

1 Министерство науки и высшего образования Российской Федерации  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Курганский государственный университет»  
(КГУ)

Кафедра «Автоматизация производственных процессов»

УТВЕРЖДАЮ:  
Первый проректор  
\_\_\_\_\_/Змызгова Т.Р. /  
« \_\_\_\_ » \_\_\_\_\_ 2025 г.

Рабочая программа учебной дисциплины  
**СТРУКТУРЫ И АЛГОРИТМЫ ОБРАБОТКИ ДАННЫХ**

образовательной программы высшего образования –  
программы бакалавриата

**27.03.04 – Управление в технических системах**

Направленность:  
**Автоматика и робототехнические системы**

Формы обучения: очная

Курган 2025

Рабочая программа дисциплины «Структуры и алгоритмы обработки данных» составлена в соответствии с учебными планами по программе бакалавриата «Управление в технических системах» направленность («Автоматика и робототехнические системы»), утвержденными:

- для очной формы обучения 27 \_» 06 2025года.

Рабочая программа дисциплины одобрена на заседании кафедры «Автоматизация производственных процессов» «29» мая 2025 года, протокол № 9

Рабочую программу составил

Доцент

И.А.Иванова

Согласовано:

Заведующий кафедрой  
«Автоматизация производственных  
процессов»

И.А.Иванова

Специалист по учебно-  
методической работе  
Учебно-методического отдела

Г.В. Казанкова

Начальник Управления  
Образовательной деятельности

И.В.Григоренко

## 1. ОБЪЕМ ДИСЦИПЛИНЫ

Всего: 3 зачетных единицы трудоемкости (108 академических часов)

### Очная форма обучения

Вид учебной работы	На всю дисциплину	Семестр
		5
<b>Аудиторные занятия (контактная работа с преподавателем), всего часов</b>	<b>32</b>	<b>32</b>
<b>в том числе:</b>		
Лекции	12	12
Лабораторные работы	20	20
<b>Самостоятельная работа, всего часов</b>	<b>76</b>	<b>76</b>
<b>в том числе:</b>		
Подготовка к зачету	18	18
Другие виды самостоятельной работы (самостоятельное изучение тем (разделов) дисциплины)	58	58
<b>Вид промежуточной аттестации</b>	<b>Зачет</b>	<b>Зачет</b>
<b>Общая трудоемкость дисциплины и трудоемкость по семестрам, часов</b>	<b>108</b>	<b>108</b>

## **2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ**

Дисциплина «Структуры и алгоритмы обработки данных» относится к блоку 1 обязательным дисциплинам части, формируемой участниками образовательных отношений учебного плана подготовки бакалавров.

Изучение дисциплины базируется на результатах обучения, сформированных при изучении следующих дисциплин:

- Математика;
- Информатика;
- Программирование и алгоритмизация.

Результаты обучения по дисциплине необходимы для изучения общеинженерных и специальных технических дисциплин, в частности «Информационные сети и телекоммуникации», «Проектирование автоматизированных систем», «Автоматизированные информационно-управляющие системы» и других профилирующих дисциплин, а также в последующей инженерной деятельности; планировании и проведении научно-исследовательских и проектно-конструкторских работ.

Требования к входным знаниям, умениям, навыкам и компетенциям:

- понимание устной (монологической и диалогической) речи на бытовые и общекультурные темы;
- знание общих теоретических принципов программирования, методов и средств разработки систем автоматизации и управления;
- владение навыками поиска, хранения и обработки цифровой информации с использованием средств компьютерной техники;
- освоение следующих компетенций на уровне не ниже порогового: ОПК-6 (способность осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий) и ОПК-9 (способность использовать навыки работы с компьютером, владеть методами информационных технологий, соблюдать основные требования информационной безопасности).

## **3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ**

Целью освоения дисциплины «Структуры и алгоритмы обработки данных» является: формирование знаний и умений в использовании современных средств программирования и интегрированных систем программирования.

Задачами дисциплины являются: сформировать базовые теоретические понятия, лежащие в основе процесса разработки алгоритмов и структур данных; научить реализации типовых алгоритмов и структур данных и их модификаций на выбранном рабочем языке программирования.

Компетенции, формируемые в результате освоения дисциплины:

- способность осуществлять сбор и анализ исходных данных для расчета и проектирования систем и средств автоматизации и управления (ПК-4);

- готов производить инсталляцию и настройку системного, прикладного и инструментального программного обеспечения систем автоматизации и управления (ПК-15).

В результате изучения дисциплины обучающийся должен:

- Знать: основные типы данных, принципы построения производных структур данных в языках программирования (для ПК-4).

- Знать: основные методы и средства реализации алгоритмов обработки базовых данных и производных структур данных, основные алгоритмы обработки данных (для ПК-15).

- Знать: типовые объекты и структуры данных предоставляемые интегрированными средами программирования для решения производственных задач (для ПК-4).

- Уметь: использовать языки программирования и интегрированную среду для построения структур данных (для ПК-15).

- Уметь: реализовывать алгоритмы и структуры в виде программного кода и готовых программных решений (для ПК-4).

- Уметь: использовать конструкции языка и стандартные библиотеки для создания приложений (для ПК-15).

- Владеть: навыками разработки структур данных и классов (для ПК-4).

- Владеть: навыками использования стандартных алгоритмов обработки данных и структур (для ПК-15).

В рамках освоения дисциплины «Структуры и алгоритмы обработки данных» обучающиеся готовятся к решению следующих профессиональных задач:

- участие в расчетах и проектировании информационных систем средств автоматизации и управления в соответствии с техническим заданием;

- участие в разработке информационных систем по автоматизации производства, управлению бизнес-процессами;

В рамках освоения дисциплины «Структуры и алгоритмы обработки данных» обучающиеся готовятся к исполнению следующих трудовых функций профессионального стандарта:

- Управление проектами в области ИТ на основе полученных планов проектов;

- Выполнение работ по созданию (модификации) и сопровождению информационных систем, автоматизирующих задачи организационного управления и бизнес-процессы;

Индикаторы и дескрипторы части соответствующей компетенции, формируемой в процессе изучения дисциплины «Структуры и алгоритмы обработки данных», оцениваются при помощи оценочных средств.

Планируемые результаты обучения по дисциплине «Структуры и алгоритмы обработки данных», индикаторы достижения компетенций ПК-4, ПК-15,

перечень оценочных средств

№ п/п	Код индикатора достижения компетенции	Наименование индикатора достижения компетенции	Код планируемого результата обучения	Планируемые результаты обучения	Наименование оценочных средств
1.	ИД-1 <sub>ПК-4</sub>	Знать: основные типы данных, принципы построения производных структур данных в языках программирования	З (ИД-1 <sub>ПК-4</sub> )	Знает: основные типы данных, принципы построения производных структур данных в языках программирования	Вопросы для сдачи зачета
2.	ИД-2 <sub>ПК-4</sub>	Уметь: реализовывать алгоритмы и структуры в виде программного кода и готовых программных решений	У (ИД-2 <sub>ПК-4</sub> )	Умеет: реализовывать алгоритмы и структуры в виде программного кода и готовых программных решений	Вопросы для сдачи зачета
3.	ИД-3 <sub>ПК-4</sub>	Владеть: навыками разработки структур данных и классов	В (ИД-3 <sub>ПК-4</sub> )	Владеет: навыками разработки структур данных и классов	Вопросы для сдачи зачета
4.	ИД-1 <sub>ПК-15</sub>	Знать: основные методы и средства реализации алгоритмов обработки базовых данных и производных структур данных, основные алгоритмы обработки данных	З (ИД-1 <sub>ПК-15</sub> )	Знает: основные методы и средства реализации алгоритмов обработки базовых данных и производных структур данных, основные алгоритмы обработки данных	Вопросы для сдачи зачета
5.	ИД-2 <sub>ПК-15</sub>	Уметь: использовать языки программирования и интегрированную среду для построения структур данных	У (ИД-2 <sub>ПК-15</sub> )	Умеет: использовать языки программирования и интегрированную среду для построения структур данных	Вопросы для сдачи зачета
6.	ИД-3 <sub>ПК-15</sub>	Владеть:	В (ИД-3 <sub>ПК-15</sub> )	Владеет:	Вопросы для

		навыками использования стандартных алгоритмов обработки данных и структур		навыками использования стандартных алгоритмов обработки данных и структур	сдачи зачета
--	--	---	--	---	--------------

## 4.СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

### 4.1. Учебно-тематический план

#### Очная форма обучения

Рубеж	Номер раздела, темы	Наименование раздела, темы	Количество часов контактной работы с преподавателем	
			Лекции	Лабораторные работы
Рубеж 1	1	Цели и задачи курса. Основные понятия и определения	1	-
	2	Базовые типы данных языков программирования, преобразование и использование.	1	-
	3	Принципы формирования производных типов данных и структур. Использование шаблонов типа данных	2	2
	4	Алгоритмы сортировки и поиска.	2	8
		Рубежный контроль 1		0,5
Рубеж 2	5	Связные структуры данных. Списки	2	4
	6	Связные структуры данных. Деревья	2	4
	7	Библиотечные реализации структур данных. Другие структуры данных	2	1
		Рубежный контроль 2		0,5
Всего:			12	20

### 4.2. Содержание лекционных занятий

#### *Тема 1. Цели и задачи курса. Основные понятия и определения.*

Цели и задачи курса. Алгоритмы, типы алгоритмов. Основные парадигмы программирования. Понятие о типе данных.

#### *Тема 2. Базовые типы данных языков программирования, преобразование и использование.*

Свойства типа данных. Базовые типы данных языка программирования. Преобразования типов данных, кодирование, машинное представление.

#### *Тема 3. Принципы формирования производных типов данных и структур. Использование шаблонов типа данных.*

Принципы формирования производных типов данных. Формирование массивов, структур данных, комбинирование массивов и структур данных. Использование шаблонов типа данных при проектировании функций (процедур) в языке программирования.

#### ***Тема 4. Алгоритмы сортировки и поиска.***

Понятие внутренней сортировки. «Классические» методы сортировки (пузырьковая, выбором, Шелла). Быстрая сортировка. Процедура разделения. Рекурсивный и не рекурсивный алгоритмы быстрой сортировки. Анализ сложности.

Понятие поиска. Процедуры сравнения данных. Обзор алгоритмов поиска данных. Последовательный поиск в сортированных и несортированных массивах.

#### ***Тема 5. Связные структуры данных. Списки.***

Понятие. Основные определения. Линейный однонаправленный список. Реализация базового класса. Вставка элементов, удаление элементов, поиск элемента, удаление списка. Понятие об итераторах. Реализация класса внешнего итератора. Функции вставки, удаления и поиска элемента списка с использованием внешнего итератора. Двухнаправленные списки.

#### ***Тема 6. Связные структуры данных. Деревья.***

Деревья. Основные понятия и определения. Бинарные деревья. Упаковка бинарных деревьев в массив. Реализация алгоритма пирамидальной сортировки. Бинарный поиск в сортированных массивах.

#### ***Тема 7. Библиотечные реализации структур данных. Другие структуры данных.***

Реализация список и деревьев в стандартной библиотеке C++ и библиотеке .NETCore. Общее представление о графах, hash-таблицах, ассоциативных массивах, кучах.

### **4.3. Лабораторные занятия**

Номер раздела, темы	Наименование раздела, темы	Наименование лабораторной работы	Норматив времени, час.
3	Принципы формирования производных типов данных и структур. Использование шаблонов типа данных	Принципы формирования производных типов данных и структур. Использование шаблонов типа данных	2
4	Алгоритмы сортировки и поиска.	Алгоритмы сортировки	4
		Алгоритмы поиска	2
		Исследование времени работы различных алгоритмов	2
		Рубежный контроль 1	0,5



5	Связные структуры данных. Списки	Реализация класса линейного однонаправленного списка	2
		Реализация внешнего итератора для линейного однонаправленного списка	2
6	Связные структуры данных. Деревья	Алгоритмы поиска и сортировки с использованием деревьев	4
7	Библиотечные реализации структур данных. Другие структуры данных	Реализация структур данных, функций сортировки и поиска в стандартной библиотеке C++	1
		Рубежный контроль 1	0,5
<b>Всего:</b>			<b>20</b>

## 5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

При прослушивании лекций рекомендуется в конспекте отмечать все важные моменты, на которых заостряет внимание преподаватель, в частности те, которые направлены на качественное выполнение соответствующих лабораторных работ.

Преподавателем запланировано использование при чтении лекций и проведении лабораторных занятий технологии учебной дискуссии. Поэтому рекомендуется фиксировать для себя интересные моменты с целью их активного обсуждения на дискуссии в конце лекции и на лабораторных занятиях.

Залогом качественного выполнения лабораторных работ является самостоятельная подготовка к ним путем повторения материалов лекций. Рекомендуется подготовить вопросы по неясным моментам и обсудить их с преподавателем в начале лабораторной работы.

Преподавателем запланировано применение на лабораторных занятиях технологий коллективного взаимодействия.

Преподавателем запланировано применение на лабораторных занятиях технологий развивающейся кооперации, коллективного взаимодействия. Поэтому приветствуется групповой метод выполнения заданий и защиты отчетов.

Для текущего контроля успеваемости по очной форме обучения преподавателем используется балльно-рейтинговая система контроля и оценки академической активности. Поэтому настоятельно рекомендуется тщательно прорабатывать материал дисциплины при самостоятельной работе, участвовать во всех формах обсуждения и взаимодействия, как на лекциях, так и на лабораторных занятиях в целях лучшего освоения материала и получения высокой оценки по результатам освоения дисциплины.

Выполнение самостоятельной работы подразумевает самостоятельное изучение разделов дисциплины, подготовку к лабораторным работам к рубежным контролям, подготовку к зачету.

Рекомендуемая трудоемкость самостоятельной работы представлена в таблице:

### Рекомендуемый режим самостоятельной работы

Наименование вида самостоятельной работы	Рекомендуемая трудоемкость, акад. час.
<b>Самостоятельное изучение тем дисциплины:</b>	<b>34</b>
Подготовка к лабораторным работам (по 2 часа на каждую лабораторную работу)	20
Подготовка к рубежным контролям (по 2 часа на каждый рубеж)	4
Изучение преобразования различных типов данных языка программирования.	1
Изучение конструирования структур данных и их использование для решения практических задач.	4
Изучение методов сортировки и поиска данных и применение их для конкретных структур данных	8
Изучение способов оптимизации сортировки и поиска.	2
Изучение связанных структур данных	8
Изучение древовидных структур данных (деревьев) и методов сортировки и поиска на таких структурах	8
Изучение библиотечных функций для реализации структур данных и алгоритмов стандартных операций	3
<b>Подготовка к зачету</b>	<b>18</b>
<b>Всего:</b>	<b>76</b>

## 6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ

### 6.1. Перечень оценочных средств

1. Балльно-рейтинговая система контроля и оценки академической активности обучающихся в КГУ.
2. Отчеты по лабораторным работам.
3. Банк заданий к рубежным контролям №1, №2.
4. Список вопросов к зачету.

### 6.2. Система балльно-рейтинговой оценки работы обучающихся по дисциплине Очная форма обучения

№	Наименование	Содержание					
		Распределение баллов					
		Вид учебной работы:	Посещение лекций	Выполнение и защита отчетов по лабораторным работам	Рубежный контроль №1	Рубежный контроль №2	Зачет
1	Распределение баллов за семестры по видам учебной работы, сроки сдачи учебной работы (доводятся до сведения студентов на	Балльная оценка:	До 6	До 38	До 12	До 12	До 30

	первом учебном занятии)	Примечания:	по 1 баллу за 2 часовую лекцию	До 2 баллов за 1 часовую (1 л.р.), до 4 баллов за 2 часовую (5 л.р.) и до 8 баллов за 4 часовую (2 л.р.)	На 4-й лабораторной	На 8-й лабораторной	
2	Критерий пересчета баллов в традиционную оценку по итогам работы в семестре и <b>зачета</b>		60 и менее баллов – незачтено; 61 и более баллов - зачтено				
3	Критерии допуска к промежуточной аттестации, возможности получения автоматического зачета (экзаменационной оценки) по дисциплине, возможность получения бонусных баллов		<p>Для допуска к промежуточной аттестации по дисциплине за семестр (зачету) обучающийся должен набрать по итогам текущего и рубежного контролей не менее 51 балла .В случае если обучающийся набрал менее 51 балла, то к аттестационным испытаниям он не допускается</p> <p>Для получения экзамена без проведения процедуры промежуточной аттестации обучающемуся необходимо набрать в ходе текущего и рубежных контролей не менее 61 балла. В этом случае итог балльной оценки, получаемой обучающимся без проведения процедуры промежуточной аттестации, определяется по количеству баллов, набранных им в ходе текущего и рубежных контролей. При этом, на усмотрение преподавателя, балльная оценка обучающегося может быть повышена за счет получения дополнительных баллов за академическую активность.</p> <p>Обучающийся, имеющий право на получение оценки без проведения процедуры промежуточной аттестации, может повысить ее путем сдачи аттестационного испытания. В случае получения обучающимся на аттестационном испытании 0 баллов итог балльной оценки по дисциплине не снижается.</p> <p>За академическую активность в ходе освоения дисциплины , участие в учебной, научно-исследовательской, спортивной, культурно-творческой и общественной деятельности обучающемуся могут быть начислены дополнительные баллы. Максимальное количество дополнительных баллов за академическую активность по одной дисциплине составляет 30.</p> <p>Основанием для получения дополнительных баллов являются:</p> <ul style="list-style-type: none"> <li>- выполнение дополнительных заданий по дисциплине дополнительные баллы начисляются преподавателем;</li> <li>- участие в течение семестра в учебной, научно-исследовательской, спортивной, культурно-творческой и общественной деятельности КГУ.</li> </ul>				

4	Формы и виды учебной работы для неуспевающих (восстановившихся на курсе обучения) обучающихся для получения недостающих баллов в конце семестра	В случае если к промежуточной аттестации (зачету) набрана сумма менее 51 балла, обучающемуся необходимо набрать недостающее количество баллов за счет выполнения дополнительных заданий, до конца последней (зачетной) недели семестра. Ликвидация академических задолженностей, возникших из-за разности в учебных планах при переводе или восстановлении, проводится путем выполнения дополнительных заданий, форма и объем которых определяется преподавателем
---	---	--

### **6.3. Процедура оценивания результатов освоения дисциплины**

Рубежные контроли и зачет проводятся в форме письменного тестирования или устного опроса.

Перед проведением каждого рубежного контроля преподаватель прорабатывает основной материал соответствующих разделов дисциплины в форме краткой лекции-дискуссии. Варианты заданий для рубежных контролей №1,2 состоят из задачи.

На каждое задание при рубежном контроле обучающемуся отводится время не менее 30 минут. По результатам рубежного контроля обучающийся получает до 10 баллов.

Возможно использование теста для неуспевающих студентов.

Преподаватель оценивает в баллах результаты тестирования каждого обучающегося по количеству правильных ответов и заносит в ведомость учета текущей успеваемости.

Зачет проводится в традиционной форме. Билет к зачету состоит из 2 вопросов. Количество баллов за каждый правильный ответ до 15 баллов. Время, отводимое на зачет, составляет 1 астрономический час.

Результаты текущего контроля успеваемости и зачета заносятся преподавателем в зачетную ведомость, которая сдается в организационный отдел института в день зачета, а также выставляются в зачетную книжку обучающегося.

### **6.4. Примеры оценочных средств для рубежных контролей и зачета**

#### **Пример задания для рубежного контроля 1**

Написать программу выполняющую следующие действия:

Сортировку информации, хранящейся в файле в виде записей заданной структуры. Сортировку производить по реквизитам различных типов, использовать один из известных алгоритмов сортировок, например, алгоритм Шелла, Пузырька, Пирамидальная сортировка и т.д...

#### **Пример задания для рубежного контроля 2**

Написать программу выполняющую следующие действия:

Написать программу, реализующую работу со списком студенческой группы. Необходимые действия: вывод на экран, сохранение в файл,

добавление и удаление студента из группы. Информацию о студенте хранить в отдельном классе.

### ***Тест для неуспевающих студентов:***

1. Алгоритм — это ...
  - а) линейная последовательность действий исполнителя, записанная с использованием его системы команд, и воспринимаемая посредством устройства чтения информации.*
  - б) конечная совокупность точно заданных правил решения некоторого класса задач или понятное и точное предписание исполнителю совершить последовательность действий, направленных на достижение указанной цели или на решение поставленной задачи.*
  - в) путь решения какой-либо задачи, записанный в виде блок-схемы.*
  - г) набор правил, устанавливающих какие комбинации символов являются осмысленными предложениями на этом языке.*
2. Какое это свойство алгоритма? «Алгоритм должен содержать предписания о выполнении только таких действий, которые входят в систему команд исполнителя».
  - а) Понятность*
  - б) Дискретность*
  - в) Определенность*
  - г) Результативность*
  - д) Массовость*
3. Выберите пункты, относящиеся к отличительным особенностям понятия «тип данных»
  - а) тип определяет класс значений, которые могут принимать переменная или выражение*
  - б) каждое значение принадлежит одному и только одному типу*
  - в) тип значения константы, переменной или выражения можно вывести либо из контекста, либо из самого операнда, не обращаясь к значениям, вычисляемым во время работы программы.*
  - г) в общем случае, один тип данных может объединять объекты, определенные в разных файлах и имеющие разную структуру.*
  - д) типы данных увеличивают неопределенность в выполнении программ, но применяется для упрощения разработки.*
4. Укажите в какой строке записаны константы типа double в C++
  - а) «Привет, мир!», «двенадцать», «88.5»*
  - б) 'x','t','4'*
  - в) 5, 8, 10*
  - г) 8.0f, 44.4f, 3.14f*
  - д) 9.0, 2.7, 53.4*

5. Описание какой парадигмы программирования дано: «Методология этой парадигмы основана на принципе конструирования программ как описаний последовательного изменения состояния вычислителя пошаговым образом. Для этой парадигмы характерна полная определенность и контролируемость переходов из одного состояния в другое»

- а) *Объектно-ориентированное*
- б) *Императивное*
- в) *Процедурное*
- г) *Функциональное*
- д) *Нейро-сетевое*

6. Укажите верное обращение к элементам объекта следующей структуры

```
struct Color
{
    char R;
    char G;
    char B;
};
Color Red;
```

- а) *Red::R = 255;*
- б) *Red->R = 255;*
- в) *Red[R] = 255;*
- г) *Red.R = 255;*

7. Чем структуры в C++ отличаются от классов?

- а) *Классы могут содержать описания функций, а структуры не могут.*
- б) *Структуры предоставляют возможность создавать собственные типы данных, а классы нет.*
- в) *Отличаются модификаторами доступа, члены классов по умолчанию считаются private, а структур – public.*
- г) *Для структур невозможно использовать модификаторы доступа private, public, protected, а для классов возможно.*

8. Укажите где верно описан массив классов

```
Class Car
{
    double tnk_vol, speed;
    Engine eng;
    Car Car (double t_v, Engine e);
    {
        tnk_vol=t, eng=e;
    }
};
```

- а) *Car Cars(15);*
- б) *Car Cars[15] = new Car();*
- в) *Car Cars[15];*
- г) *Car = new Car(15);*

9. Для сравнения сложности алгоритмов используют класс (набор) функций  $O(f)$ . Выберите верное определение этого класса

- а) *Функции, описывающие количество элементов массива, которые могут быть обработаны данным алгоритмом за 1 секунду. Для каждого типа данных своя функция.*
- б) *Функции, определяющие конечное число шагов, которые необходимо сделать для обработки алгоритмом 1 элемента.*
- в) *Функции, растущие не быстрее  $f$ .  $f$  – является верхней границей для класса.*
- г) *Функции, растущие не медленнее  $f$ .  $f$  – является нижней границей для класса.*

10. Выберите где классы функций роста сложности алгоритмов расположены от самой медленной до самой быстрой

- а)  $\log_2(n)$ ,  $n$ ,  $n \cdot \log_2(n)$ ,  $n^2$ ,  $n^3$ ,  $2^n$
- б)  $2^n$ ,  $n^3$ ,  $n^2$ ,  $n \cdot \log_2(n)$ ,  $n$ ,  $\log_2(n)$
- в)  $n \cdot \log_2(n)$ ,  $2^n$ ,  $n^3$ ,  $n^2$ ,  $n$ ,  $\log_2(n)$
- г)  $\log_2(n)$ ,  $n \cdot \log_2(n)$ ,  $n^3$ ,  $n^2$ ,  $n$ ,  $2^n$

11. Код функции какой сортировки приведен:

```
void SortInt1(int*array, int n)
{
    for(int i=1; i<n; i++)
        for(int j=n-1; j>=i; j--)
            if(array[j]<array[j-1])
            {
                int copy = array[j];
                array[j] = array[j-1];
                array[j-1] = copy;
            }
}
```

- а) *Быстрая*
- б) *Простыми подстановками*
- в) *Простыми обменами (пузырьковая)*
- г) *Пирамидальная*
- д) *Шелла*

12. Укажите где приведен корректный прототип шаблонной функции в C++?

- а) *template <class T> void BetterBubbleSortPlus(T\*a, int n);*
- б) *void BetterBubbleSortPlus( template T\*a, int n);*

в) `<class T> void BetterBubbleSortPlus(int*a, int n);`  
 г) `void template <class T>BetterBubbleSortPlus(T*a, int n);`

13. Описание элемента какой структуры данных приведено?

```
template <class ITEM>
struct TItem
{
    ITEM data;
    TItem<ITEM> *next;
    TItem(const ITEM& data, TItem<ITEM> *next = 0)
    {
        this->data = data;
        this->next = next;
    }
};
```

- а) Массив
- б) Линейный двунаправленный список
- в) Линейный однонаправленный список
- г) Бинарное дерево
- д) Граф

14. Узел дерева, не имеющий родителя, называется:

- а) Листом
- б) Сиротой
- в) Корневым узлом
- г) главным узлом

15. Для чего предназначена приведенная ниже функция

```
void Func( int size, const char*filename)
{
    fInput = fopen( filename, "rt");
    if(fInput==NULL)
    {
        printf("Error while opening key file\n");
        exit( 1 );
    }
    InitSearchTree( size );
    PrepareTree( 1 );
    if( fclose( fInput ) == EOF )
    {
        printf( "Error while closing key file\n" );
        exit( 2 );
    }
}
```



*а) Подготовка таблицы поиска на основе информации, загружаемой из файла*

*б) Создание дерева поиска заданного размера*

*в) Первоначальная подготовка массива ключей для бинарного поиска*

*г) Чтение текстового файла и вывод его содержимого на экран*

### **Примерный список вопросов к зачету:**

1. Алгоритмы, типы алгоритмов. Основные парадигмы программирования
2. Понятие типа данных. Свойства типа данных. Базовые типы данных в языке программирования C++.
3. Структуры и классы. Их конструирование. Способы доступа к элементам структуры (класса).
4. Конструирование своих типов данных на основе структур и массивов, конструирование массива структур и структур, содержащих массивы.
5. Преобразование типов данных.
6. Сортировка. Понятие внутренней сортировки.
7. Назовите и кратко опишите известные Вам алгоритмы сортировки.
8. Примеры использования алгоритма сортировки для произвольной структуры данных.
9. Назначение алгоритмов поиска данных. Применение алгоритмов поиска, условия применения.
10. Приведите примеры кода алгоритмов поиска данных.
11. Связные структуры данных. Понятие. Основные определения.
12. Линейный однонаправленный список. Реализация базового класса (без кода функций).
13. Линейный однонаправленный список. Базовый действия. Вставка элементов, удаление элементов, поиск элемента, удаление списка.
14. Линейный однонаправленный список. Понятие об итераторах. Реализация класса внешнего итератора. Базовые действия.
15. Линейный однонаправленный список. Функции вставки, удаления и поиска элемента списка с использованием внешнего итератора.
16. Деревья. Основные понятия и определения.
17. Деревья. Упаковка деревьев в массив. Реализация алгоритма пирамидальной сортировки.
18. Напишите код программы «Сортировка пузырьком» для заданной структуры данных хранящейся в файле.
19. Напишите код программы «Сортировка выбором» для заданной структуры данных хранящейся в файле.
20. Напишите код программы «Сортировка Шелла» для заданной структуры данных хранящейся в файле.
21. Напишите код программы «Пирамидальная сортировка» для заданной структуры данных хранящейся в файле.

22. Напишите код программы «Быстрая сортировка» для заданной структуры данных хранящейся в файле.

### **6.5. Фонд оценочных средств**

Полный банк заданий для текущего, рубежных контролей и промежуточной аттестации по дисциплине, показатели, критерии, шкалы оценивания компетенций, методические материалы, определяющие процедуры оценивания образовательных результатов, приведены в учебно-методическом комплексе дисциплины.

## **7. ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ УЧЕБНАЯ ЛИТЕРАТУРА**

### **7.1. Основная учебная литература**

1. Кубенский А.А. Структуры и алгоритмы обработки данных: объектно-ориентированный подход на C++. - СПб.: Питер, 2004. - 464 с.
2. Камаев В.А. Технологии программирования: Учебник/В.А. Камаев, В.В. Костерин. — 2-е изд., перераб. и доп. — М.: Высш. шк., 2006. - 454 с: ил.
3. Жоголев Е.А. Технология программирования/Е.А. Жоголев. — М.: Научный мир, 2004. — 216 с.

### **7.2. Дополнительная учебная литература**

1. Пол А. Объектно-ориентированное программирование на C++. М.: Финансы и статистика, 2005. - 477 с.
2. Буч Г. Объектно-ориентированное проектирование с примерами применения, М.: Финансы и статистика, 2002 515 с.
3. Дж.Элджер, C++: библиотека программиста, СПб: Питер, 2000. – 320 с.
4. Секунов Н. Самоучитель visual C++ 6, СПб: Питер, 2002. – 920 с.

## **8. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ**

1. Камкин И.П. Методические указания к контрольной работе по дисциплине «Структуры и алгоритмы обработки данных» для студентов заочной формы обучения – Курган, 2019.
2. Камкин И.П. Методические указания к комплексу практических работ по дисциплине «Структуры и алгоритмы обработки данных» для студентов заочной формы обучения – Курган, 2019.
2. Камкин И.П. Методические указания к комплексу лабораторных работ по дисциплине «Структуры и алгоритмы обработки данных» для студентов очной и заочной формы обучения – Курган, 2019.

## **9. РЕСУРСЫ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫЕ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ**

1. [dist.kgsu.ru](http://dist.kgsu.ru) - Система поддержки учебного процесса КГУ;
2. <http://www.cetiforum.ru> - Электронные версии учебников, самоучителей по программированию на алгоритмических языках

3. <http://www.docs.microsoft.com/ru-ru/> - Документация по Microsoft Visual Studio и многое другое.

4. <https://code-live.ru/tag/cpp-manual/> - Простые задачи на C++ для начинающих.

5.  
[https://www.youtube.com/watch?v=Sy\\_wba7l1UU&list=PLawfWYMUziZqyUL5QDLVbe3j5BKWj42E5](https://www.youtube.com/watch?v=Sy_wba7l1UU&list=PLawfWYMUziZqyUL5QDLVbe3j5BKWj42E5) – Курс CS50 по программированию от Гарварда. Видео лекций на русском языке.

## **10. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ, ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ И ИНФОРМАЦИОННЫЕ СПРАВОЧНЫЕ СИСТЕМЫ**

1. ЭБС «Лань»
2. ЭБС «Консультант студента»
3. ЭБС «Znaniium.com»
4. «Гарант» - справочно-правовая система

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

Материально-техническое обеспечение по реализации дисциплины осуществляется в соответствии с требованиями ФГОС ВО по данной образовательной программе.

## **12. Для студентов, обучающихся с использованием дистанционных образовательных технологий**

При использовании электронного обучения и дистанционных образовательных технологий (далее ЭО и ДОТ) занятия полностью или частично проводятся в режиме онлайн. Объем дисциплины и распределение нагрузки по видам работ соответствует п. 4.1. Распределение баллов соответствует п. 6.2, либо может быть изменено в соответствии с решением кафедры, в случае перехода на ЭО и ДОТ в процессе обучения. Решение кафедры об используемых технологиях и системе оценивания достижений обучающихся принимается с учетом мнения ведущего преподавателя и доводится до сведения обучающихся.

Аннотация к рабочей программе дисциплины  
**«СТРУКТУРЫ И АЛГОРИТМЫ ОБРАБОТКИ ДАННЫХ»**

образовательной программы высшего образования –  
программы бакалавриата

**27.03.04 – Управление в технических системах**

Направленность:

**Автоматика и робототехнические системы**

Трудоемкость дисциплины: 3 ЗЕ (108 академических часов)

Семестр: 5 (очная форма обучения)

Форма промежуточной аттестации: Зачет

Содержание дисциплины

Базовые данные языка программирования, преобразование данных. Методы построение сложных структур данных и способы их использования в практическом программировании. Сортировки: «сортировка пузырьком», «сортировка выбором», «сортировка вставками», «сортировка Шелла», «Пирамидальная сортировка», «Быстрая сортировка». Поиск данных, в упорядоченном и не упорядоченном массиве.