

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ



УТВЕРЖДАЮ  
Декан ФИиВТ

УТВЕРЖДАЮ /А.А. Кречетов/  
(Ф.И.О. декана (директора института))

31.01.2023 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

С.1.1.17 Технология и методы программирования

*(код и наименование дисциплины по учебному плану)*

Направление подготовки (специальность) 10.05.03 Информационная безопасность автоматизированных систем

Квалификация выпускника Специалист  
(бакалавр/магистр/специалист)

Специализация Анализ безопасности информационных систем

Курс 2  
Семестр 3, 4

**Распределение учебного времени**

Трудоемкость по учебному плану	288 / 8	часов/зачетных единиц
Лекции	36	часов
Лабораторные работы	108	часов
Практические занятия	-	часов
Иная контактная работа	-	часов
Всего контактной работы (без учета экз.)	144	часов
Контактная работа по экзамену	6	часов
Курсовой проект (работа)	-	семестр
Самостоятельная работа обучающихся (без учета экз.)	108	часов
Самостоятельная работа по подготовке к экзамену	30	часов
Экзамен	4	семестр
Зачет	3	семестр
БРК, ДЗ	-	семестр

(год)

Программа составлена в соответствии с требованиями ФГОС ВО направления подготовки (специальности) 10.05.03 Информационная безопасность автоматизированных систем

Программу составили:

доцент с ученой степенью кандидата наук	ИБ	СОГЛАСОВАНО	О.Л. Сорокин
(должность)	(кафедра)		(И.О. Фамилия)

РАССМОТРЕНА и ОДОБРЕНА на заседании кафедры, за которой закреплена дисциплина  
Кафедра информационной безопасности

	(наименование кафедры)	
31.01.2023	протокол №	10/1
(дата)		
Заведующий кафедрой	СОГЛАСОВАНО	И.Г. Сидоркина
		(И.О. Фамилия)

Рабочая программа СОГЛАСОВАНА с факультетом (институтом), выпускающей(ими)  
кафедрой(ами).  
СООТВЕТСТВУЕТ действующей ОП.

Заведующий кафедрой	СОГЛАСОВАНО	И.Г. Сидоркина
		(И.О. Фамилия)

Председатель методической комиссии факультета (института), в который входит  
выпускающая кафедра

	СОГЛАСОВАНО	А.А. Кречетов
		(И.О. Фамилия)

Эксперт(ы): Зверева Екатерина Васильевна, Начальник отдела ПД ИТР ОАО ММЗ

Рабочая программа проверена и зарегистрирована в УМЦ 22.02.2023 г.  
Специалист учебно-методического центра СОГЛАСОВАНО /И.Р. Валиева/

## Раздел 1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является достижение планируемых результатов обучения, соответствующих установленным в ОПОП индикаторам достижения компетенций:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Результаты обучения
1. ОПК-7 Способен создавать программы на языках общего назначения, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ	ОПК-7.1 знает язык программирования высокого уровня (объектно-ориентированное программирование)	<b>знания:</b> знает язык программирования высокого уровня (объектно-ориентированное программирование) <b>умения:</b> <b>навыки:</b>
	ОПК-7.2 умеет работать с интегрированной средой разработки программного обеспечения умеет разрабатывать и реализовывать на языке высокого уровня алгоритмы решения типовых профессиональных задач	<b>знания:</b> <b>умения:</b> умеет работать с интегрированной средой разработки программного обеспечения умеет разрабатывать и реализовывать на языке высокого уровня алгоритмы решения типовых профессиональных задач <b>навыки:</b>
	ОПК-7.3 Разработка программного обеспечения, технических средств, баз данных и компьютерных сетей с учетом требований по обеспечению защиты информации	<b>знания:</b> Знает, как разработать программные обеспечения, технические средства, базы данных и компьютерные сети с учетом требований <b>умения:</b> Умеет разрабатывать программные обеспечения, технические средства, базы данных и компьютерные сети <b>навыки:</b> Разрабатывает программное обеспечение, технические средства, базы данных и компьютерных сетей с учетом требований по обеспечению защиты информации

## Раздел 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина относится к обязательной части ОПОП.

Дисциплина является обязательной

Для продолжения формирования заявленных компетенций необходимы знания предшествующих дисциплин: Языки программирования (ОПК-7)

Изучаемая дисциплина является основой для продолжения формирования указанных компетенций в следующих дисциплинах: Безопасность операционных систем (ОПК-7), Программирование на языках высокого уровня (ОПК-7), Объектно-ориентированное программирование (ОПК-7); государственной итоговой аттестации в форме: Подготовка к сдаче и сдача государственного экзамена (ОПК-7)

## Раздел 3. ОПИСАНИЕ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ

Для формирования заявленных компетенций используются методологические технологии, реализующие деятельностный, личностно-ориентированный, практико-ориентированный подходы.

Основными стратегическими технологиями являются: лекционные занятия, практические и лабораторные занятия

На достижение конкретных целей обучения направлены применяемые тактические технологии: задания, классическая лекция

#### Раздел 4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

##### 3 семестр

Виды и темы занятий	Количество часов	Формируемые компетенции
<b>Линейные списки</b>	<b>23</b>	ОПК-7
Лекция. Статические и динамические структуры данных	1	
Лекция. Стеки, очереди, деки	1	
Лекция. Преобразование выражений на основе стеков	1	
Лабораторная работа. Линейные списки	12	
Задания для самостоятельной работы, в том числе выполнение 1. Проработка лекций 2. Подготовка к лабораторным работам 3. Освоение способов организации списков на основе статических и динамических структур данных 4. Изучение контейнерных классов языка C++ для организации и обработки списков. 5. Преобразование в постфиксную форму простых выражений	8	
<b>Деревья</b>	<b>24</b>	ОПК-7
Лекция. Организация деревьев в памяти	1	
Лекция. Алгоритмы обхода деревьев	1	
Лабораторная работа. Деревья	12	
Задания для самостоятельной работы, в том числе выполнение 1. Проработка лекций 2. Подготовка к лабораторным работам 3. Освоение схем рекурсивного обхода деревьев. 4. Определение веса дерева путем использования рекурсии и стека.	10	
<b>Графы</b>	<b>29</b>	ОПК-7
Лекция. Представление графов	1	
Лекция. Обходы в глубину и ширину	1	
Лекция. Поиск кратчайших путей	1	
Лекция. Остовные деревья	1	
Лекция. Связность графов	1	
Лабораторная работа. Графы	14	

Задания для самостоятельной работы, в том числе выполнение 1. Проработка лекций 2. Подготовка к лабораторным работам 3. Изучить достоинства и недостатки различных представлений графа. 4. Освоение алгоритмов DFS и BFS – обхода графа в глубину и ширину. 5. Изучение алгоритмов Дейкстры и Флойда на примерах тестовых графов. 6. Проведение сравнение вычислительной сложности алгоритмов Прима и Краскала для полных и разреженных графов	10	
<b>Поиск и сортировка данных</b>	<b>32</b>	ОПК-7
Лекция. Последовательный и бинарный поиск. Деревья поиска.	1	
Лекция. Балансировка деревьев поиска	1	
Лекция. Б-деревья	1	
Лекция. Хеширование	1	
Лекция. Методы внутренней сортировки	2	
Лекция. Методы внешней сортировки	2	
Лабораторная работа. Поиск и сортировка данных	16	
Задания для самостоятельной работы, в том числе выполнение 1. Проработка лекций 2. Подготовка к лабораторным работам 3. Сравнение характеристик методов поиска данных. 4. Освоение методов поиска данных во внешней памяти. 5. Освоение быстрой сортировки из библиотеки STL в языке C++. 6. Оценка целесообразность применения усовершенствованных методов внешней сортировки в зависимости от объема данных 7. Выполнение тестов для самоконтроля	8	
Иная контактная работа:	0	

#### 4 семестр

Виды и темы занятий	Количество часов	Формируемые компетенции
<b>Вычислительная сложность алгоритмов</b>	<b>24</b>	ОПК-7
Лекция. Оценка трудоемкости алгоритмов	1	
Лекция. NP-полные и NP-трудные задачи	1	
Лекция. Проведение и разбор теста	2	
Задания для самостоятельной работы, в том числе выполнение 1. Проработка лекций 2. Подготовка к лабораторным работам 3. Изучение примеров оценки сложности	20	
<b>Перестановки и подмножества</b>	<b>50</b>	ОПК-7
Лекция. Смешанные системы счисления. Алгоритмы генерации и выбора перестановок	2	
Лекция. Коды Грея. Подмножества множеств	2	
Лабораторная работа. Перестановки и подмножества	24	

Задания для самостоятельной работы, в том числе выполнение 1. Проработка лекций 2. Подготовка к лабораторным работам 3. Оценка сложности алгоритмов с перестановками 4. Применение кодов Грея	22	
<b>Технологии программирования</b>	<b>70</b>	ОПК-7
Лекция. Проекты разработки программного обеспечения	2	
Лекция. Модели жизненного цикла проекта программных проектов	2	
Лекция. Технологии разработки проекта	2	
Лекция. Документирование проекта. Стандарты	2	
Лекция. Сопровождение проекта. Тестирование и отладка.	2	
Лабораторная работа. Проектирование программной системы	30	
Задания для самостоятельной работы, в том числе выполнение 1. Проработка лекций 2. Подготовка к лабораторным работам 3. Сравнение методов разработки программного проекта	30	
Иная контактная работа:	0	
Подготовка к экзамену	30	
Проведение экзамена	6	

## Раздел 5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Изучение дисциплины рекомендуется начать с ознакомления с рабочей программой, ее структурой и содержанием разделов. Учебный материал структурирован, изучение дисциплины осуществляется в тематической последовательности.

**Занятия лекционного типа** дают систематизированные знания по дисциплине, концентрируют внимание на наиболее сложных и важных вопросах. Во время лекционных занятий рекомендуется вести конспектирование учебного материала; обращать внимание на формулировки и категории, раскрывающие суть проблемы, явления или процесса; зафиксировать выводы и практические рекомендации. Подготовка к занятиям **семинарского типа** включает ознакомление с планом лабораторного занятия; работу с конспектом лекций, выполнение домашнего задания, работу с учебной и учебно-методической литературой, научными изданиями и электронными образовательными ресурсами, рекомендованными рабочей программой дисциплины.

Содержание **самостоятельной работы** определяется рабочей программой дисциплины, оценочными и методическими материалами, заданиями и указаниями преподавателя. Самостоятельная работа может осуществляться в аудиторной и внеаудиторной формах. Эффективным средством осуществления самостоятельной работы является электронная информационно-образовательная среда университета, которая обеспечивает доступ к образовательной программе, рабочей программе дисциплины, к электронным библиотечным системам, профессиональным базам данных и информационным справочным системам.

Периодичность проведения, формы текущего контроля успеваемости, система оценивания хода освоения дисциплин представлены в рабочей программе. Формой промежуточной аттестации по дисциплине является зачет в третьем семестре,

## Раздел 6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 6.1. Учебно-методическое обеспечение

№№ п/п	Список используемой литературы	Количество экземпляров печатных изданий, имеющих в библиотеке, или электронный адрес издания (ресурса) в сети Интернет
<b>УЧЕБНЫЕ, УЧЕБНО-МЕТОДИЧЕСКИЕ И НАУЧНЫЕ ИЗДАНИЯ</b>		
1.	Давыдов, Владимир Григорьевич. Технологии программирования C++ [Текст] : [учеб. пособие для студентов вузов по специальности 210100 "Упр. и информатика в техн. системах"] / В. Давыдов. СПб.: БХВ-Петербург, 2005. - 654 с. ISBN 5-94157-605-6. Экземпляры: всего 17.	17
2.	Сысоев, А. В. Технологии программирования на базе Microsoft Solutions Framework [Электронный ресурс] / Сысоев А. В., Козинев Е. А., Мееров И. Б. 2-е изд. Москва: ИНТУИТ, 2016. - 154 с.	<a href="https://e.lanbook.com/book/100518">https://e.lanbook.com/book/100518</a>
3.	Галочкин, Владимир Иванович. Структуры и алгоритмы обработки данных [Текст] : учеб. пособие / В. И. Галочкин. Йошкар-Ола: МарГТУ, 2004. - 147 с. ISBN 5-8158-0350-2. Экземпляры: всего 58.	58
4.	Галочкин, Владимир Иванович. Алгоритмы и программы [Текст] : задачи повышенной сложности : учеб. пособие / В. И. Галочкин; М-во образования и науки Рос. Федерации, ФГБОУ ВПО "Мар. гос. техн. ун-т". Йошкар-Ола: МарГТУ, 2012. - 207 с. ISBN 978-5-8158-0968-0. Экземпляры: всего 88.	88 / <a href="https://portal.volgatech.net/books/Galochkin_Algoritmy_i_programmy.pdf">https://portal.volgatech.net/books/Galochkin_Algoritmy_i_programmy.pdf</a>
5.	Серебряков, В. А. Теория и реализация языков программирования [Электронный ресурс] / Серебряков В. А., Галочкин М. П., Гончар Д. Р., Фуругян М. Г. 2-е изд. Москва: ИНТУИТ, 2016. - 372 с.	<a href="https://e.lanbook.com/book/100529">https://e.lanbook.com/book/100529</a>
<b>ЭЛЕКТРОННЫЕ ОБРАЗОВАТЕЛЬНЫЕ РЕСУРСЫ</b>		
1.	Научная электронная библиотека eLIBRARY.RU	<a href="http://elibrary.ru">http://elibrary.ru</a>
2.	Научная электронная библиотека «Киберленинка»	<a href="http://cyberleninka.ru">http://cyberleninka.ru</a>
<b>ПРОФЕССИОНАЛЬНЫЕ БАЗЫ ДАННЫХ И ИНФОРМАЦИОННЫЕ СПРАВОЧНЫЕ СИСТЕМЫ</b>		
1.	Справочно-правовая система Консультант+	<a href="http://www.consultant.ru">http://www.consultant.ru</a>
2.	Информационно-правовой портал Гарант	<a href="http://www.garant.ru">http://www.garant.ru</a>
3.	Профессиональные справочные системы Техэксперт	<a href="http://www.cntd.ru">http://www.cntd.ru</a>

## 6.2. Материально-техническая база и программное обеспечение

№№ п/п	Аудитории для проведения учебных занятий, самостоятельной работы и проведения государственной итоговой аттестации	Перечень основного оборудования	Программное обеспечение
1.	111б (III)	Комплект учебной мебели (1)	Microsoft Windows Enterprise, Справочная правовая система "Консультант Плюс",

		Microsoft Office Standard, Агент Dr.Web, Комплект ГАРАНТ-Мастер, Microsoft Access, Microsoft Visio Professional, Microsoft Project Professional, Microsoft Visual Studio Enterprise, Комплект ПО для решения основных пользовательских задач
--	--	--

## Раздел 7. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ/ ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Критерии оценивания индикаторов достижения компетенций направлены на:

- усвоение теоретического материала (объем знаний, глубина усвоения), предусмотренного рабочей программой;
- умение излагать материал (четкость, грамотность изложения материала, точность и полнота воспроизведения учебного материала);
- умение применять теоретические знания при решении практических заданий.

Шкала оценивания представлена ниже.

Уровень сформированности элементов компетенции	Критерии оценивания	Шкала оценивания
Пороговый уровень	Обучающийся имеет знания основного материала, проявляет умение логично его излагать, но может допускать неточности в изложении материала, недостаточно правильные формулировки, испытывает затруднения в выполнении практических заданий.	удовлетворительно
Продвинутый уровень	Обучающийся твердо знает программный материал, излагает его грамотно и по существу, не допускает существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения	хорошо
Высокий уровень	Обучающийся глубоко и прочно усвоил программный материал, грамотно и логически стройно его излагает, дает исчерпывающие ответы на поставленные вопросы. В ответе тесно увязывается теория с практикой, при этом обучающийся не затрудняется с ответом при видоизменении задания, свободно справляется с задачами, вопросами и другими видами применения знаний, показывает знакомство с монографической литературой, периодическими изданиями, правильно обосновывает принятые решения, свободно владеет разносторонними навыками, приемами выполнения практических работ	отлично

### 7.1. Текущий контроль успеваемости

Текущий контроль успеваемости обеспечивает оценивание хода освоения дисциплины

(модуля) и производится с применением технологии рейтингового контроля в соответствии с технологической картой дисциплины. Порядок составления технологической карты и алгоритм проведения процедуры оценивания видов деятельности обучающихся, направленных на освоение знаний, умений, навыков и/или опыта деятельности, по накопительной системе в баллах устанавливается положением о системе РИТМ в ФГБОУ ВО «ПГТУ»

## 7.2. Промежуточная аттестация обучающихся

Промежуточная аттестация обучающихся направлена на оценивание результатов обучения по дисциплине (модулю) и проводится с использованием фондов оценочных средств.

Примеры типовых контрольных заданий из базы фонда оценочных средств по образовательной программе.

1. Дайте определение типа данных.
2. Где и как применяется тип данных “множество”?
3. Как представляются и реализуются множества в разных языках программирования?
4. Какова область применения текстовых файлов? В чем их достоинства и недостатки?
5. Укажите достоинства и недостатки файлов прямого доступа.
6. Как применяются команды SEEK в Паскале и fseek в C++?
7. Что такое статическое представление данных в языках программирования?
8. Что такое динамическое представление данных в языках программирования?
9. Дайте определение линейного списка.
10. Опишите применение стеков в программировании.
11. Укажите основные операции со стеками.
12. Как представляются стеки в языках программирования?
13. Опишите применение очередей в программировании.
14. Укажите основные операции со очередями.
15. Как представляются очереди в языках программирования?
16. Укажите достоинства двусвязных списков.
17. Где применяются кольцевые списки?
18. Что такое мультисписки?
19. Дайте определение дерева.

20. Что такое обход дерева? Назовите основные типы обходов.
21. Как применяются рекурсивные функции при работе с деревьями?
22. Опишите области применения бинарных деревьев.
23. В чем удобство представления алгебраических выражений в префиксной, постфиксной и инфиксной формах с помощью бинарных деревьев?
24. Что такое И-ИЛИ дерево?
25. Дайте определение графа. Какие виды графов Вы знаете?
26. Опишите способы представления графов в языках программирования.
27. Как реализуется обход графа в глубину?
28. Как реализуется обход графа в ширину?
29. Как применяются стеки, очереди и рекурсия при обходах графа?
30. Как по поиску в глубину перечислить все пути графа, ведущие из начальной вершины в конечную?
31. Оцените объем вычислений при поиске в глубину всех путей графа.
32. Назовите основные шаги алгоритма Дейкстры поиска кратчайших путей.
33. Приведите пример работы алгоритма Дейкстры на тестовом графе.
34. Назовите основные шаги алгоритма Флойда поиска кратчайших путей.
35. Приведите пример работы алгоритма Флойда на тестовом графе.
36. Проведите сравнительный анализ алгоритмов Дейкстры и Флойда.
37. Как реализовать поиск циклов на графе без их повторения?
38. Что такое каркасные деревья?
39. Опишите алгоритм Прима нахождения минимального каркасного дерева.
40. Опишите алгоритм Краскала нахождения минимального каркасного дерева.
41. Проведите сравнительный анализ трудоемкости алгоритмов Прима и Краскала.
42. В чем особенности реализации алгоритма Прима?
43. В чем особенности реализации алгоритма Краскала?

44. Перечислите основные задачи поиска данных.
45. Опишите области применения последовательного и индексно-последовательного методов поиска данных.
46. Что такое бинарный поиск? Каковы разновидности бинарного поиска?
47. Дайте определение бинарного дерева поиска.
48. Как в бинарное дерево поиска включаются новые записи?
49. Как производится удаление записей из бинарного дерева поиска?
50. Зачем нужна балансировка деревьев поиска?
51. Назовите уровни балансировки бинарных деревьев поиска.
52. Как происходит включение записей в AVL-дерево?
53. Как происходит удаление записей из AVL-дерева?
54. Дайте определение Б-дерева.
55. Как производится включение записей в Б-дерево?
56. Как происходит удаление записей из Б-дерева?
57. Приведите пример индексации записей на основе Б-дерева.
58. Опишите требования к функции хеширования.
59. Какие существуют способы разрешения коллизий?
60. Как производится удаление записей при хешировании?
61. Назовите недостатки и достоинства хеширования.
62. Укажите требования к внутренней и внешней сортировкам.
63. Перечислите простые методы внутренней сортировки.
64. Какова оценка трудоемкости сортировки Шелла?
65. Дайте определение пирамиды.
66. Опишите алгоритм Флойда начального построения пирамиды.
67. Покажите на примере первые 3 шага пирамидальной сортировки.
68. На чем основана быстрая сортировка Хоара?

69. Каковы оценки трудоемкости среднего и худшего случаев в сортировке Хоара?
70. Как производится слияние отсортированных последовательностей?
71. Приведите пример 3-ленточной 2-фазной сортировки простым слиянием.
72. Как уменьшить трудоемкость сортировки простым слиянием за счет использования дополнительных лент?
73. Приведите преимущества сортировки естественным слиянием.
74. Почему в сортировке естественным слиянием количество серий после разделения на 2 ленты может существенно отличаться?
75. Приведите пример, когда целесообразно применение многофазной сортировки.
76. Как совместно использовать методы внутренней и внешней сортировок?
77. Как оценивается вычислительная сложность алгоритмов? Приведите примеры.
78. Что такое вычислительная сложность задачи?
79. Как определяется полиномиальная и экспоненциальная сложность алгоритмов?
80. Приведите примеры NP-полных алгоритмов.

Высокий уровень

81. Приведите пример смешанной системы счисления.
82. Дайте определение перестановки.
83. Опишите алгоритм перечисления перестановок в лексикографическом порядке.
84. Как по перестановке определяется вектор инверсий?
85. Как по вектору инверсий восстановить перестановку?
86. Опишите способ нумерации перестановок с помощью векторов инверсий.
87. Где используется нумерация перестановок?
88. Как перечислить перестановки путем транспозиции соседних элементов?
89. Опишите способы получения случайных перестановок.
90. Дайте определение циклического кода Грея.

91. Опишите способы генерации кода Грея.
92. Как производится нумерация слов Грея? Для чего она используется?
93. Что такое программный продукт?
94. Какими требованиями определяется качество программной системы?
95. Перечислите этапы проектирования программной системы.
96. Назовите методы проектирования архитектуры программной системы.
97. Приведите примеры гибких технологий разработки программных систем.
98. Назовите способы организации коллектива программистов.
99. Что такое сопровождение программной системы?
100. Что такое стиль программирования?
101. Зачем и кому нужны стандарты программирования?
102. Дайте определения тестирования и отладки программной системы.
103. Сформулируйте основные принципы организации тестирования.
104. Чем отличаются функциональное и структурное тестирование?
105. Назовите этапы тестирования многомодульных программных комплексов.
- Пример нулевого билета:
1. Где и как применяется тип данных "множество"?
2. Как совместно использовать методы внутренней и внешней сортировок?
3. Что такое стиль программирования?

## Перечень вопросов для проведения промежуточной аттестации

Вопросы для зачета:

1. Множества. Способы представление множеств в памяти.
2. Указатели. Применение динамических структур данных.
3. Файлы. Операции над файлами. Текстовые файлы.
4. Линейные списки. Стеки и операции над ними. Применение стеков.

5. Очереди и операции над ними. Организация очередей в памяти и их применение.
6. Двухсвязные и кольцевые списки. Мультисписки.
7. Деревья и способы их организации в памяти.
8. Бинарные деревья. Варианты обхода деревьев. Представление алгебраических выражений с помощью деревьев. Рекурсивные алгоритмы обхода деревьев.
9. Обход деревьев с использованием стека. Леса.
10. Представление графов с помощью матрицы смежности. Матрица достижимости. Алгоритм Уоршела построения транзитивного замыкания.
11. Представление графов на основе списковых структур.
12. Поиск путей на графе в глубину.
13. Поиск путей на графе в ширину.
14. Алгоритм Дейкстры поиска кратчайших путей на графе.
15. Алгоритм Флойда поиска кратчайших путей на графе.
16. Остовные деревья. Алгоритмы Прима и Крускала.
17. Связность графов.
18. Задачи поиска данных. Последовательный поиск. Бинарный поиск.
19. Индексно-последовательный поиск.
20. Бинарные деревья поиска. Поиск с включением.
21. Бинарные деревья поиска. Удаление вершин.
22. AVL-деревья. Балансировка AVL-деревьев.
23. Исключение из AVL-дерева.
24. Б-деревья. Поиск с включением.
25. Б-деревья. Удаление элементов.
26. Хеширование. Разрешение коллизий.
27. Задачи сортировки. Внутренняя и внешняя сортировки.
28. Сортировка Шелла и турнирная сортировка.
29. Пирамидальная сортировка.

30. Быстрая сортировка Хоара.
31. Методы внешней сортировки. Простое слияние.
32. Сортировка методом естественного слияния.
33. Многофазная сортировка. Совмещение внутренней и внешней сортировок.
39. Смешанные системы счисления.
40. Генерация перестановок в лексикографическом порядке.
41. Перечисление и случайный выбор перестановок с помощью векторов инверсий.
42. Перечисление перестановок путем транспозиции соседних элементов.
43. Способы получения случайных перестановок.
44. Перечисление подмножеств заданного множества. Коды Грея.
45. Перечисление подмножеств и выбор случайного подмножества на основе кода Грея.
46. Перечисление К-подмножеств в лексикографическом порядке.
47. Трудоемкость алгоритмов.
48. Понятие программного проекта. Внешний проект.
49. Модели жизненного цикла программного проекта.
50. Структурные технологии разработки программного проекта.
51. Гибкие технологии разработки программного проекта.
52. Задачи управления программным проектом.
53. Документирование программного проекта.
54. Принципы управления рисками
55. Документирование программного проекта.
56. Тестирование программной системы.

Вопросы для экзамена:

## базовый уровень

1. Дайте определение типа данных.
2. Где и как применяется тип данных “множество”?
3. Как представляются и реализуются множества в разных языках программирования?
4. Какова область применения текстовых файлов? В чем их достоинства и недостатки?
5. Укажите достоинства и недостатки файлов прямого доступа.
6. Как применяются команды SEEK в Паскале и fseek в C++?
7. Что такое статическое представление данных в языках программирования?
8. Что такое динамическое представление данных в языках программирования?
9. Дайте определение линейного списка.
10. Опишите применение стеков в программировании.
11. Укажите основные операции со стеками.
12. Как представляются стеки в языках программирования?
13. Опишите применение очередей в программировании.
14. Укажите основные операции со очередями.
15. Как представляются очереди в языках программирования?
16. Укажите достоинства двусвязных списков.
17. Где применяются кольцевые списки?
18. Что такое мультисписки?
19. Дайте определение дерева.
20. Что такое обход дерева? Назовите основные типы обходов.

## Продвинутый уровень

21. Перечислите основные задачи поиска данных.
22. Опишите области применения последовательного и индексно-последовательного методов поиска данных.
23. Что такое бинарный поиск? Каковы разновидности бинарного поиска?

24. Дайте определение бинарного дерева поиска.
25. Как в бинарное дерево поиска включаются новые записи?
26. Как производится удаление записей из бинарного дерева поиска?
27. Зачем нужна балансировка деревьев поиска?
28. Назовите уровни балансировки бинарных деревьев поиска.
29. Как происходит включение записей в AVL-дерево?
30. Как происходит удаление записей из AVL-дерева?
31. Дайте определение Б-дерева.
32. Как производится включение записей в Б-дерево?
33. Как происходит удаление записей из Б-дерева?
34. Приведите пример индексации записей на основе Б-дерева.
35. Опишите требования к функции хеширования.
36. Какие существуют способы разрешения коллизий?
37. Как производится удаление записей при хешировании?
38. Назовите недостатки и достоинства хеширования.
39. Укажите требования к внутренней и внешней сортировкам.
40. Перечислите простые методы внутренней сортировки.
41. Какова оценка трудоемкости сортировки Шелла?
42. Дайте определение пирамиды.
43. Опишите алгоритм Флойда начального построения пирамиды.
44. Покажите на примере первые 3 шага пирамидальной сортировки.
45. На чем основана быстрая сортировка Хоара?
46. Каковы оценки трудоемкости среднего и худшего случаев в сортировке Хоара?
47. Как производится слияние отсортированных последовательностей?
48. Приведите пример 3-ленточной 2-фазной сортировки простым слиянием.
49. Как уменьшить трудоемкость сортировки простым слиянием за счет использования

дополнительных лент?

- 50. Приведите преимущества сортировки естественным слиянием.
- 51. Почему в сортировке естественным слиянием количество серий после разделения на 2 ленты может существенно отличаться?
- 52. Приведите пример, когда целесообразно применение многофазной сортировки.
- 53. Как совместно использовать методы внутренней и внешней сортировок?
- 54. Как оценивается вычислительная сложность алгоритмов? Приведите примеры.
- 55. Что такое вычислительная сложность задачи?
- 56. Как определяется полиномиальная и экспоненциальная сложность алгоритмов?
- 57. Приведите примеры NP-полных алгоритмов.

Высокий уровень

- 58. Приведите пример смешанной системы счисления.
- 59. Дайте определение перестановки.
- 60. Опишите алгоритм перечисления перестановок в лексикографическом порядке.
- 61. Как по перестановке определяется вектор инверсий?
- 62. Как по вектору инверсий восстановить перестановку?
- 63. Опишите способ нумерации перестановок с помощью векторов инверсий.
- 64. Где используется нумерация перестановок?
- 65. Как перечислить перестановки путем транспозиции соседних элементов?
- 66. Опишите способы получения случайных перестановок.
- 67. Дайте определение циклического кода Грея.
- 68. Опишите способы генерации кода Грея.
- 69. Как производится нумерация слов Грея? Для чего она используется?
- 70. Что такое программный продукт?
- 71. Какими требованиями определяется качество программной системы?
- 72. Перечислите этапы проектирования программной системы.

- 73. Назовите методы проектирования архитектуры программной системы.
- 74. Приведите примеры гибких технологий разработки программных систем.
- 75. Назовите способы организации коллектива программистов.
- 76. Что такое сопровождение программной системы?
- 77. Что такое стиль программирования?
- 78. Зачем и кому нужны стандарты программирования?
- 79. Дайте определения тестирования и отладки программной системы.
- 80. Сформулируйте основные принципы организации тестирования.
- 81. Чем отличаются функциональное и структурное тестирование?
- 82. Назовите этапы тестирования многомодульных программных комплексов.