


ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО РЫБОЛОВСТВУ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«КАМЧАТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «КамчатГТУ»)

Факультет информационных технологий
Кафедра «Системы управления»

УТВЕРЖДАЮ
Декан ФИТ

 /И.А. Рычка/

«17» 03 2021 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Структура и алгоритмы обработки данных»

направление подготовки
09.03.01 «Информатика и вычислительная техника»
(уровень бакалавриата)

направленность (профиль)
«Программное обеспечение средств вычислительной техники и автоматизи-
рованных систем»

Петропавловск-Камчатский
2021

Рабочая программа разработана в соответствии с ФГОС ВО по направлению подготовки 09.03.01 «Информатика и вычислительная техника», профиль «Программное обеспечение средств вычислительной техники и автоматизированных систем», и учебного плана ФГБОУ ВО «КамчатГТУ».

Составитель рабочей программы:
доцент кафедры СУ, к.т.н. _____



Луковенкова О.О.

Рабочая программа рассмотрена на заседании кафедры «Системы управления»
Протокол № 8 от «19» 02 2021 года.

«19» 02 2021 г.



Заведующий кафедрой
«Системы управления»
А.А. Марченко

1. Цели и задачи учебной дисциплины

Учебная дисциплина «Структуры и алгоритмы обработки данных» относится к вариативной части Блока 1 «Дисциплины (Модули)» основной профессиональной образовательной программы по направлению подготовки 09.03.01 «Информатика и вычислительная техника», профиль «Программное обеспечение средств вычислительной техники и автоматизированных систем», предусмотренной учебным планом ФГОУ ВПО «КамчатГТУ».

Цель изучения дисциплины:

научить студентов в процессе проектирования программ квалифицированно выбирать рациональные структуры данных и языковые конструкции, обеспечивающие построение эффективных алгоритмов и программ применительно к задачам со сложной организацией данных.

Задачи изучения дисциплины:

- формирование базовых теоретических понятий, лежащих в основе процесса разработки алгоритмов и структур данных;
- формирование представления и знания об основных классах алгоритмов (поиска, сортировки), используемых в них структурах данных и общих схемах решения задач на их основе;
- обучение реализации типовых алгоритмов и структур данных и их модификаций на выбранном языке программирования;
- формирование представления и знаний об анализе сложности алгоритмов и программ.

Компетенции обучающегося, формируемые в результате освоения дисциплины

Дисциплина «Структура и алгоритмы обработки данных» направлена на освоение следующих компетенций основной профессиональной образовательной программы по направлению 09.03.01 «Информатика и вычислительная техника» федерального государственного образовательного стандарта высшего образования:

- способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью (ОПК-4);
- способен разрабатывать алгоритмы и программы, пригодные для практического применения (ОПК-8).

Планируемые результаты обучения при изучении дисциплины, соотнесенные с планируемыми результатами освоения образовательной программы, представлены в таблице.

Код компетенции	Планируемые результаты освоения образовательной программы	Код и наименование индикатора	Планируемый результат обучения по дисциплине	Код показателя освоения
ОПК-4	способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью	ИД-1 _{опк-4} Знает основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы ИД-2 _{опк-4} Умеет применять стандарты оформ-	Знать: – основные принципы описания алгоритмов; – основные принципы описания структур данных. Уметь:	З(ОПК-4)1 З(ОПК-4)2

Код компетенции	Планируемые результаты освоения образовательной программы	Код и наименование индикатора	Планируемый результат обучения по дисциплине	Код показателя освоения
		<p>ления технической документации на различных стадиях жизненного цикла информационной системы</p> <p>ИД-3_{опк-4} Имеет навыки составления технической документации на различных этапах жизненного цикла информационной системы</p>	<ul style="list-style-type: none"> – разрабатывать техническое описание для программ; – описывать используемые алгоритмы и структуры данных; – производить оценку сложности разработанного алгоритма и программы в целом. <p>Владеть:</p> <ul style="list-style-type: none"> – методологией описания предметной области программы; – методами сравнения вычислительной сложности программ. 	<p>У(ОПК-4)1</p> <p>У(ОПК-4)2</p> <p>У(ОПК-4)3</p> <p>В(ОПК-4)1</p> <p>В(ОПК-4)2</p>
ОПК-8	способен разрабатывать алгоритмы и программы, пригодные для практического применения	<p>ИД-1_{опк-8} Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий</p> <p>ИД-2_{опк-8} Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ</p> <p>ИД-3_{опк-8} Имеет навыки программирования, отладки и тестирования</p>	<p>Знать:</p> <ul style="list-style-type: none"> – языковые средства описания различных структур данных. <p>Уметь:</p> <ul style="list-style-type: none"> – разрабатывать эффективные алгоритмы обработки данных и программировать их на известных языках программирования. <p>Владеть:</p> <ul style="list-style-type: none"> – навыками разработки алгоритмов и структур данных, пригодных для практического использования. 	<p>З(ОПК-8)1</p> <p>У(ОПК-8)1</p> <p>В(ОПК-8)1</p>

Код компетенции	Планируемые результаты освоения образовательной программы	Код и наименование индикатора	Планируемый результат обучения по дисциплине	Код показателя освоения
		прототипов программно-технических комплексов задач		

2. Место дисциплины в структуре образовательной программы

Связь с предшествующими дисциплинами

Данная дисциплина изучается в 6 семестре. Изучение данной дисциплины опирается на такие курсы как «Математика», «Математическая логика и теория алгоритмов», «Программирование на языке высокого уровня», «Информатика» и т.д.

Связь с последующими дисциплинами

Знания, умения и навыки, приобретенные студентами при изучении дисциплины «Структура и алгоритмы обработки данных», позволят им в дальнейшем успешно изучать дисциплины «Технология разработки программного обеспечения», «Объектно-ориентированное программирование», «Комбинаторные алгоритмы», а также успешно пройти преддипломную практику и выполнить выпускную квалификационную работу.

3. Содержание дисциплины

3.1. Тематический план дисциплины

Наименование разделов и тем	Всего часов	Аудиторные занятия	Контактная работа по видам учебных занятий			Самостоятельная работа	Формы текущего контроля	Итоговый контроль знаний по дисциплине
			ЛК	ПЗ	ЛЗ			
Тема 1: Анализ вычислительной сложности алгоритмов	26	16	8	-	8	10	Опрос, РЗ	
Тема 2: Алгоритмы обработки данных	26	16	8	-	8	10	Опрос, РЗ	
Тема 3: Линейные структуры данных	28	18	9	-	9	10	Опрос, РЗ	
Тема 4: Нелинейные структуры данных	28	18	9	-	9	10	Опрос, РЗ	
Экзамен, КР	-	-	-	-	-	-	Опрос	
Всего	144	68	34	-	34	40		36

*ПЗ – практическое задание, РЗ – решение задач, КС – конкретная ситуация

3.2. Описание содержания дисциплины

Тема 1. Анализ вычислительной сложности алгоритмов

Лекция 1.1. *Анализ вычислительной сложности алгоритмов.* Основы анализа. Оценка размера входных данных. Порядок роста. Эффективность алгоритма в разных случаях. Асимптотический анализ. Асимптотические отношения. Асимптотический анализ и пределы. Основные классы вычислительной сложности. Рекурсивные алгоритмы. Вычислительная сложность рекурсивных алгоритмов. Асимптотический анализ рекурсивных алгоритмов.

Лабораторная работа 1.1. Анализ вычислительной сложности алгоритмов.

Лабораторная работа 1.2. Анализ вычислительной сложности рекурсивных алгоритмов.

Тема 2. Алгоритмы обработки данных

Лекция 1.2. *Сортировки.* Общая задача сортировки. Классификация алгоритмов сортировки. Алгоритмы сортировки с квадратичной вычислительной сложностью. Алгоритмы сортировки с линейно-логарифмической вычислительной сложностью. Алгоритмы сортировки, не основанные на сравнениях. Анализ вычислительной сложности алгоритмов сортировок в различных случаях.

Лабораторная работа 1.3. Реализация различных алгоритмов сортировок.

Лекция 1.3. *Поиск.* Общая задача поиска. Последовательный поиск. Бинарный поиск. Поиск подстроки в строке.

Лабораторная работа 1.4. Реализация различных алгоритмов поиска.

Тема 3. Линейные структуры данных

Лекция 2.1. *Списки.* Понятие линейной структуры данных. Односвязные, двусвязные, кольцевые списки. Реализация списков на базе массива и с помощью указателей.

Лабораторная работа 2.1. Реализация односвязных и двусвязных списков.

Лекция 2.2. *Стеки. Очереди. Деки.* Понятие стека. Способы реализации стека. Алгоритмы, использующие стек. Префиксная и постфиксная нотации арифметических выражений. Понятие очереди. Алгоритмы, использующие очередь. Способы реализации очереди. Модель электронной очереди банка. Понятие дека. Способы реализации дека. Алгоритмы, использующие дек. Построение выпуклого многоугольника по множеству точек.

Лабораторная работа 2.2. Реализация стека, дека и очереди.

Тема 4. Нелинейные структуры данных

Лекция 2.3. *Деревья.* Понятие нелинейной структуры данных. Дерево как частный случай графа. Бинарные деревья. Обходы бинарных деревьев. Бинарные деревья поиска. Кучи. Структуры данных множество и очередь с приоритетом. AVL-деревья.

Лабораторная работа 2.3. Реализация бинарного дерева поиска.

Лабораторная работа 2.4. Реализация бинарной кучи.

Лекция 2.4. Графы. Понятие графа. Виды графов. Способы представления графов. Матрица смежности. Списки смежности. Виды обходов графа. Задачи на графах. Поиск минимального остовного дерева. Поиск кратчайших путей на графе.

Лабораторная работа 2.5. Задачи на графах.

Лекция 2.5. Хеш-таблицы. Структуры хранения данных. Понятие хеш-таблицы. Способы разрешения коллизий. Открытая адресация. Метод цепочек. Хеш-функции: числовые и строковые. Алгоритмы кодирования данных.

Лабораторная работа 2.6. Организация хеш-таблиц.

СРС. Проработка теоретического материала по основной и дополнительной литературе. Подготовка отчетов к лабораторным работам. Подготовка к экзамену.

3.3. Учебно-методическое обеспечение для самостоятельной работы обучающихся

В целом внеаудиторная самостоятельная работа обучающегося при изучении курса включает в себя следующие виды работ:

- проработка (изучение) материалов лекций;
- чтение и проработка рекомендованной основной и дополнительной литературы;
- подготовка к лабораторным работам;
- поиск и проработка материалов из Интернет-ресурсов, периодической печати;
- выполнение домашних заданий в форме творческих (проблемно-поисковых, групповых) заданий, докладов;
- подготовка к текущему и итоговому (промежуточная аттестация) контролю знаний по дисциплине.

Основная доля самостоятельной работы обучающихся приходится на подготовку к лабораторным работам, тематика которых полностью охватывает содержание курса. Самостоятельная работа по подготовке к лабораторным работам предполагает умение работать с первичной информацией.

4. Фонд оценочных средств

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине «Структуры и алгоритмы обработки данных» представлен в приложении к рабочей программе дисциплины и включает в себя:

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы;
2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания;
3. Типовые контрольные задания или материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций;
4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Примерный перечень вопросов к промежуточной аттестации

1. Оценка вычислительной сложности алгоритмов. Основные классы вычислительной сложности.
2. Оценка вычислительной сложности алгоритмов. Скорость роста функций. Сравнение скорости роста функций.
3. Оценка вычислительной сложности рекурсивных алгоритмов. Основная теорема.
4. Общая задача сортировки. Виды алгоритмов сортировки. Сортировка «пузырьком», быстрая сортировка, поразрядная сортировка для чисел
5. Общая задача сортировки. Виды алгоритмов сортировки. Сортировка выбором, сортировка слиянием, сортировка подсчетом
6. Общая задача сортировки. Виды алгоритмов сортировки. Сортировка вставками, пирамидальная сортировка, поразрядная сортировка для строк
7. Общая задача поиска. Виды алгоритмов поиска. Линейный поиск. Бинарный поиск.
8. Линейные структуры данных. Односвязные списки
9. Линейные структуры данных. Двусвязные списки
10. Линейные структуры данных. Стек. Способы реализации стека
11. Линейные структуры данных. Стек. Построение обратной польской нотации
12. Линейные структуры данных. Очередь. Способы реализации очереди
13. Линейные структуры данных. Очередь. Обход графа в ширину
14. Линейные структуры данных. Дек. Способы реализации дека
15. Линейные структуры данных. Дек. Построение выпуклого многоугольника по множеству точек
16. Нелинейные структуры данных. Графы. Граф как модель. Способы реализации графа. Насыщенность. Обходы графов.
17. Нелинейные структуры данных. Графы. Задачи на графах (поиск кратчайших путей, построение минимального остовного дерева)
18. Нелинейные структуры данных. Деревья. Виды деревьев. Бинарные деревья. Способы реализации произвольных и бинарных деревьев. Обходы бинарных деревьев.
19. Нелинейные структуры данных. Бинарные деревья поиска. Реализация бинарных деревьев поиска.
20. Нелинейные структуры данных. Множества. Операции над множествами. Реализация множества.
21. Нелинейные структуры данных. Очередь с приоритетом. Реализация очереди с приоритетом.
22. Нелинейные структуры данных. Хеш-таблицы. Понятие хеш-функции. Способы разрешения коллизий.
23. Нелинейные структуры данных. Ассоциативные массивы.
24. Библиотека STL. Линейные структуры данных
25. Библиотека STL. Нелинейные структуры данных

5. Рекомендуемая литература

5.1. Основная литература

1. Вирт Н. Алгоритмы и структуры данных. Новая версия для Оберона. 2010 (ЭБС «Лань»).

5.2. Дополнительная литература

1. Бакнелл Д ж. Фундаментальные алгоритмы и структуры данных в Delphi, 2006г. (11)
2. Объектно-ориентированное программирование: учебник/ Г.С. Иванова, Т.Н. Ни-чущкина, Е.К. Пугачев. Иванова Г.С.2003г. (388)

6. Методические указания для обучающихся по освоению дисциплины

Подготовка к лекционным занятиям

Лекции составляют основу теоретического обучения и дают систематизированные основы научных знаний по дисциплине, концентрируют внимание обучающихся на наиболее сложных и узловых вопросах, стимулируют их активную познавательную деятельность и способствуют формированию творческого мышления.

При подготовке к лекции следует предварительно ознакомиться с учебным материалом по теме занятия и при конспектировании лекции акцентировать внимание на новых теоретических положениях и иных данных, не нашедших отражения в учебной литературе.

Для успешного изучения дисциплины студенту рекомендуется систематически готовиться к каждому занятию по следующей схеме:

- повторить материал предыдущей лекции, используя конспекты, учебную и специальную литературу
- ответить на контрольные вопросы по изучаемой теме.

Подготовка к лабораторным работам

Лабораторная работа – это выполнение студентами под руководством преподавателя или по инструкции заданий (решение задач, написание программ) с применением персонального компьютера.

В ходе лабораторных работ студенты воспринимают и осмысливают новый учебный материал. Лабораторные занятия носят систематический характер, регулярно следуя за лекционными занятиями. Лабораторные работы выполняются согласно графику, при этом соблюдается принцип индивидуального выполнения работ.

Обучающийся должен подготовить отчет к каждой лабораторной работе, предусмотренной планом.

При подготовке к лабораторным занятиям необходимо заранее изучить методические рекомендации по его проведению. Обратит внимание на цель занятия, на основные вопросы для подготовки к занятию, на содержание темы занятия.

7. Курсовой проект (работа)

В соответствии с учебным планом предусмотрено курсовое проектирование по дисциплине «Структуры и алгоритмы обработки данных».

Примерная тематика курсовых работ

1. Красно-черные деревья
2. 2-3 деревья
3. AVL деревья
4. Деревья турниров
5. Алгоритм Косарайо для поиска компонент сильной связности в графе
6. Алгоритм Тарьяна для поиска компонент сильной связности в графе
7. Алгоритмы ранжирования. PageRank
8. Алгоритм кластеризации k-средних (k-means)
9. Древовидные таблицы
10. Хеширование строк

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

В учебном процессе используются следующие информационные технологии:

- презентации;

- облачные хранилища для обмена информацией;
- электронные учебники;
- информационные ресурсы сети Интернет.

Программное обеспечение

Для формирования отчетов по лабораторным работам

1. MS Word
2. Latex

Для просмотра электронных документов

1. MS Word
2. MS PowerPoint
3. STDViewer

Для разработки на языке программирования C/C++

1. Visual Studio Community (<https://visualstudio.microsoft.com>)
2. Dev C++ (<http://orwelldevcpp.blogspot.com/>)
3. CodeLite (<https://codelite.org/>)
4. Code::Blocks (<http://www.codeblocks.org/>)

Для разработки на Pascal

1. ABC Pascal (<http://pascalabc.net/>)

Для разработки на Python

1. Python 3.x (<https://www.python.org/>)
2. PyCharm (<https://www.jetbrains.com/pycharm/download/#section=windows>)
3. Anaconda (<https://www.anaconda.com/distribution/>)
4. Wing IDE (<http://www.wingware.com/>)

9. Материально-техническое обеспечение дисциплины

На кафедре имеются аудитории для проведения лекционных занятий и компьютерные классы для проведения лабораторных занятий.