доступа). Стек, очередь и дек как абстрактный тип данных (АТД). Функциональные спецификации и аксиомы. Представление и реализация (непрерывная, ссылочная в связанной памяти и на базе вектора). Реализация АТД стек и очередь средствами C++ в объектно-ориентированном стиле. Примеры алгоритмов, использующих стек, очередь, дек.</p>

№ п/п	Наименование темы дисциплины	Содержание
5	Тема 4. Деревья и леса	<p>Определение дерева, леса, бинарного дерева. Графическое и текстовое (скобочное) представление леса. Спецификация дерева, леса, бинарного дерева: базовые функции и аксиомы. Естественное соответствие бинарного дерева и леса. Обходы бинарных деревьев: рекурсивные и не рекурсивные алгоритмы. Обходы дерева (леса). Бинарные деревья с размеченными листьями (комбинации). Соответствие комбинаций и иерархических списков. Соответствие комбинаций и бинарных деревьев. Представления и реализации бинарных деревьев: ссылочная реализация в связанной памяти, ссылочная реализация ограниченного бинарного дерева на базе вектора. Реализация АТД бинарного дерева средствами C++ в объектно-ориентированном стиле. Пример использования бинарных деревьев в задаче упаковки сообщений. Префиксные коды и бинарные деревья. Метод кодирования Фано-Шеннона. Критерий оптимальности кода. Алгоритм кодирования по Хаффману (построение дерева, кодирование и декодирование). Реализация алгоритмов Хаффмана и Фано-Шеннона на C++. Доказательство оптимальности кода Хаффмана. Неравенство Крафта. Теорема кодирования в отсутствие шума (энтропийная оценка средней длины кода). Динамическое кодирование и декодирование по Хаффману.</p>
6	Тема 5. Сортировка	<p>Поиск и другие операции над таблицами. Последовательный и бинарный поиск. Бинарные деревья поиска (БДП). Идеально сбалансированные бинарные деревья. Случайные бинарные деревья поиска: определение и основные операции. Подсчет числа структурно различных бинарных деревьев с заданным числом узлов. Среднее время поиска в случайных деревьях. Операции «вращения» в БДП. Вставка в корень в случайном БДП. Случайные БДП с рандомизацией. Рандомизированные бинарные деревья поиска (пирамиды поиска – Treaps, Дерамиды). Сбалансированные по высоте бинарные деревья (АВЛ-деревья). Включение в АВЛ-дерево. Исключение из АВЛ-дерева. Оценка сложности в худшем случае: деревья Фибоначчи. Реализация упорядоченных линейных списков на базе БДП (АВЛ-деревьев или рандомизированных пирамид поиска). Операции поиска, вставки и удаления элементов; операции сцепления и расщепления списков. Метод поиска с использованием функции расстановки (хеширование). Разрешение коллизий: метод внутренних и внешних цепочек, метод открытой адресации. Коэффициент загрузки (заполнения), оценки сложности. Выбор функции расстановки.</p>

№ п/п	Наименование темы дисциплины	Содержание
7	Тема 6. Быстрый поиск	Поиск и другие операции над таблицами. Последовательный и бинарный поиск. Бинарные деревья поиска (БДП). Идеально сбалансированные бинарные деревья. Случайные бинарные деревья поиска: определение и основные операции. Подсчет числа структурно различных бинарных деревьев с заданным числом узлов. Среднее время поиска в случайных деревьях. Операции «вращения» в БДП. Вставка в корень в случайном БДП. Случайные БДП с рандомизацией. Рандомизированные бинарные деревья поиска (пирамиды поиска – Treaps, Дерамиды). Сбалансированные по высоте бинарные деревья (АВЛ-деревья). Включение в АВЛ-дерево. Исключение из АВЛ-дерева. Оценка сложности в худшем случае: деревья Фибоначчи. Реализация упорядоченных линейных списков на базе БДП (АВЛ-деревьев или рандомизированных пирамид поиска). Операции поиска, вставки и удаления элементов; операции сцепления и расщепления списков. Метод поиска с использованием функции расстановки (хеширование). Разрешение коллизий: метод внутренних и внешних цепочек, метод открытой адресации. Коэффициент загрузки (заполнения), оценки сложности. Выбор функции расстановки.
8	Заключение	Обзор проблематики структур данных и алгоритмов. Связь с учебной дисциплиной следующего семестра «Построение и анализ алгоритмов».

## 4.2 Перечень лабораторных работ

Наименование лабораторной работы	Количество ауд. часов
1. Программирование рекурсивных алгоритмов	6
2. Программирование алгоритмов сортировки	8
3. Программирование алгоритмов с бинарными деревьями	10
4. Программирование алгоритмов хеширования	10
Итого	34

## 4.3 Перечень практических занятий

Практические занятия не предусмотрены.

## 4.4 Курсовое проектирование

Цель работы (проекта): Приобретение навыков программной реализации на языке C++ и экспериментальное машинное исследование алгоритмов кодирования, быстрого поиска.

Содержание работы (проекта): Студенты получают задание на курсовую работу в начале семестра.

Курсовая работа состоит из следующих этапов:

1. Анализ задачи, определение цели, технологии проведения и составление плана экспериментального исследования;
2. Генерация представительного множества реализаций входных данных (с заданными особенностями распределения);
3. Выполнение исследуемых алгоритмов на сгенерированных наборах данных. При этом в ходе вычислительного процесса фиксируются как характеристики (например, время) работы программы, так и количество произведенных базовых операций алгоритма;
4. Фиксация результатов испытаний алгоритма, накопление статистики;
5. Представление результатов испытаний, их интерпретация и сопоставление с теоретическими оценками.

Результаты выполнения курсовой работы представляются в виде отчета.

Отчёт по курсовой работе оформляется в соответствии с шаблоном оформления курсовой работы, размещённым на сайте вуза. Отчет должен содержать 10-35 страниц формата А4. Помимо листов, предусмотренных шаблоном (титульный, лист задания, аннотация, содержание, введение, выводы), в отчёте должны быть:

- формальная постановка задачи (допустимо размещать в составе введения);

- описание алгоритма;
- описание структур данных и функций;
- описание интерфейса пользователя;
- тестирование и примеры работы программы;
- исследование алгоритма;
- программный код (с комментариями) в приложении.

Все этапы выполнения курсовой работы должны найти отражение в отчете.

Преподаватель проверяет курсовую работу, беседует со студентом, оценивает каждый этап курсовой работы оценивается максимум в 4 балла, баллы снижаются в случае неполного описания содержания этапа курсовой работы.

Критерии оценивания курсовой работы:

Студентам, набравшим 19-20 баллов, ставится оценка Отлично,

16-18 баллов - Хорошо,

11-15 баллов - Удовлетворительно,

ниже 11 баллов - Неудовлетворительно.

Темы:

№ п/п	Название темы	Перевод темы
1	Статическое кодирование и декодирование текстового файла методом Хаффмана	Text file static encode and decode using the Huffman method
2	Статическое кодирование и декодирование текстового файла методом Фано-Шеннона	Text file static encode and decode using the Fano-Shannon method
3	Динамическое кодирование и декодирование по Хаффману – сравнительное исследование со “статическим” методом	Dynamic encoding and decoding according to Huffman -a comparative study with the "static" method
4	Рандомизированные пирамиды поиска (Treaps) □ вставка и исключение	Randomized search pyramids (Treaps) □ insertion and exclusion
5	Случайные бинарные деревья поиска □ вставка и исключение	Random binary search trees □ insertion and exclusion

#### **4.5 Реферат**

Реферат не предусмотрен.

#### **4.6 Индивидуальное домашнее задание**

Индивидуальное домашнее задание не предусмотрено.

#### **4.7 Доклад**

Доклад не предусмотрен.

#### **4.8 Кейс**

Кейс не предусмотрен.

#### **4.9 Организация и учебно-методическое обеспечение самостоятельной работы**

Изучение дисциплины сопровождается самостоятельной работой студентов с рекомендованными преподавателем литературными источниками и информационными ресурсами сети Интернет.

Планирование времени для изучения дисциплины осуществляется на весь период обучения, предусматривая при этом регулярное повторение пройденного материала. Обучающимся, в рамках внеаудиторной самостоятельной работы, необходимо регулярно дополнять сведениями из литературных источников материал, законспектированный на лекциях. При этом на основе изучения рекомендованной литературы целесообразно составить конспект основных положений, терминов и определений, необходимых для освоения разделов учебной дисциплины.

Особое место уделяется консультированию, как одной из форм обучения и контроля самостоятельной работы. Консультирование предполагает особым



образом организованное взаимодействие между преподавателем и студентами, при этом предполагается, что консультант либо знает готовое решение, которое он может предписать консультируемому, либо он владеет способами деятельности, которые указывают путь решения проблемы.

<b>Текущая СРС</b>	<b>Примерная трудоемкость, ач</b>
Работа с лекционным материалом, с учебной литературой	8
Опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях)	
Самостоятельное изучение разделов дисциплины	4
Выполнение домашних заданий, домашних контрольных работ	
Подготовка к лабораторным работам, к практическим и семинарским занятиям	10
Подготовка к контрольным работам, коллоквиумам	8
Выполнение расчетно-графических работ	
Выполнение курсового проекта или курсовой работы	8
Поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме	
Работа над междисциплинарным проектом	
Анализ данных по заданной теме, выполнение расчетов, составление схем и моделей, на основе собранных данных	
Подготовка к зачету, дифференцированному зачету, экзамену	35
<b>ИТОГО СРС</b>	<b>73</b>

## 5 Учебно-методическое обеспечение дисциплины

### 5.1 Перечень основной и дополнительной литературы, необходимой для освоения дисциплины

№ п/п	Название, библиографическое описание	К-во экз. в библ.
Основная литература		
1	Алексеев, Андрей Юрьевич. Динамические структуры данных [Текст] : учеб. пособие / А.Ю. Алексеев, С.А. Ивановский, Д.В. Куликов, 2004. -83 с.	151
2	Ивановский, Сергей Алексеевич. Деревья кодирования и поиска [Текст] : учеб. пособие / С.А. Ивановский, 2006. -83 с.	158
3	Алексеев, Андрей Юрьевич. Алгоритмы сортировки [Текст] : учеб. пособие / А.Ю. Алексеев, С.А.Ивановский, С.А. Фролова, 2009. -64 с.	157
4	Методические указания к лабораторным работам по дисциплине "Базы данных" [Текст] : учеб. пособие / Сост.: А.Б. Верховинский, С.В. Иванов, Г.В. Разумовский [и др.]; СПб.ГЭТУ им. В.И. Ульянова (Ленина), 1994. - 32 с. с.	9
Дополнительная литература		
1	Ахо, Альфред В. Структуры данных и алгоритмы [Текст] : монография / А.В.Ахо, Д.Э.Хопкрофт, Дж.Д.Ульман, 2001. -382 с.	43
2	Кнут Д.Э. Искусство программирования [Текст] : в 3 т. : [учеб. пособие] : пер. с англ. -(Классический труд). Т. 1 : Основные алгоритмы : учебное пособие, 2000. -712 с.	14
3	Кнут Д.Э. Искусство программирования: В 3 т.: Пер с англ [Текст] : Учеб. пособие. Т. 3 : Сортировка и поиск : учебное пособие, 2000. -822 с.	14
4	Кормен, Томас. Алгоритмы: построение и анализ [Текст] : Учеб. / Т. Кормен; Ч.Лейзерсон, Р.Ривест; Пер. с англ. под ред. А.Шен, 1999. -955 с.	8
5	Макконелл Дж. Анализ алгоритмов. Вводный курс [Текст] : учебное пособие / Дж. Макконелл; пер. с англ. С.К. Ландо, 2002. -302 с.	9
6	Кубенский, Александр Александрович. Структуры и алгоритмы обработки данных : объектно-ориентированный подход и реализация на С++ [Текст] : учеб. пособие для вузов по специальности "Математическое обеспечение и администрирование информ. систем" -351500 / А.А. Кубенский, 2004. -464 с.	27
7	Вирт, Никлаус. Алгоритмы + структуры данных = программы [Текст] : науч. изд. / Н. Вирт ; пер. с англ. Л.Ю. Иоффе ; под ред. Д.Б. Подшивалова, 1985. -406 с.	61
8	Седжвик, Роберт. Фундаментальные алгоритмы на С [Текст] : [пер. с англ.]. Ч. 1-5 : Анализ структуры данных. Сортировка. Поиск. Алгоритмы на графах / Р. Седжвик, 2003. -1127 с.	40
9	Бауэр, Фридрих Л. Информатика: ввод. курс [Текст] : в 2 ч. Ч. 2, 1990. - 742 с	83

<b>№ п/п</b>	<b>Название, библиографическое описание</b>	<b>К-во экз. в библ.</b>
10	Лисков, Барбара. Использование абстракций и спецификаций при разработке программ [Текст] / Б. Лисков, Дж. Гатэг ; пер. с англ. С. А. Жигалкина [и др.], 1989. -424 с.	33
11	Гудрич, Майкл Т. Структуры данных и алгоритмы в Java [Текст] : монография / М.Т. Гудрич, Р. Тамассия ; [пер.с англ. А.М. Чернухо], 2003. -670 с.	21

## **5.2 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», используемых при освоении дисциплины**

<b>№ п/п</b>	<b>Электронный адрес</b>
1	Бьерн Страуструп. Язык программирования C++ <a href="http://lib.ru/CPPIHV/cpptut.txt">http://lib.ru/CPPIHV/cpptut.txt</a>
2	В. Е. Алексеев, В. А. Таланов Структуры данных и модели вычислений (ИНТУИТ) <a href="http://www.intuit.ru/department/algorithms/dscm/">http://www.intuit.ru/department/algorithms/dscm/</a>
3	Инструменты, алгоритмы и структуры данных Автор: Б. Мейер (ИНТУИТ) <a href="http://www.intuit.ru/department/se/ialgdate/">http://www.intuit.ru/department/se/ialgdate/</a>

## **5.3 Адрес сайта курса**

Адрес сайта курса: <https://vec.etu.ru/moodle/course/view.php?id=10640>

## 6 Критерии оценивания и оценочные материалы

### 6.1 Критерии оценивания

Для дисциплины «Алгоритмы и структуры данных» предусмотрены следующие формы промежуточной аттестации: экзамен.

#### Экзамен

Оценка	Описание
Неудовлетворительно	Курс не освоен. Студент испытывает серьезные трудности при ответе на ключевые вопросы дисциплины
Удовлетворительно	Студент в целом овладел курсом, но некоторые разделы освоены на уровне определений и формулировок
Хорошо	Студент овладел курсом, но в отдельных вопросах испытывает затруднения. Умеет решать задачи
Отлично	Студент демонстрирует полное овладение курсом, способен применять полученные знания при решении конкретных задач

## Особенности допуска

К экзамену допускаются студенты, выполнившие и защитившие 4 лабораторные работы, написавшие 2 контрольные работы на оценки не ниже "удовлетворительно", защитившие курсовую работу на оценку не ниже "удовлетворительно". Экзамен проводится в форме ответов на вопросы билетов.

## 6.2 Оценочные материалы для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине

### Вопросы к экзамену

№ п/п	Описание
1	Рекурсивные алгоритмы.
2	Выполнение рекурсивных алгоритмов: подготовка трассы, стек (магазин).
3	Анализ рекурсивных алгоритмов.
4	Представление списков и реализация базовых функций на языке C++.
5	Элементы функционального стиля при программировании рекурсивной обработки списков на языке C++
6	Примеры операций над разреженными матрицами.
7	Стек, очередь и дек как последовательности с ограниченными наборами операций (доступа)
8	Метод кодирования Фано-Шеннона.
9	Критерий оптимальности кода.
10	Оценка сложности в худшем случае: деревья Фибоначчи.
11	Реализация упорядоченных линейных списков на базе БДП (АВЛ-деревьев или рандомизированных пирамид поиска).
12	Операции поиска, вставки и удаления элементов; операции сцепления и расщепления списков
13	Метод поиска с использованием функции расстановки (хеширование)
14	Выбор функции расстановки.
15	Бинарные деревья поиска (БДП)
16	Подсчет числа структурно различных бинарных деревьев с заданным числом узлов
17	Рандомизированные бинарные деревья поиска (пирамиды поиска – Treaps, Дерамиды)
18	Разрешение коллизий: метод внутренних и внешних цепочек, метод открытой адресации
19	Среднее время поиска в случайных деревьях
20	Исключение из АВЛ-дерева.
21	Реализация алгоритмов Хаффмана и Фано-Шеннона на C++
22	Соответствие комбинаций и иерархических списков.

## Форма билета

Министерство науки и высшего образования Российской Федерации  
ФГАОУ ВО «Санкт-Петербургский государственный электротехнический  
университет «ЛЭТИ» имени В.И. Ульянова (Ленина)»

---

### ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

Дисциплина **Алгоритмы и структуры данных ФКТИ**

1. Выполнение рекурсивных алгоритмов: подготовка трассы, стек (магазин)..
2. Реализация упорядоченных линейных списков на базе БДП (АВЛ-деревьев или рандомизированных пирамид поиска).
3. Задача.

УТВЕРЖДАЮ

Заведующий кафедрой

### Образцы задач (заданий) для контрольных (проверочных) работ

Примеры задач для формирования вариантов контрольных работ и билетов:

1. Записать скобочное представление заданного графически (рисунком) *леса*.

Рисунок выдается.

2. По заданному скобочному представлению нарисовать *дерево (лес)*

(a (d) (e)) (b (f (m) (n)) (g) (h)) (c (i) (j (o)) (k) (l (p) (q)))

3. (1) Найти результат выполнения заданной функции `Аноупт` с заданными аргументами. (2) Указать смысл действия этой функции в общем

случае?

Функция *anonym*:

```
unsigned int anonym (binTree t)
{
    if (isNull(t)) return 0;
    else return anonym (Left(t))+ anonym (Right(t))+ 1;
} // end anonym

(a)  $t = (a (b) (e))$ ;
(b)  $t = (a (b (c) (d)) (e \wedge (f)))$ .
```

**4. Описать рекурсивную функцию подсчета заданной числовой характеристики бинарного дерева.**

Числовая характеристика: *высота дерева*, т.е. максимальный уровень его элементов.

Весь комплект контрольно-измерительных материалов для проверки сформированности компетенции (индикатора компетенции) размещен в закрытой части по адресу, указанному в п. 5.3

### 6.3 График текущего контроля успеваемости

Неделя	Темы занятий	Вид контроля
1	Тема 1. Программирование рекурсивных алгоритмов	
2	Тема 2. Рекурсивная обработка иерархических списков	Отчет по лаб. работе
3	Тема 5. Сортировка	
4		
5		Отчет по лаб. работе
6	Тема 3. Линейные структуры данных	
7		Контрольная работа
8	Тема 4. Деревья и леса	
9		
10		
11		Отчет по лаб. работе
12	Тема 6. Быстрый поиск	
13		
14		Контрольная работа
15	Тема 6. Быстрый поиск	
16		Отчет по лаб. работе
17	Заключение	Защита КР / КП

### 6.4 Методика текущего контроля

#### на лекционных занятиях

Текущий контроль включает в себя:

– контроль посещаемости (не менее 80% занятий).

#### на контрольных работах

В течение семестра студенты пишут 2 контрольные работы. Каждая контрольная работа включает в себя одну задачу.

Контрольная работа оценивается следующим образом:

- оценка "неудовлетворительно" (0 баллов) - задача не решена;
- оценка "удовлетворительно" (1-3 балла) – частично решенная задача;
- оценка "хорошо" (4 балла) – полностью решенная задача с более или менее значительными недочетами;
- оценка "отлично" (5 баллов) – полностью решенная задача.



## **на лабораторных занятиях**

В процессе обучения по дисциплине студент обязан выполнить и успешно защитить 4 лабораторные работы. Под выполнением лабораторных работ подразумевается подготовка к работе, выполнение задания, подготовка отчета и его защита. Отчет представляется преподавателю на проверку. После проверки отчет либо возвращается (при наличии замечаний) на доработку, либо допускается к защите.

Лабораторные работы защищаются студентами индивидуально, в часы, отведенные для лабораторных работ. На защите лабораторной работы студент должен показать: понимание постановки задачи, подхода к ее решению, умение объяснять ход решения, выбор тех или иных методик решения задачи. Преподаватель задает вопросы, позволяющие определить глубину понимания теоретического материала, который лежит в основе решения задачи лабораторной работы, а также самостоятельность ее выполнения.

Текущий контроль включает в себя контроль выполнения лабораторной работы и сдачи в срок отчета по лабораторной работе.

Критерии оценивания: «не зачтено» - ставится, если основное содержание материала работы не раскрыто, не даны ответы на вопросы преподавателя, допущены грубые ошибки в определении понятий и в использовании терминологии; «зачтено» ставится, если продемонстрировано усвоение основного содержания материала, работа выполнена полностью, самостоятельно и оформлена в соответствии с требованиями.

### **самостоятельной работы студентов**

Контроль самостоятельной работы студентов осуществляется на лекционных, лабораторных занятиях по методикам, описанным выше

### **при выполнении курсового проекта (работы)**

Текущий контроль при выполнении курсового проекта (работы) осуществ-

ляется в соответствии с методическими указаниями по курсовом проектированию и заданием на курсовой проект (работу).

Оформление пояснительной записки на курсовой проект (работу) выполняется в соответствии с требованиями к студенческим работам принятым в СПбГЭТУ.

Критерии оценивания курсовой работы:

«отлично» - вопрос раскрыт полностью, задача решена правильно - 19-20 баллов

«хорошо» - вопрос раскрыт не полностью, задача решена частично - 16-18 баллов

«удовлетворительно» - в ответе на вопрос имеются существенные ошибки; задача не решена или решена неправильно, ход решения правильный - 11-15 баллов

«неудовлетворительно» - отсутствует ответ на вопрос или содержание ответа не совпадает с поставленным вопросом, задача не решена, ход решения неправильный - ниже 11 баллов

## 7 Описание информационных технологий и материально-технической базы

Тип занятий	Тип помещения	Требования к помещению	Требования к программному обеспечению
Лекция	Лекционная аудитория	Количество посадочных мест – в соответствии с контингентом, рабочее место преподавателя, IBM совместимый компьютер Pentium или выше, проектор, экран, меловая или маркерная доска	1) Linux Альт Образование 10 и выше; 2) P7-Офис 7 и выше либо LibreOffice 7 и выше
Лабораторные работы	Лаборатория	Количество посадочных мест, оборудованных персональными IBM совместимыми компьютерами Pentium или выше в соответствии с контингентом, рабочее место преподавателя, IBM совместимый компьютер Pentium или выше, проектор, экран, меловая или маркерная доска	1) Linux Альт Образование 10 и выше; 2) P7-Офис 7 и выше либо LibreOffice 7 и выше
Самостоятельная работа	Помещение для самостоятельной работы	Оснащено компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационнообразовательную среду университета	1) Linux Альт Образование 10 и выше; 2) P7-Офис 7 и выше либо LibreOffice 7 и выше

## **8 Адаптация рабочей программы для лиц с ОВЗ**

Адаптированная программа разрабатывается при наличии заявления со стороны обучающегося (родителей, законных представителей) и медицинских показаний (рекомендациями психолого-медико-педагогической комиссии). Для инвалидов адаптированная образовательная программа разрабатывается в соответствии с индивидуальной программой реабилитации.

## ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

<b>№ п/п</b>	<b>Дата</b>	<b>Изменение</b>	<b>Дата и номер протокола заседания УМК</b>	<b>Автор</b>	<b>Начальник ОМОЛА</b>