

УТВЕРЖДАЮ  
 Декан факультета

\_\_\_\_\_ Матвеев П.В.  
 (подпись) ФИО  
 «\_\_\_» \_\_\_\_\_ 20\_\_

## РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ СТРУКТУРЫ И ОРГАНИЗАЦИЯ ДАННЫХ

Направление/специальность подготовки	09.03.04 Программная инженерия
Специализация/профиль/программа подготовки	Разработка программно-информационных систем
Уровень высшего образования	Бакалавриат
Форма обучения	Очная
Факультет	О Естественнонаучный
Выпускающая кафедра	О7 Информационные системы и программная инженерия
Кафедра-разработчик рабочей программы	О7 Информационные системы и программная инженерия

КУРС	СЕМЕСТР	ОБЩАЯ ТРУДОЁМКОСТЬ (ЗАЧЕТНЫХ ЕДИНИЦ)	ЧАСЫ (по наличию видов занятий)									ВИД ПРОМЕЖУТОЧНОГО КОНТРОЛЯ
			ОБЩАЯ ТРУДОЁМКОСТЬ	АУДИТОРНЫЕ ЗАНЯТИЯ				САМОСТОЯТЕЛЬНАЯ РАБОТА				
				ВСЕГО	ЛЕКЦИИ	ЛАБОРАТОРНЫЙ ПРАКТИКУМ	ПРАКТИЧЕСКИЕ ЗАНЯТИЯ	ВСЕГО	КУРСОВОЙ ПРОЕКТ	КУРСОВАЯ РАБОТА	ДРУГИЕ ВИДЫ САМОСТ. РАБОТЫ	
1	2	4	144	51	34	0	17	93	0	0	93	ЭКЗ.

*ЛИСТ СОГЛАСОВАНИЯ*

РАБОЧАЯ ПРОГРАММА СОСТАВЛЕНА В СООТВЕТСТВИИ С ТРЕБОВАНИЯМИ ФЕДЕРАЛЬНОГО  
ГОСУДАРСТВЕННОГО ОБРАЗОВАТЕЛЬНОГО СТАНДАРТА ВЫСШЕГО ОБРАЗОВАНИЯ (ФГОС ВО)

**09.03.04 Программная инженерия**

год набора группы: 2024

Программу составил:

Кафедра О7 Информационные системы и программная инженерия  
Палехова Ольга Александровна, старший преподаватель

\_\_\_\_\_

Программа рассмотрена  
на заседании кафедры-разработчика  
рабочей программы **О7 Информационные системы и программная инженерия**

Заведующий кафедрой Семенова Е.Г., д.т.н., проф.

\_\_\_\_\_

Программа рассмотрена  
на заседании выпускающей кафедры

**О7 Информационные системы и программная инженерия**

Заведующий кафедрой Семенова Е.Г., д.т.н., проф.

\_\_\_\_\_

## **РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ СТРУКТУРЫ И ОРГАНИЗАЦИЯ ДАННЫХ**

### **Разделы рабочей программы**

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО
3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ
4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ
5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ
6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### **Приложения к рабочей программе дисциплины**

- Приложение 1. Аннотация рабочей программы
- Приложение 2. Технологии и формы обучения
- Приложение 3. Фонды оценочных средств

# 1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является формирование следующих компетенций:

ОПК-6 — способность разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов

ПК-94 — способен к управлению информацией и данными, поиску источников информации и данных, восприятию, анализу, запоминанию и передаче информации с использованием цифровых средств, а также с помощью алгоритмов при работе с полученными из различных источников данными с целью эффективного использования полученной информации для решения задач

Формированию компетенций служит достижение следующих результатов образования:

## **ОПК-6**

*знания:*

современного спектра задач, принципов построения программных систем;  
системного подхода к построению программных систем;  
основных структур данных и базовых алгоритмов их обработки;  
принципов выбора структур данных и алгоритмов и способов их реализации;  
теории структур данных и алгоритмов как основы построения программных систем;  
принципов оценки объемной и временной сложности алгоритмов;

*умения:*

применять общие принципы организации программных систем;  
выбирать оптимальные структуры данных и алгоритмы при проектировании программ;  
работать с основными структурами данных: стеками, очередями, списками, деревьями и графами;

владеть методами сравнительной оценки сложности алгоритмов;

*навыки:*

программирования различных структур данных;  
написания программ с использованием различных структур данных и стандартных алгоритмов их обработки;  
применения методов анализа объемно-временной сложности разрабатываемых программных компонент.

## **ПК-94**

*знания:*

основных структур данных и базовых алгоритмов их обработки;  
принципов оценки объемной и временной сложности алгоритмов;  
принципов выбора структур данных и алгоритмов и способов их реализации;  
теории структур данных и алгоритмов как основы построения программных систем;

*умения:*

выбирать оптимальные структуры данных и алгоритмы при проектировании программ;  
работать с основными структурами данных: стеками, очередями, списками, деревьями и графами;

владеть методами сравнительной оценки сложности алгоритмов;

*навыки:*

программирования различных структур данных;  
написания программ с использованием различных структур данных и стандартных алгоритмов их обработки;  
применения методов анализа объемно-временной сложности разрабатываемых программных компонент.

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО

Дисциплина **СТРУКТУРЫ И ОРГАНИЗАЦИЯ ДАННЫХ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению *09.03.04 Программная инженерия*.

Содержание дисциплины является логическим продолжением дисциплин: **ВВЕДЕНИЕ В ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ, ИНФОРМАТИКА: ОСНОВЫ ПРОГРАММИРОВАНИЯ.**

Содержание дисциплины является основой для освоения дисциплин: **ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ПРОГРАММИРОВАНИЕ, БАЗЫ ДАННЫХ, СИСТЕМНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ.**

Предварительные компетенции, сформированные у обучающегося до начала изучения дисциплины:

- ОПК-2 — Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности
- ОПК-3 — Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности
- ОПК-6 — Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов
- ПК-94 — способен к управлению информацией и данными, поиску источников информации и данных, восприятию, анализу, запоминанию и передаче информации с использованием цифровых средств, а также с помощью алгоритмов при работе с полученными из различных источников данными с целью эффективного использования полученной информации для решения задач

### 3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 4 з.е., 144 ч.

#### 3.1. Содержание (дидактика) дисциплины

КУРС	СЕМЕСТР	Наименование разделов и дидактических единиц	ВСЕГО	Аудиторные занятия в контактной форме			Самостоятельная работа студентов	Формируемая компетенция, %	
				ВСЕГО	Лекции	Практические занятия		ОПК-6	ПК-94
1	2	<b>Раздел 1. Структуры данных и структуры хранения. Линейные структуры данных.</b> 1.1. Построение баз данных и технология их программирования с помощью сложных структур данных. 1.2. Понятия структуры данных (СД) и структуры хранения (СХ). 1.3. Структура данных, абстрактный тип данных (АТД), тип данных в языке программирования. Классификация СД. 1.4. Структурные типы данных в языке Си: массивы, структуры, объединения, записи с вариантами. Классы и шаблоны классов. 1.5. Классификация структур хранения. Векторные, связные и гибридные СХ. Достоинства и недостатки разных СХ, основные принципы выбора СХ для организации выбранной СД при решении задач. 1.6. Организация связных структур хранения: односвязного и двусвязного линейных списков.	21	6	4	2	15	20	20
1	2	<b>Раздел 2. Линейные структуры данных.</b> 2.1. СД Стек. Организация стека с помощью массива и односвязного линейного списка. Примеры использования стеков. Вычисление выражения, записанного в постфиксной форме. АТД Стек. Класс Стек. Шаблон класса Стек. 2.2. СД Очередь. АТД Очередь. Организация очереди с помощью массива и односвязного линейного списка. Примеры задач, решаемых с помощью очередей. 2.3. СД Дек, дек с ограниченным входом, дек с ограниченным выходом. Организация деков с помощью векторной и связных СХ. Примеры задач, решаемых с помощью деков. 2.4. СД Очередь с приоритетом и способы ее организации. 2.5. СД Список. Виды списков. Способы организации списков. Примеры задач.	24	6	4	2	18	20	20
1	2	<b>Раздел 3. Нелинейные структуры данных.</b> 3.1. АТД множество, мультимножество, ассоциативный массив (словарь, map), multimap. 3.2. Организация множества с помощью битовой карты. 3.3. Хеш-таблицы. Выбор хеш-функции. Разрешение коллизий при хешировании методом открытой адресации и методом цепочек. 3.4. Древовидные структуры данных и структуры хранения. Основные термины. 3.5. СД Бинарное дерево. Примеры задач, решаемых с помощью бинарных деревьев. Способы организации бинарного дерева. Рекурсивные и не рекурсивные алгоритмы обхода дерева. 3.6. СД Пирамида (куча). Организация пирамиды с помощью векторной СХ, алгоритм просеивания ключа через пирамиду. 3.7. СД Бинарное дерево поиска. Организация бинарного дерева поиска с помощью связной СХ, алгоритмы вставки и удаления элементов из дерева бинарного поиска. Сбалансированное дерево бинарного поиска, методы балансировки. 3.8. Общие деревья, лес. Способы организации деревьев с помощью разных СХ. 3.9. СД Граф. Виды графов. Примеры задач на графах. Способы задания графов: матрица смежности, списки смежности, матрица инцидентности. АТД Граф. Способы организации графов с помощью различных СХ. Основные алгоритмы обработки графов: добавление и удаление узлов и ребер, поиск в глубину и в ширину, поиск циклов, транзитивное замыкание матрицы смежности, определение компонент связности, построение остоного дерева (леса). 3.10. СД Взвешенный (помеченный) граф. Способы организации взвешенных графов с помощью различных СХ. Полезные алгоритмы обработки взвешенных графов: поиск кратчайших путей, вычисление максимального потока, определение центра и медианы орграфа, построение минимального остоного леса графа.	49	19	14	5	30	30	30
1	2	<b>Раздел 4. Алгоритмы сортировки и поиска.</b> 4.1. Понятие сортировки. Устойчивость сортировки. Внешние и внутренние сортировки. Прямые (прямого обмена, вставки, выбора) и улучшенные методы сортировки. Шейкерная сортировка, сортировка Хоара, сортировка Шелла, пирамидальная сортировка. Сортировка слиянием. Простое, естественное, многопутевое и многофазное слияние. Подготовка последовательностей перед слиянием. Сортировка Timsort. Сравнительная оценка трудоемкости различных методов сортировки. 4.2. Основные понятия поиска. Внутренний и внешний ключи поиска. Основные методы поиска. Последовательный поиск. Поиск в упорядоченной таблице: индексно-последовательный поиск, бинарный поиск. Поиск по дереву: АВЛ-дерево, красно-черное дерево, декартово дерево, рандомизированное дерево поиска, splay-дерево, scapegoat-дерево, Б-дерева, бор, префиксное дерево, radix-tree. Поиск с использованием хеш-таблиц.	50	20	12	8	30	30	30
Всего за 2 семестр			144	51	34	17	93	100	100
Всего по дисциплине			144	51	34	17	93	100	100

#### 3.2. Аудиторный практикум

№ п/п	Номер и наименование раздела дисциплины	Тема практического занятия	Объем, ауд. часов
1	Раздел 1. Структуры данных и структуры хранения. Линейные структуры данных.	Построение связных линейных списков	2
2	Раздел 2. Линейные структуры данных.	Способы организации стеков, очередей и деков. Решение задач с использованием линейных СД.	2

3	Раздел 3. Нелинейные структуры данных.	Построение и обходы бинарного дерева	2
4		Способы задания графов и выбор структур хранения для их различных представлений	1
5		Алгоритмы на графах	2
6	Раздел 4. Алгоритмы сортировки и поиска.	Оценка временной и пространственной сложности алгоритмов на примере алгоритмов сортировки.	4
7		Оценка временной и пространственной сложности поиска при использовании различных способов организации данных	4
Всего за 2 семестр			17

### 3.3. Самостоятельная работа студента (СРС)

№ п/п	Номер и наименование раздела дисциплины	Содержание учебного задания	Объем, часов
1	Раздел 1. Структуры данных и структуры хранения. Линейные структуры данных.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	4
2		Подготовка к практическому занятию: анализ задачи, построение алгоритмов	2
3		Написание программного кода	8
4		Оформление отчета по ПР	1
5	Раздел 2. Линейные структуры данных.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	5
6		Написание программного кода	10
7		Оформление отчета по ПР	1
8		Подготовка к практическому занятию: анализ задачи, выбор структуры хранения, построение алгоритмов	2
9	Раздел 3. Нелинейные структуры данных.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	6
10		Подготовка к практическим занятиям: анализ задачи, выбор структуры хранения, построение алгоритмов	2
11		Написание программного кода	20
12		Оформление отчета по ПР	2
13	Раздел 4. Алгоритмы сортировки и поиска.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	16
14		Подготовка к практическим занятиям: оценка трудоемкости заданных алгоритмов по заданному критерию	6
15		Написание программного кода	6
16		Оформление отчета по ПР	2
Всего за 2 семестр			93

## 4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

СЕМЕСТР	НЕДЕЛИ СЕМЕСТРА																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
2				Отч. по ПЗ		ДР	Отч. по ПЗ, ОС			ДР	Отч. по ПЗ, ОС					ДР	Отч. по ПЗ, Вопр. Экз, ОС

Условные обозначения:

- ДР – диагностическая работа;
- Отч. по ПЗ – отчет по практическому заданию;

- Вопр. Экз – вопросы к экзамену;
- ОС – устный опрос студентов.

**Текущий контроль успеваемости** студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- отчет по практическому заданию;
- вопросы к экзамену;
- устный опрос студентов.

**Промежуточная аттестация** проводится в формах:

- экзамен.



## 5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 5.1. Основная литература по дисциплине:

1. Л. А. Павлов, Н. В. Первова. . Структуры и алгоритмы обработки данных. Санкт-Петербург: Лань, 2021, эл. рес.
2. Л. А. Павлов, Н. В. Первова. . Структуры и алгоритмы обработки данных. СПб.: Лань, 2021, 50 экз.
3. Н. Вирт. . Алгоритмы и структуры данных. М.: Мир, 1989, эл. рес.
4. Р. Л. Круз. . Структуры данных и проектирование программ. М.: БИНОМ. Лаборатория знаний, 2008, 5 экз.
5. С. А. Апанасевич. . Структуры и алгоритмы обработки данных. Линейные структуры. Санкт-Петербург: Лань, 2022, эл. рес.
6. Т. Н. Варфоломеева. . Структуры данных и основные алгоритмы их обработки. Москва: Флинта, 2017, эл. рес.

### 5.2. Дополнительная литература по дисциплине:

1. А. В. Ахо, Д. Холткрофт, Д. Д. Ульман. . Структуры данных и алгоритмы. М.: Вильямс, 2000, 2 экз.
2. Д. Э. Кнут. . Искусство программирования. М.: Вильямс, 2001, 0 экз.
3. Т. Кормен, Ч. Лейзерсон, Р. Ривест. . Алгоритмы: построение и анализ. М.: МЦНМО, 2000, 0 экз.

### 5.3. Периодические издания:

не требуются.

### 5.4. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины, электронные библиотечные системы:

1. <https://e.lanbook.com/> - ЭБС Лань;
2. <https://urait.ru/> - ЭБС Юрайт;
3. <http://www.intuit.ru/studies/courses/12181/1174/info> - курс "Алгоритмы на C++";
4. <http://www.lektorium.tv/lecture/13343> – видеокурс лекций по алгоритмам и структурам данных;
5. <http://algotlist.manual.ru/> - подборка полезных материалов по структурам данных и алгоритмам;
6. [http://library.voenmeh.ru/jirbis2/index.php?option=com\\_irbis&view=irbis&Itemid=474](http://library.voenmeh.ru/jirbis2/index.php?option=com_irbis&view=irbis&Itemid=474) - электронная библиотека университета — Фундаментальная библиотека БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова;
7. <https://www.cs.usfca.edu/~galles/visualization/Algorithms.html> - визуализаторы структур данных и алгоритмов.

### Современные профессиональные базы данных:

1. <https://rusneb.ru> – Национальная электронная библиотека (НЭБ);
2. <https://cyberleninka.ru/> - Научная электронная библиотека «Киберленинка»;  
<http://www.rfbg.ru/rffi/ru/library> - Полнотекстовая электронная библиотека Российского фонда фундаментальных исследований.

### Информационные справочные системы:

1. Техэксперт – Информационный портал технического регулирования: Нормы, правила, стандарты РФ;
2. [http://library.voenmeh.ru/jirbis2/index.php?option=com\\_irbis&view=irbis&Itemid=457](http://library.voenmeh.ru/jirbis2/index.php?option=com_irbis&view=irbis&Itemid=457) - БД ГОСТов собственной генерации БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова;
3. <http://www.consultant.ru/>- КонсультантПлюс- информационный портал правовой информации.

### 5.5. Программное обеспечение:

1. Офисный пакет Libre Office;
2. Интегрированная среда разработки Code::Blocks;
3. Интернет-браузер Mozilla Firefox.

#### 5.6. Информационные технологии:

взаимодействие с обучающимися посредством ЭИОС Moodle БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова.

## **6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

### **6.1. Лекционные занятия:**

специализированные требования по оборудованию отсутствуют; аудитория с посадочными местами по количеству студентов; доска.

### **6.2. Практические занятия:**

1. Офисный пакет Libre Office;
2. Интегрированная среда разработки Code::Blocks;
3. Интернет-браузер Mozilla Firefox.

### **6.3. Прочее:**

1. рабочее место преподавателя, оснащенное компьютером с доступом в Интернет;
2. рабочие места студентов, оснащенные компьютерами с доступом в Интернет, предназначенные для работы в электронной образовательной среде.

### Аннотация рабочей программы

Дисциплина **СТРУКТУРЫ И ОРГАНИЗАЦИЯ ДАННЫХ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению *09.03.04 Программная инженерия*. Дисциплина реализуется на факультете *О Естественнотехнический БГТУ "ВОЕНМЕХ"* им. Д.Ф. Устинова кафедрой *О7 Информационные системы и программная инженерия*.

Дисциплина нацелена на формирование *компетенций*:

ОПК-6 способность разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов;

ПК-94 способен к управлению информацией и данными, поиску источников информации и данных, восприятию, анализу, запоминанию и передаче информации с использованием цифровых средств, а также с помощью алгоритмов при работе с полученными из различных источников данными с целью эффективного использования полученной информации для решения задач.

Содержание дисциплины охватывает круг вопросов, связанных с поиском, сбором, хранением, преобразованием и использованием данных. Приводятся разнообразные реализации абстрактных типов данных. Рассматриваются методы анализа и построения алгоритмов для работы с графами, внутренней и внешней сортировкой, управлением памятью, которые играют важную роль во множестве приложений, таких как сетевая связность, конструирование электронных схем, составление графиков, обработка транзакций и выделение ресурсов. Акцентируется внимание на объектно-ориентированном и обобщенном программировании.

Программой дисциплины предусмотрены следующие **виды контроля**:

**Текущий контроль успеваемости** студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- отчет по практическому заданию;
- вопросы к экзамену;
- устный опрос студентов.

**Промежуточная аттестация** проводится в формах:

- экзамен.

Общая трудоемкость освоения дисциплины составляет **4 з.е., 144 ч.** Программой дисциплины предусмотрены лекционные занятия (**34 ч.**), практические занятия (**17 ч.**), самостоятельная работа студента (**93 ч.**).

## ТЕХНОЛОГИИ И ФОРМЫ ОБУЧЕНИЯ

### Рекомендации по освоению дисциплины для студента

Трудоемкость освоения дисциплины составляет 144 ч., из них 51 ч. аудиторных занятий, и 93 ч., отведенных на самостоятельную работу студента.

Рекомендации по распределению учебного времени по видам самостоятельной работы и разделам дисциплины приведены в таблице.

Контроль освоения дисциплины производится в соответствии с Положением о текущем, рубежном контроле успеваемости и промежуточной аттестации обучающихся.

Формы контроля и критерии оценивания приведены в приложении 3 к Рабочей программе.

Наименование работы	Рекомендуемая литература	Трудоемкость, час.
<b>Раздел 1. Структуры данных и структуры хранения. Линейные структуры данных.</b>		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Н. Вирт. . Алгоритмы и структуры данных: М.: Мир, 1989 (1.1-1.7)	4
Подготовка к практическому занятию: анализ задачи, построение алгоритмов	Т. Н. Варфоломеева. . Структуры данных и основные алгоритмы их обработки: Москва: Флинта, 2017 (1.1)	2
Написание программного кода	А. В. Ахо, Д. Холткрофт, Д. Д. Ульман. . Структуры данных и алгоритмы: М.: Вильямс, 2000 (4.1-4.4)	8
Оформление отчета по ПР		1
Итого по разделу 1		15
<b>Раздел 2. Линейные структуры данных.</b>		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Н. Вирт. . Алгоритмы и структуры данных: М.: Мир, 1989 (4.1-4.3) Д. Э. Кнут. . Искусство программирования: М.: Вильямс, 2001 (т.1 гл. 2.1, 2.2) Т. Кормен, Ч. Лейзерсон, Р. Ривест. . Алгоритмы: построение и анализ: М.: МЦНМО, 2000 (11.1-11.3)	5
Написание программного кода	А. В. Ахо, Д. Холткрофт, Д. Д. Ульман. . Структуры данных и алгоритмы: М.: Вильямс, 2000 (2)	10
Оформление отчета по ПР	Л. А. Павлов, Н. В. Первова. . Структуры и алгоритмы обработки данных: Санкт-Петербург: Лань, 2021 (2.1-2.4) Р. Л. Круз. . Структуры данных и проектирование программ: М.: БИНОМ. Лаборатория знаний, 2008 (3-6)	1
Подготовка к практическому занятию: анализ задачи, выбор структуры хранения, построение алгоритмов	С. А. Апанасевич. . Структуры и алгоритмы обработки данных. Линейные структуры: Санкт-Петербург: Лань, 2022 (3) Т. Н. Варфоломеева. . Структуры данных и основные алгоритмы их обработки: Москва: Флинта, 2017 (2.2, 3.1-3.4)	2
Итого по разделу 2		18
<b>Раздел 3. Нелинейные структуры данных.</b>		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Т. Кормен, Ч. Лейзерсон, Р. Ривест. . Алгоритмы: построение и анализ: М.: МЦНМО, 2000 (5.4, 5.5, 23-26)	6
Подготовка к практическим занятиям: анализ задачи, выбор структуры хранения, построение алгоритмов	Р. Л. Круз. . Структуры данных и проектирование программ: М.: БИНОМ. Лаборатория знаний, 2008 (10, 12)	2

Написание программного кода	Н. Вирт. . Алгоритмы и структуры данных: М.: Мир, 1989 (4.4)	20
Оформление отчета по ПР	А. В. Ахо, Д. Холткрофт, Д. Д. Ульман. . Структуры данных и алгоритмы: М.: Вильямс, 2000 (3, 6, 7) Д. Э. Кнут. . Искусство программирования: М.: Вильямс, 2001 (т.1 гл. 2.3, 2.4) Л. А. Павлов, Н. В. Первова. . Структуры и алгоритмы обработки данных: СПб.: Лань, 2021 (2.5, 3, 6) Т. Н. Варфоломеева. . Структуры данных и основные алгоритмы их обработки: Москва: Флинта, 2017 (3.5, 3.6) Л. А. Павлов, Н. В. Первова. . Структуры и алгоритмы обработки данных: Санкт-Петербург: Лань, 2021 (2.5, 3, 6)	2
Итого по разделу 3		30
<b>Раздел 4. Алгоритмы сортировки и поиска.</b>		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	А. В. Ахо, Д. Холткрофт, Д. Д. Ульман. . Структуры данных и алгоритмы: М.: Вильямс, 2000 (8-10) Т. Кормен, Ч. Лейзерсон, Р. Ривест. . Алгоритмы: построение и анализ: М.: МЦНМО, 2000 (2-4, 7-9, 12-14, 19) Д. Э. Кнут. . Искусство программирования: М.: Вильямс, 2001 (т.3)	16
Подготовка к практическим занятиям: оценка трудоемкости заданных алгоритмов по заданному критерию	С. А. Апанасевич. . Структуры и алгоритмы обработки данных. Линейные структуры: Санкт-Петербург: Лань, 2022 (2) Л. А. Павлов, Н. В. Первова. . Структуры и алгоритмы обработки данных: СПб.: Лань, 2021 (1, 4, 5)	6
Написание программного кода	Л. А. Павлов, Н. В. Первова. . Структуры и алгоритмы обработки данных: Санкт-Петербург: Лань, 2021 (1, 4, 5)	6
Оформление отчета по ПР	Р. Л. Круз. . Структуры данных и проектирование программ: М.: БИНОМ. Лаборатория знаний, 2008 (7-11)	2
Итого по разделу 4		30

## ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств, позволяющие оценить результаты обучения по данной дисциплине, включают в себя:

- диагностическая работа
- вопросы к экзамену;
- отчет по практическому заданию;
- устный опрос студентов;
- экзамен.

### Критерии оценивания

#### Диагностическая работа

Диагностическая работа проводится в форме теста в ЭИОС Moodle:

- при правильном ответе менее чем на 60% вопросов - не аттестация;
- при правильном ответе на 60% вопросов и более - аттестация.

#### Вопросы к экзамену

Список вопросов размещен в УМК дисциплины.

#### Отчет по практическому заданию

Шаблоны отчетов по практическим работам, а также критерии оценки отчетов приведены в соответствующих разделах курса в ЭИОС moodle

#### Устный опрос студентов

Оценка "отлично" выставляется, если при ответе на вопросы показано глубокое понимание материала.

Оценка "хорошо" выставляется, если показано понимание материала, хотя при ответе на некоторые вопросы допущены ошибки.

Оценка "удовлетворительно" выставляется, если даны правильные ответы хотя бы на половину вопросов.

Перевод оценок за устный опрос в баллы приведен в технологической карте.

Вопросы приведены в файлах с практическими заданиями, а также в УМК дисциплины.

#### Экзамен

Итоговая оценка "удовлетворительно" выставляется при наборе обучающимся 51 балла и более за выполненные ИДЗ, оформленные по ним отчеты, диагностические работы, ответы на устные вопросы. Порядок начисления баллов прописан в технологической карте.

Оценка "хорошо" выставляется при наборе обучающимся 75-84 баллов, набранных обучающимся за выполненные ИДЗ, оформленные по ним отчеты, диагностические работы, ответы на устные вопросы.

Оценка "отлично" выставляется при наборе 85 и более баллов, набранных обучающимся за выполненные ИДЗ, оформленные по ним отчеты, диагностические работы, ответы на устные вопросы.

В случае несогласия с оценкой по сумме баллов обучающийся имеет право сдавать экзамен в традиционной форме. Экзаменационный билет включает в себя два теоретических вопроса и задачу на написание программы.

Оценка "отлично" выставляется, если программа работоспособна и написана грамотно, а при ответе на теоретические вопросы показано глубокое понимание материала.

Оценка "хорошо" выставляется, если программа работоспособна, но написана безграмотно, при ответе на теоретические вопросы обнаруживаются пробелы в знаниях.

Оценка "удовлетворительно" выставляется, если обучающийся не может написать программу, но хотя бы может осознанно выбрать структуру данных и структуру хранения, а также алгоритм решения задачи, при ответе на теоретические вопросы обнаруживаются большие пробелы в знаниях, но обучающийся может более-менее разумно ответить хотя бы на один вопрос билета.

"Не сдано" выставляется, если в течение семестра обучающийся набрал менее 51 балла и за отведенное на проведение экзамена время не смог придумать решение задачи.

Паспорт фонда оценочных средств

КУРС	СЕМЕСТР	Наименование разделов и дидактических единиц	ВСЕГО	Аудиторные занятия в контактной форме			Самостоятельная работа студентов	Формируемая компетенция, %		НАИМЕНОВАНИЕ ОЦЕНОЧНОГО СРЕДСТВА
				ВСЕГО	Лекции	Практические занятия		ОПК-6	ПК-94	
1	2	Раздел 1. Структуры данных и структуры хранения. Линейные структуры данных.	21	6	4	2	15	20	20	Вопросы к экзамену, Отчет по практическому заданию
1	2	Раздел 2. Линейные структуры данных.	24	6	4	2	18	20	20	Вопросы к экзамену, Отчет по практическому заданию, Устный опрос студентов
1	2	Раздел 3. Нелинейные структуры данных.	49	19	14	5	30	30	30	Вопросы к экзамену, Отчет по практическому заданию, Устный опрос студентов
1	2	Раздел 4. Алгоритмы сортировки и поиска.	50	20	12	8	30	30	30	Вопросы к экзамену, Отчет по практическому заданию, Устный опрос студентов
Всего за 2 семестр			144	51	34	17	93	100	100	
Всего по дисциплине			144	51	34	17	93	100	100	



## Критерии оценивания

### ОПК-6

- Вопросы открытого типа:*
- № 1 Что такое структура хранения?
- № 2 Элемент списка объявлен на языке Си следующим образом:
- ```
struct list  
{  
    long a;  
    list *next;  
};
```
- Переменная first содержит адрес первого элемента списка. Что определяется в приведенном ниже фрагменте программы?
- ```
for (s = 0, p = first; p != NULL; s++, p = p->next);
```
- № 3 Как чаще всего организуются очереди с приоритетом?
- № 4 Опишите на естественном языке алгоритм обхода бинарного дерева в прямом порядке (сверху вниз)
- № 5 По входной последовательности 5, 1, 3, 10, 2, 12, 15, 7, 4 постройте дерево бинарного поиска и обойдите его слева направо. Какая последовательность получится в процессе обхода?
- № 6 Для упорядочения массива { 5, 12, 24, 14, 7, 2, 3, 9, 1, 18 } была выбрана быстрая сортировка. Какой будет последовательность элементов в массиве после первого прохода по массиву, если в качестве опорного выбран пятый элемент массива (7), а массив должен быть упорядочен по возрастанию?
- № 7 Для каждой структуры данных: очередь с приоритетом, стек, очередь, дек – укажите количество определённых для нее основных операций и сами операции.
- № 8 Что называется высотой дерева?
- № 9 Какое бинарное дерево называется сбалансированным?
- № 10 Какие алгоритмы называют жадными?
- Вопросы закрытого типа:*
- № 1 Какие утверждения справедливы относительно внутренних методов поиска?
- Эти методы применимы только к наборам упорядоченных данных
  - Эти методы применимы только к массивам
  - Эти методы применимы к данным, полностью размещенным в оперативной памяти
  - Главный критерий в этих методах – число выполняемых сравнений
- № 2 Как определить количество элементов двусвязного списка?
- Вычесть из адреса последнего элемента списка адрес первого элемента
  - Подсчитать количество повторов в цикле перебора всех элементов списка
  - Вычесть из адреса последнего элемента списка адрес первого элемента и прибавить 1
  - Вычесть из адреса последнего элемента списка адрес первого элемента, полученную разность разделить на размер одного элемента
  - В двусвязном списке всегда только 2 элемента
- № 3 Какие утверждения относительно связной реализации списков являются правильными?
- Каждый элемент списка имеет специальное поле с адресом следующего элемента
  - Каждому элементу списка во время выполнения программы выделяется своя область памяти
  - Логический порядок следования элементов в списке может не совпадать с физическим размещением элементов в памяти
  - Максимальное число элементов в списке должно быть известно заранее

- № 4 Какие из перечисленных утверждений, касающихся структур хранения, являются верными?
- При одном и том же объеме занятой памяти в связную структуру можно записать меньше элементов, чем в векторную
  - Добавление крайних элементов списка осуществляется быстрее при использовании связной структуры хранения
  - При использовании векторной структуры хранения невозможно организовать эффективный поиск
  - Добавление элементов в середину списка осуществляется быстрее при использовании связной структуры хранения
  - С помощью связных структур хранения можно реализовать только линейную структуру данных
- № 5 Какие утверждения справедливы для векторной реализации очереди?
- Индекс головы непустой очереди всегда не больше индекса хвоста очереди
  - При добавлении элементов в очередь и извлечении элементов из очереди требуется выполнить сдвиг элементов массива
  - Очередь организуется с помощью циклического массива
  - Максимальное количество элементов очереди ограничено объемом свободной оперативной памяти
  - При отсутствии счетчика элементов определить, заполнена очередь или пуста, можно по взаимному расположению индексов головы и хвоста очереди
- № 6 Какие утверждения справедливы относительно понятия "пирамида"?
- Пирамида – разновидность поискового дерева
  - Пирамида – разновидность двоичного дерева
  - В пирамиде или ключи любой вершины не больше ключей всех потомков этой вершины, или ключи любой вершины не меньше ключей всех потомков этой вершины
  - При обходе пирамиды в прямом порядке можно получить возрастающую или убывающую последовательность ключей
- № 7 Поставьте в соответствие порядку действий название алгоритма обхода дерева
- 1) Сначала обрабатывается левое поддерево, потом правое поддерево, потом корень дерева
  - 2) Сначала обрабатывается корень поддерева, потом элементы первого уровня, потом элементы второго уровня и т.д.
  - 3) Сначала обрабатывается корень дерева, потом левое поддерево, потом правое поддерево
  - 4) Сначала обрабатывается левое поддерево, потом корень дерева, потом правое поддерево
- А) Прямой обход  
Б) Симметричный обход  
В) Поуровневый (поперечный) обход  
Г) Обратный обход
- № 8 Вычислительная сложность каких алгоритмов оценивается как  $O(n)$ ?
- Алгоритм сортировки простыми вставками
  - Добавление элемента в упорядоченный список
  - Построение АВЛ-дерева из упорядоченной последовательности
  - Построение декартова дерева из упорядоченной последовательности
  - Построение пирамиды
- № 9 При выполнении какого условия из перечисленных метод сортировки является неустойчивым?
- Не всегда получается упорядоченная последовательность
  - Изменяется относительное расположение элементов с равными ключами

- Не меняется относительное расположение элементов с равными ключами
- Может получиться как возрастающая, так и убывающая последовательность

№ 10 Какие утверждения справедливы для поразрядной сортировки?

- Поразрядная сортировка эффективна только для массивов, содержащих большое число элементов
- Поразрядная сортировка массива требует  $O(n+m)$  дополнительной памяти, где  $m$  - мощность алфавита
- Худшим случаем для поразрядной сортировки является обратно упорядоченная последовательность ключей
- Устойчивость поразрядной сортировки зависит от типа сортируемых ключей
- Поразрядная сортировка не использует сравнение ключей между собой

#### ПК-94

##### *Вопросы открытого типа:*

- № 1 Чему равна максимальная высота префиксного дерева?
- № 2 При добавлении элемента в какое бинарное дерево поиска используется метод ножниц и клея?
- № 3 Какое условие должно выполняться, чтобы алгоритм бинарного поиска работал правильно?
- № 4 Какова верхняя асимптотическая оценка трудоемкости поиска в красно-черном дереве?
- № 5 Если никакие две записи в файле не имеют одинакового значения ключа, то такой ключ называется \_\_\_\_\_ ключом.
- № 6 Какова верхняя асимптотическая оценка сложности поиска элемента в неупорядоченном списке?
- № 7 Какая структура данных предназначена для быстрого доступа к максимальному (минимальному) элементу?
- № 8 Для чего используется индекс при индексно-последовательном поиске?
- № 9 Проведите сравнительный анализ реализаций ассоциативного массива хеш-таблицей и сбалансированным бинарным деревом поиска и укажите, в каких случаях какой вариант предпочтительнее.
- № 10 Какова цель сортировки данных?

##### *Вопросы закрытого типа:*

- № 1 Когда улучшенные методы сортировки имеют значительное преимущество в сравнении с базовыми методами?
- При большом количестве сортируемых элементов
  - Когда массив обратно упорядочен
  - При малых количествах сортируемых элементов
  - В любом случае
- № 2 Какие методы сортировки массивов (из перечисленных) относятся к улучшенным?
- Быстрая сортировка
  - Сортировка прямым слиянием
  - Пирамидальная сортировка
  - Поразрядная сортировка
- № 3 Какие из перечисленных методов сортировки массивов являются устойчивыми?
- Быстрая сортировка
  - Поразрядная сортировка
  - Сортировка Шелла
  - Сортировка прямым слиянием
  - TimSort
- № 4 Поставьте в соответствие улучшенному алгоритму базовый, улучшением которого он является.

- 1 – Сортировка Шелла  
2 – Быстрая сортировка  
3 – Пирамидальная сортировка
- А – Сортировка простыми вставками  
Б – Сортировка пузырьком  
В – Сортировка выбором  
Г – Сортировка бинарными вставками
- № 5 Какие преимущества имеют двусвязные линейные списки в сравнении с односвязными списками?
- Простой переход от текущего элемента к любому из двух соседей
  - Меньшие затраты памяти
  - Возможность прохода и поиска как в прямом, так и в обратном направлениях
  - Меньшее количество операций при добавлении и удалении элементов
- № 6 Укажите алгоритмы, которые лежат в основе сортировки TimSort
- Сортировка вставками
  - Алгоритм бинарного поиска
  - Сортировка естественным слиянием
  - Сортировка прямым слиянием
  - Быстрая сортировка
  - Разделение ключа на разряды
- № 7 Какие методы поиска относятся к внутренним?
- Методы двоичного поиска
  - Методы последовательного перебора
  - Методы поиска с использованием Б-деревьев
  - Методы хеширования
- № 8 Какие утверждения справедливы для splay-дерева?
- Splay-дерево является бинарным деревом поиска
  - Основная операция splay-дерева `splay()` представляет собой последовательность `zig`, `zig-zig` и `zig-zag` поворотов
  - Для splay-дерева устанавливается коэффициент жесткости, который позволяет регулировать его баланс
  - Максимальная высота splay-дерева не превышает  $4 \log N$
  - Splay-дерево наиболее эффективно используется для поиска в системах, в которых вероятность поиска различных ключей одинакова.
  - Splay-дерево изменяет свою структуру после каждой операции
- № 9 Выберите верные утверждения
- Неуспешный поиск (когда искомого элемента в таблице нет) всегда требует больше операций сравнения, чем успешный (когда искомым элемент есть)
  - Бинарный поиск не применим к связным спискам, даже если элементы в списке упорядочен
  - Асимптотическая сложность алгоритмов поиска и вставки в связных списках и в массивах одинакова
  - Поиск элемента в массиве по внутреннему ключу всегда имеет асимптотическую сложность  $O(n)$
- № 10 Выберите утверждения, справедливые для префиксного дерева/
- Префиксное дерево не может быть бинарным
  - Максимальная высота префиксного дерева не зависит от количества размещенных в нем ключей
  - При поиске по префиксному дереву не происходит сравнений ключей
  - Префиксное дерево не хранит в узлах значения ключей
  - Префиксное дерево называется также нагруженным деревом

