

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Чувашский государственный университет имени И. Н. Ульянова»

Факультет информатики и вычислительной техники

Кафедра вычислительной техники

«УТВЕРЖДАЮ»
Проректор по учебной работе

И.Е. Певринов

«15» августа 2017 г.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
«СТРУКТУРЫ И АЛГОРИТМЫ ОБРАБОТКИ ДАННЫХ»

Направление подготовки (специальность) **09.03.01 «Информатика и вычислительная техника»**

Квалификация (степень) выпускника Бакалавр

Профиль (направленность) *Программное обеспечение средств вычислительной техники и автоматизированных систем*

Академический бакалавриат

Рабочая программа основана на требованиях Федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.01 Информатика и вычислительная техника, утвержденного приказом Министерства образования и науки 12.01.2016 г. №5.


СОСТАВИТЕЛЬ (СОСТАВИТЕЛИ):

кандидат технических наук, доцент  Л.А. Павлов

ОБСУЖДЕНО:

на заседании кафедры вычислительной техники «30» августа 2017 г., протокол № 1


заведующий кафедрой

 А.В. Щипцова

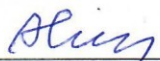
СОГЛАСОВАНО:

Методическая комиссия факультета информатики и вычислительной техники «30» августа 2017 г., протокол № 1

Декан факультета

 А.В. Щипцова

Директор научной библиотеки

 Н.Д. Никитина

Начальник управления информатизации

 И.П. Пивоваров

Начальник учебно-методического управления

 В.И. Маколов

Оглавление

1. Цель и задачи обучения по дисциплине.....	4
2. Место дисциплины в структуре основной образовательной программы (ООП).....	4
3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ООП	5
4. Структура и содержание дисциплины.....	5
5. Содержание разделов дисциплины.....	7
6. Образовательные технологии.....	11
7. Формы аттестации и оценочные материалы.....	12
8. Учебно-методическое и информационное обеспечение дисциплины	15
9. Материально-техническое обеспечение дисциплины	17
10. Средства адаптации преподавания дисциплины к потребностям лиц с ограниченными возможностями	17
11. Методические рекомендации по освоению дисциплины	17

1. Цель и задачи обучения по дисциплине

Цель обучения по дисциплине – формирование комплекса знаний, умений и навыков по разработке и программной реализации алгоритмов и структур данных, теоретического и экспериментального анализа алгоритмов для определения их вычислительной сложности для принятия обоснованных проектных решений при разработке программных комплексов.

Дисциплина обеспечивает совершенствование навыков, полученных при изучении основ программирования. Акцент смещается на стадию разработки алгоритмов, под которую подводится теоретическая база. В основу закладывается понятие эффективности алгоритма. Дисциплина находится на стыке программирования и математики с большей ориентацией на практическое использование в программировании. Дисциплина формирует базовые знания, умения и навыки для решения профессиональных задач, связанных с разработкой программного обеспечения.

Задачами преподавания дисциплины являются:

- сформировать базовые теоретические понятия, лежащие в основе процесса разработки алгоритмов и структур данных;
- развить навыки применения основных структур данных и типовых алгоритмов их создания и обработки; определения теоретической и экспериментальной оценок вычислительной сложности алгоритмов; выбора структур данных при проектировании алгоритмов с целью повышения их эффективности;
- выработать представление о возможностях конкретной системы программирования в плане реализации различных структур данных и об эффекте, достигаемом при применении структур и алгоритмов в программировании;
- сформировать умение правильно выбирать структуры данных при проектировании алгоритмов с целью повышения эффективности алгоритмов, реализовать их в конкретной системе программирования;
- обеспечить получение практического опыта определения теоретической и экспериментальной оценок вычислительной сложности алгоритмов, уяснить связь сложности алгоритма со свойствами структур данных.

2. Место дисциплины в структуре основной образовательной программы (ООП)

Дисциплина реализуется в рамках обязательных дисциплин вариативной части образовательной программы бакалавра.

Дисциплина опирается на знания, умения и навыки, которые студенты должны получить при изучении дисциплин: «Алгебра и геометрия», «Математический анализ», «Дискретная математика», «Математическая логика и теория алгоритмов», «Вычислительная математика», «Теория вероятностей, математическая статистика и случайные процессы», «Информатика», «Программирование».

Дисциплина направлена на изучение основных структур и алгоритмов обработки данных, наиболее часто используемых при разработке программных продуктов. В рамках дисциплины студенты получают знания об абстрактных типах данных; линейных и нелинейных структурах данных; алгоритмах исчерпывающего поиска, методах быстрого поиска, сортировки, основных алгоритмах на графах. Рассматриваются вопросы аналитической и экспериментальной оценки вычислительной сложности алгоритмов. В процессе выполнения лабораторных и расчетно-графических работ студенты получают практические навыки по разработке алгоритмов и структур данных, выбора эффективных алгоритмов и структур данных для решения поставленных задач.

Дисциплина дает теоретическую основу для успешного освоения последующих дисциплин: «Базы данных», «Web-программирование», «Объектно-ориентированное программирование», «Функциональное и логическое программирование», «Теория язы-

ков программирования и методы трансляции», «Технология разработки программного обеспечения» и других дисциплин и практик, связанных с изучением и практическим применением различных языков и технологий программирования.

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ООП

Процесс обучения по дисциплине направлен на формирование следующих профессиональных компетенций:

способность разрабатывать компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии программирования (ПК-2);

способность обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности (ПК-3).

В результате обучения по дисциплине, обучающийся должен (ЗУН):

знать:

компетенция ПК-2:

– основные структуры данных, используемые для представления типовых информационных объектов (З1),

– основные алгоритмы и структуры данных, используемые для решения типовых задач, часто встречающихся при разработке программных комплексов (З2),

компетенция ПК-3:

– основные задачи и методы анализа алгоритмов (З3),

– вычислительную сложность основных алгоритмов для решения типовых задач, часто встречающихся при разработке программных комплексов (З4);

уметь:

компетенция ПК-2:

– разрабатывать алгоритмы решения задач, возникающих при проектировании программных комплексов, выбирая (разрабатывая) подходящие структуры данных для представления информационных объектов (У1),

– реализовывать алгоритмы и используемые структуры данных средствами языков программирования (У2),

компетенция ПК-3:

– применять математический аппарат для анализа вычислительной сложности алгоритмов (У3),

– экспериментально исследовать вычислительную сложность алгоритмов и программ (У4);

владеть навыками:

компетенция ПК-2:

– разработки структур, алгоритмов и программ обработки данных (Н1),

компетенция ПК-3:

– применения теоретических и экспериментальных методов анализа алгоритмов для оценки их вычислительной сложности (Н2).

4. Структура и содержание дисциплины

Образовательная деятельность по дисциплине проводится:

– в форме контактной работы обучающихся с педагогическими работниками организации и (или) лицами, привлекаемыми организацией к реализации образовательных программ на иных условиях (далее – контактная работа);

– в форме самостоятельной работы.

Контактная работа включает в себя занятия лекционного типа, занятия семинарского типа (лабораторные работы), групповые и (или) индивидуальные консультации, в том числе в электронной информационно-образовательной среде.

Обозначения:

Л – лекции, л/р – лабораторные работы, п/р – практические занятия, КСР – контроль самостоятельной работы, СРС – самостоятельная работа студента, ИФР – интерактивная форма работы, К – контроль.

4.1. Содержание дисциплины

Содержание	Формируемые компетенции	Формируемые ЗУН
Раздел 1. Алгоритмы и их сложности	ПК-3	ЗЗ, УЗ, У4, Н2
1.1. Введение		
1.2. Асимптотические обозначения		
1.3. Рекуррентные соотношения		
Раздел 2. Структуры данных	ПК-2, ПК-3	З1, З2, З4, У1, У2, У3, Н1, Н2
2.1. Абстрактные типы данных. Представление множеств		
2.2. Стеки и очереди		
2.3. Деревья		
2.4. Прохождения деревьев		
2.5. Расширенные бинарные деревья		
Раздел 3. Исчерпывающий поиск	ПК-2, ПК-3	З1, З2, З4, У1, У2, У3, У4, Н1, Н2
3.1. Поиск с возвратом		
3.2. Метод ветвей и границ		
3.3. Методы решета		
3.4. Генерация элементарных комбинаторных объектов		
Раздел 4. Методы поиска	ПК-2, ПК-3	З1, З2, З4, У1, У2, У3, У4, Н1, Н2
4.1. Последовательный поиск. Логарифмический поиск в статических таблицах		
4.2. Логарифмический поиск в динамических таблицах		
4.3. Поиск с хешированием		
4.4. Внешний поиск		
Раздел 5. Сортировка	ПК-2, ПК-3	З1, З2, З4, У1, У2, У3, У4, Н1, Н2
5.1. Методы сортировки		
5.2. Внутренняя сортировка		
5.3. Внешняя сортировка		
Раздел 6. Алгоритмы на графах	ПК-2, ПК-3	З1, З2, З4, У1, У2, У3, У4, Н1, Н2
6.1. Представление графов. Поиск в глубину. Поиск в ширину		
6.2. Остовные деревья		
6.3. Связность графов		
6.4. Топологическая сортировка. Транзитивное замыкание		
6.5. Циклы в графе		
6.6. Кратчайшие пути		
Расчетно-графическая работа	ПК-2, ПК-3	З1, З2, З4, У1, У2, У3, У4, Н1, Н2
Зачет	ПК-2, ПК-3	З1, З2, З3, З4, У1, У2, У3, У4, Н1, Н2
Экзамен	ПК-2, ПК-3	З1, З2, З3, З4, У1, У2, У3, У4, Н1, Н2

4.2. Объем дисциплины, виды учебной работы обучающихся по очной форме обучения

Содержание	Всего, час	Контактная работа, час				СРС, час	ИФР, час	К, час
		Л	л/р	п/р	КСР			
Раздел 1. Алгоритмы и их сложности								
1.1. Введение	2	2						
1.2. Асимптотические обозначения	3	2				1		
1.3. Рекуррентные соотношения	4	2				2		
Раздел 2. Структуры данных								
2.1. Абстрактные типы данных. Пред-	6	4				2		

ставление множеств								
2.2. Стеки и очереди	8	2	4			2	4	
2.3. Деревья	6	2	2			2	2	
2.4. Прохождения деревьев	8	2	4			2	4	
2.5. Расширенные бинарные деревья	4	2				2		
Раздел 3 Исчерпывающий поиск								
3.1. Поиск с возвратом	8	4	2			2	2	
3.2. Метод ветвей и границ	10	6	2			2	2	
3.3. Методы решета	6	2	2			2	2	
3.4. Генерация элементарных комбинаторных объектов	3	2				1		
Зачет	4					4		
Раздел 4. Методы поиска								
4.1. Последовательный поиск. Логарифмический поиск в статических таблицах	5	2	2			1	2	
4.2. Логарифмический поиск в динамических таблицах	16	4	8			4	8	
4.3. Поиск с хешированием	6	2	2			2	2	
4.4. Внешний поиск	3	2				1		
Раздел 5 Сортировка								
5.1. Методы сортировки	3	2				1		
5.2. Внутренняя сортировка	16	4	8			4	8	
5.3. Внешняя сортировка	3	2				1		
Раздел 6 Алгоритмы на графах								
6.1. Представление графов. Поиск в глубину. Поиск в ширину	5	2	2			1	2	
6.2. Остовные деревья	8	4	2			2	2	
6.3. Связность графов	6	2	2			2	2	
6.4. Топологическая сортировка. Транзитивное замыкание	5	2	2			1	2	
6.5. Циклы в графе	5	2	2			1	2	
6.6. Кратчайшие пути	5	2	2			1	2	
Расчетно-графическая работа	22				2	20		
Экзамен	36							36
Итого	216, 6 з.е.	64	48		2	66	48	36

5. Содержание разделов дисциплины

5.1. Лекции

Раздел 1. Алгоритмы и их сложности

Тема 1.1. Введение

Лекция 1. Предмет дисциплины, ее объем, содержание и связь с другими дисциплинами учебного плана. Цели и задачи дисциплины. Алгоритмизация как основная стадия проектирования прикладных программ. Псевдокод для записи алгоритмов.

Тема 1.2. Асимптотические обозначения

Лекция 2. Временная и емкостная сложности алгоритмов. Асимптотическая сложность алгоритмов как скорость роста функций. Асимптотические обозначения скорости роста функций. Определение времени работы алгоритма и его асимптотической вычислительной сложности.

Тема 1.3. Рекуррентные соотношения

Лекция 3. Получение рекуррентных соотношений при анализе рекурсивных алгоритмов. Методы решения рекуррентных соотношений: метод подстановки, метод итераций, общий метод.

Раздел 2. Структуры данных

Тема 2.1. Абстрактные типы данных. Представление множеств

Лекция 4. Понятие данных. Прimitивные данные. Абстрактные типы данных. Структуры данных для представления множеств (мультимножеств и последовательностей). Последовательное распределение. Характеристический вектор.

Лекция 5. Связанное распределение. Использование динамических структур (связанных списков) для представления множеств. Реализация связанных списков. Разновидности связанных списков (списки с заголовком, циклические списки, дважды связанные списки).

Тема 2.2. Стеки и очереди.

Лекция 6. Стек как динамическая структура типа LIFO. Основные операции со стеком. Последовательная и связанная реализация стека. Очередь как динамическая структура типа FIFO. Основные операции с очередью. Последовательная и связанная реализация очереди. Очередь с приоритетами.

Тема 2.3. Деревья

Лекция 7. Понятие конечного корневого дерева, леса. Основные определения. Дерево (лес) как структура данных для представления множеств. Бинарное дерево. Естественное соответствие между лесами и бинарными деревьями. Представления деревьев.

Тема 2.4. Прохождения деревьев

Лекция 8. Основные прохождения деревьев: прямое, обратное, горизонтальное и симметричное (для бинарных деревьев). Алгоритмы прохождения деревьев. Повышение эффективности прохождения деревьев. Прошитые бинарные деревья. Алгоритмы прохождения.

Тема 2.5. Расширенные бинарные деревья

Лекция 9. Деревья как математический инструмент анализа алгоритмов. Расширенные бинарные деревья. Длины внутренних и внешних путей. Средняя и минимальная длина путей. Полностью сбалансированные деревья.

Раздел 3. Исчерпывающий поиск

Тема 3.1. Поиск с возвратом

Лекция 10. Область применения поиска с возвратом. Общий алгоритм поиска с возвратом. Дерево поиска. Пример применения общего алгоритма к решению конкретной задачи.

Лекция 11. Повышение эффективности поиска с возвратом (поиск с ограничениями, эквивалентность решений, слияние ветвей, переупорядочение поиска, декомпозиция). Оценка сложности поиска с возвратом (метод Монте-Карло). Способы программирования поиска с возвратом.

Тема 3.2. Метод ветвей и границ

Лекция 12. Метод ветвей и границ как разновидность поиска с возвратом для решения оптимизационных задач. Понятие нижней границы стоимости решений и его использование в качестве критерия для отсечения ветвей дерева поиска. Общий алгоритм метода ветвей и границ.

Лекция 13. Применение переупорядочения поиска для решения задачи коммивояжера методом ветвей и границ. Сравнение эффективности алгоритма, использующего переупорядочение поиска, с прямым применением общего алгоритма метода ветвей и границ.

Лекция 14. Метод альфа-бета отсечений как разновидность метода ветвей и границ для решения игровых задач. Принцип минимакса. Определение альфа- и бета-значений как нижней и верхней границы значений соответственно. Применение этих значений для отсечения ветвей (альфа- и бета-отсечения). Эвристические алгоритмы.

Тема 3.3. Методы решета

Лекция 15. Методы решета как альтернатива поиску с возвратом для решения теоретико-числовых задач. Модульное решето. Рекурсивное решето. Примеры решет.

Тема 3.4. Генерация элементарных комбинаторных объектов

Лекция 16. Общая схема систематического порождения комбинаторных объектов. Порождение перестановок в лексикографическом порядке. Использование вложенных

циклов для порождения перестановок. Порождение перестановок транспозицией смежных элементов. Порождение случайных перестановок. Порождение всех подмножеств множества. Порождение k -подмножеств (сочетаний из n по k). Порождение случайных сочетаний из n по k .

Раздел 4. Методы поиска

Тема 4.1. Последовательный поиск. Логарифмический поиск в статических таблицах

Лекция 17. Формулировка операции поиска. Статические и динамические таблицы. Основные табличные операции. Последовательный поиск. Методы повышения эффективности успешного и безуспешного поиска. Бинарный поиск. Однородный бинарный поиск. Поиск Фибоначчи. Интерполяционный поиск.

Тема 4.2. Логарифмический поиск в динамических таблицах

Лекция 18. Деревья бинарного поиска (ДБП). Реализация операций поиска, включения и исключения для ДБП. Недостатки ДБП без балансировки. ДБП, сбалансированные по высоте (АВЛ-деревья). Восстановление сбалансированности АВЛ-деревьев с помощью операций вращения и двойного вращения. Операции включения и исключения с балансировкой.

Лекция 19. Красно-черные деревья (RB-деревья) как пример еще одного способа организации сбалансированных ДБП. Включение и исключение с балансировкой в RB-деревьях. Цифровой поиск. Дерево цифрового поиска и его отличие от ДБП. Представление дерева цифрового поиска в памяти. Реализация операций поиска, включения и исключения.

Тема 4.3. Поиск с хешированием

Лекция 20. Основные идеи поиска с хешированием. Варианты хеширования. Методы построения хеш-функций и рекомендации по их выбору. Возникновение коллизий. Методы разрешения коллизий. Эффективность хеширования.

Тема 4.4. Внешний поиск

Лекция 21. Особенности внешнего поиска. Структуры данных для организации внешнего поиска. B-деревья. Реализация поиска, включения и исключения в B-деревьях.

Раздел 5. Сортировка

Тема 5.1. Методы сортировки

Лекция 22. Задача сортировки. Классификация методов сортировки. Внутренняя и внешняя сортировка. Нижние оценки эффективности алгоритмов сортировки.

Тема 5.2. Внутренняя сортировка

Лекция 23. Простая сортировка вставками. Анализ эффективности. Модификации простой сортировки вставками: сортировка бинарными вставками, сортировка вставками в связанный список. Сортировка с убывающим шагом (сортировка Шелла). Пузырьковая сортировка. Анализ эффективности. Модификация пузырьковой сортировки – шейкер-сортировка. Быстрая сортировка (рекурсивный и итерационный варианты). Методы выбора разделяющего имени для улучшения эффективности. Цифровая обменная сортировка.

Лекция 24. Простая сортировка выбором. Анализ эффективности. Пирамидальная сортировка. Понятие пирамиды. Процедура восстановления пирамиды. Построение пирамиды. Распределяющая сортировка. Сортировка подсчетом. Сортировка распределяющим подсчетом. Слияние двух и более отсортированных таблиц в одну. Сортировка слиянием (естественное и прямое двухпутевое слияние). Порядковые статистики.

Тема 5.3. Внешняя сортировка

Лекция 25. Особенности организации внешней сортировки. Порождение исходных отрезков (простое порождение и метод выбора с замещением). Распределение и слияние отрезков (равномерное распределение, многофазное слияние).

Раздел 6. Алгоритмы на графах

Тема 6.1. Представления графов. Поиск в глубину. Поиск в ширину

Лекция 26. Основы теории графов. Представления графов: матрица смежности, матрица инцидентий, матрица весов, список ребер, структура смежности. Основные методы систематического исследования графов. Поиск в глубину. Поиск в ширину.

Тема 6.2. Остовные деревья

Лекция 27. Понятие остовного дерева графа. Построение остовных деревьев поиском в глубину и поиском в ширину. Остовное дерево ориентированного графа.

Лекция 28. Минимальное остовное дерево. Построение минимального остовного дерева. Алгоритм Крускала. Построение минимального остовного дерева. Алгоритм Дейкстры-Прима.

Тема 6.3. Связность графов

Лекция 29. Поиск связных компонент неориентированного графа. Поиск двусвязных компонент неориентированного графа. Поиск сильно связных компонент ориентированного графа.

Тема 6.4. Топологическая сортировка. Транзитивное замыкание

Лекция 30. Топологическая сортировка ориентированного графа. Транзитивное замыкание ориентированного графа. Алгоритм Уоршелла. Алгоритм Уоррена.

Тема 6.5. Циклы в графе

Лекция 31. Понятие фундаментального множества циклов графа. Алгоритм построения фундаментального множества циклов. Критерии существования эйлера цикла. Построение эйлера цикла. Построение гамильтонова цикла.

Тема 6.6. Кратчайшие пути

Лекция 32. Поиск кратчайших путей от фиксированной вершины. Алгоритм Дейкстры. Поиск кратчайших путей между всеми парами вершин. Алгоритм Флойда.

5.2. Лабораторные работы

Лабораторная работа 1. Стеки. Очереди (пособие [1], работа 1).

Лабораторная работа 2. Деревья. Бинарные деревья (пособие [1], работа 2).

Лабораторная работа 3. Исчерпывающий поиск (пособие [1], работа 3).

Лабораторная работа 4. Исследование методов поиска (пособие [1], работа 4).

Лабораторная работа 5. Исследование методов сортировки (указания [4]).

Лабораторная работа 6. Алгоритмы на графах (указания [4]).

Перечень и график выполнения лабораторных работ

№	Тема работы	Часы аудиторных занятий
1	Стеки. Очереди.	4
2	Деревья. Бинарные деревья.	6
3	Исчерпывающий поиск.	6
4	Исследование методов поиска.	12
5	Исследование методов сортировки.	8
6	Алгоритмы на графах.	12
	Всего	48

5.3. Вопросы для самостоятельной работы студента в соответствии с содержанием разделов дисциплины

Целью самостоятельной работы студента (СРС) является закрепление полученных теоретических знаний и приобретение практических навыков применения и исследования алгоритмов и структур данных при проектировании прикладных программ. СРС включает в себя самостоятельное изучение учебных вопросов, подготовку к лабораторным занятиям, выполнение расчетно-графической работы, подготовку к экзамену.

Перечень вопросов и заданий для самостоятельной работы по подготовке к лабораторным занятиям приводится в методических указаниях [4] в описании каждой лабораторной работы.

Перечень вопросов и заданий для самостоятельной работы по выполнению расчетно-графической работы приводится в методических указаниях [3].

Перечень тем для самостоятельного изучения учебных вопросов, закрепления теоретических знаний и приобретения практических навыков:

1. Определить время работы заданного алгоритма и его вычислительную сложность (тема 1.2).

2. Решить заданные рекуррентные соотношения методом подстановки, методом итераций, общим методом и сравнить результаты (тема 1.3).

3. Разработать алгоритмы и программы копирования списка, включения и исключения элементов для разновидностей связанных списков: циклические списки, дважды связанные списки, дважды связанные циклические списки (тема 2.1).

4. Для заданного произвольного бинарного дерева с n вершинами построить соответствующее ему полностью сбалансированное бинарное дерево с n вершинами. Определить среднюю и минимальную длину внутренних и внешних путей для этих деревьев и сравнить результаты (тема 2.5).

5. Программно реализовать рассмотренные на лекционных занятиях алгоритмы порождения перестановок в лексикографическом порядке, порождения перестановок транспозицией смежных элементов, порождения случайных перестановок, порождения всех подмножеств множества, порождения k -подмножеств (сочетаний из n по k), порождения случайных сочетаний из n по k (тема 3.4).

6. Разработать алгоритмы и программы поиска, включения и исключения элементов для Б-деревьев, 2-3-деревьев (тема 4.4).

6. Образовательные технологии

В соответствии со структурой образовательного процесса по дисциплине применяются следующие технологии:

- диагностики;
- целеполагания;
- управления процессом освоения учебной информации;
- применения знаний на практике, поиска новой учебной информации;
- организации совместной и самостоятельной деятельности обучающихся (учебно-познавательной, научно-исследовательской, частично-поисковой, репродуктивной, творческой и пр.);
- контроля качества и оценивания результатов образовательной деятельности.

В соответствии с требованиями ФГОС ВО для реализации компетентностного подхода при обучении дисциплине предусмотрено широкое использование в учебном процессе активных и интерактивных методов проведения занятий:

При обучении дисциплине применяются следующие формы занятий:

- лекции, направленные на получение новых и углубление научно-теоретических знаний, в том числе вводная лекция, информационная лекция, обзорная лекция, лекция-консультация, проблемная лекция, лекции-дискуссии, лекции-беседы и др.;
- лабораторные занятия, проводимые под руководством преподавателя в учебной лаборатории с использованием компьютеров и учебного оборудования, направленные на закрепление и получение новых умений и навыков, применение знаний и умений, полученных на теоретических занятиях, при решении практических задач и др.

Все занятия обеспечены мультимедийными средствами (проекторы, экраны) для повышения качества восприятия изучаемого материала. В образовательном процессе широко используются информационно-коммуникационные технологии.

Самостоятельная работа студентов – это планируемая работа студентов, выполняе-

мая по заданию при методическом руководстве преподавателя, но без его непосредственного участия. Формы самостоятельной работы студентов определяются содержанием учебной дисциплины, степенью подготовленности студентов. Они могут иметь учебный или учебно-исследовательский характер: подготовка к лабораторным работам, выполнение расчетно-графической работы и др.

Формами контроля самостоятельной работы выступают проверка письменных отчетов по результатам выполненных заданий и лабораторных работ, расчетно-графической работы. Результаты самостоятельной работы учитываются при оценке знаний на зачёте и экзамене.

7. Формы аттестации и оценочные материалы

Текущий контроль предназначен для проверки хода и качества усвоения учебного материала, стимулирования учебной работы обучающихся и совершенствования методики проведения занятий. Он проводится в ходе всех видов занятий в форме, избранной преподавателем.

Промежуточная аттестация имеет целью определить степень достижения учебных целей по учебной дисциплине и проводится в форме зачета (3 семестр) и экзамена (4 семестр). Принимается зачет и экзамен преподавателями, читающими лекции по данной учебной дисциплине в соответствии с перечнем основных вопросов, выносимых для контроля знаний обучающихся.

7.1. Вопросы и задачи к зачету

Зачет в семестре проставляется по итогам выполнения и защиты всех запланированных на семестр лабораторных работ.

7.2. Вопросы и задачи к экзамену

К экзамену допускаются студенты, получившие зачет и полностью выполнившие план следующего семестра, т.е. выполнившие и защитившие все лабораторные работы и расчетно-графическую работу.

Вопросы к экзамену:

1. Представление последовательностей.
2. Характеристические векторы.
3. Связанное распределение. Разновидности связанных списков.
4. Организация стеков.
5. Организация очередей.
6. Деревья. Машинные представления деревьев.
7. Прохождения деревьев.
8. Бинарные деревья. Естественное соответствие бинарных деревьев и деревьев общего вида.
9. Прямое прохождение бинарных деревьев.
10. Обратное прохождение бинарных деревьев.
11. Симметричное прохождение бинарных деревьев.
12. Горизонтальное прохождение бинарных деревьев.
13. Прохождение бинарного дерева, соответствующее горизонтальному прохождению леса.
14. Прошитые деревья. Прохождения прошитых деревьев.
15. Общий алгоритм поиска с возвратом.
16. Оценка сложности выполнения поиска с возвратом (метод Монте-Карло).
17. Способы программирования поиска с возвратом.
18. Метод ветвей и границ.
19. Эффективный метод решения задачи коммивояжера.
20. Метод ветвей и границ для решения игровых задач.

21. Эвристические методы решения задачи поиска с возвратом.
22. Методы решета. Нерекурсивное модульное решето.
23. Методы решета. Рекурсивное решето.
24. Порождение элементарных комбинаторных объектов.
25. Порождение перестановок в лексикографическом порядке.
26. Порождение перестановок транспозицией смежных элементов.
27. Порождение всех подмножеств множества.
28. Порождение k -подмножеств (сочетаний из n по k).
29. Последовательный поиск.
30. Бинарный поиск в статических таблицах.
31. Равномерный бинарный поиск.
32. Поиск Фибоначчи.
33. Интерполяционный поиск.
34. Деревья бинарного поиска. Алгоритм поиска.
35. Включение узла в дерево бинарного поиска.
36. Исключение узла из дерева бинарного поиска.
37. Балансировка деревьев бинарного поиска. АВЛ-деревья.
38. Включение и исключение узла в АВЛ-дерево.
39. Красно-черные деревья.
40. Включение и исключение узла в красно-черное дерево.
41. Цифровой поиск.
42. В-деревья.
43. Включение узла в В-дерево.
44. Исключение узла из В-дерева.
45. Хеширование для варианта с малым пространством имен и варианта статической таблицы.
46. Хеширование для динамических таблиц.
47. Хеш-функции.
48. Методы разрешения коллизий.
49. Сортировка. Оценки эффективности алгоритмов сортировки.
50. Сортировка вставками.
51. Сортировка Шелла.
52. Пузырьковая сортировка.
53. Быстрая сортировка.
54. Сортировка выбором.
55. Пирамидальная сортировка.
56. Цифровая распределяющая сортировка.
57. Сортировка естественным двухпутевым слиянием.
58. Сортировка простым двухпутевым слиянием.
59. Сортировка подсчетом.
60. Внешняя сортировка.
61. Внешняя сортировка. Порождение исходных отрезков.
62. Внешняя сортировка. Слияние отрезков.
63. Порядковые статистики.
64. Прямое и бинарное слияние.
65. Представления графов.
66. Поиск в глубину. Поиск в ширину.
67. Построение остовных деревьев. DFS-дерево. BFS-дерево.
68. Минимальное остовное дерево. Алгоритм Крускала.
69. Минимальное остовное дерево. Алгоритм Дейкстры-Прима.
70. Остовное дерево орграфов.
71. Поиск связных компонент графа.

72. Топологическая сортировка.
73. Определение двусвязных компонент графа.
74. Определение сильно связных компонент графа.
75. Поиск фундаментальных множеств циклов.
76. Транзитивное замыкание.
77. Кратчайшие пути от фиксированной вершины.
78. Кратчайшие пути между всеми парами вершин графа.
79. Эйлеров цикл. Гамильтонов цикл.

Примеры задач (все решения необходимо программно реализовать, для алгоритмов необходимо определить вычислительную сложность):

1. Разработать алгоритм, который в строке, содержащей три набора скобок (,),{,},[,], проверяет правильность их расстановки.
2. Разработать метод поддержания в одном линейном массиве двух стеков, при котором ни один из стеков не переполняется до тех пор, пока весь массив не будет заполнен. При этом стек никогда не перемещается внутри массива на другие позиции.
3. Разработать представления, реализующие необходимые операции для очередей с приоритетом. Выбор обосновать.
4. Разработать алгоритм ввода и формирования бинарного дерева.
5. Разработать алгоритм копирования бинарного дерева.
6. Разработать алгоритм копирования связанного списка.
7. Разработать алгоритм преобразования бинарного дерева в симметрично прошитое бинарное дерево.
8. Разработать алгоритм прохождения прошитого дерева в прямом порядке без использования стека.
9. Разработать алгоритм прохождения прошитого дерева в симметричном порядке без использования стека.
10. Разработать алгоритм решения задачи коммивояжера методом ближайшего соседа.
11. Разработать алгоритм вычисления обратной перестановки на месте.
12. Разработать алгоритм сортировки бинарными вставками.
13. Разработать алгоритм сортировки вставками с использованием связанного списка.
14. Разработать алгоритм шейкер-сортировки.
15. Разработать алгоритм цифровой распределяющей сортировки.

7.3. Выполнение и примерные задания расчетно-графической работы

Расчетно-графическая работа выполняется в процессе изучения дисциплины. Общее руководство и контроль за ходом выполнения расчетно-графической работы осуществляет преподаватель соответствующей дисциплины. Расчетно-графическая работа выполняется в соответствии с методическими указаниями для обучающихся [3].

Основными функциями руководителя расчетно-графической работы являются:

- определение и формулирование задания расчетно-графической работы;
- консультирование по вопросам содержания и последовательности выполнения расчетно-графической работы;
- оказание помощи студенту в подборе необходимой литературы;
- контроль хода выполнения расчетно-графической работы.

Примерные задания для выполнения расчетно-графической работы:

1. Спелеолог

Вы попали в трехмерную пещеру и вам необходимо найти кратчайший путь к выходу. Пещера представляет собой куб, в котором есть проходы. Перемещение в любом направлении (вверх, вниз, вправо, влево, вперед, назад) занимает ровно одну минуту. Перемещаться по диагонали и через стены пещеры не разрешается. Возможен ли выход из та-

кой пещеры и если «да», то сколько времени вам понадобится? Если существует множество решений, определить наилучшее. Входные данные: описание пещеры.

Исследовать асимптотическую временную сложность решения задачи в зависимости от размеров пещеры.

2. Разбиение массива

Массив натуральных чисел $A (A_1, \dots, A_n)$ разбить на два непересекающиеся массива B и C (то есть каждый элемент массива A должен попасть точно в один из двух массивов: B или C), так, чтобы сумма чисел в B равнялась сумме чисел в C .

Входными данными являются количество чисел (n) и последовательность из n чисел.

Исследовать асимптотическую временную сложность решения задачи в зависимости от n .

3. Зашифрованные операции

Задаются арифметические операции, в которых цифры заменены буквами. В данной операции одна и та же буква всегда заменяет одну и ту же цифру, разные буквы представляют разные цифры. Наборы букв генерируются случайным образом, знак операции – символом, не являющимся буквой. Число разрядов исходных чисел (не результат операции) – не более N . Восстановить все возможные значения букв и операций.

Исследовать асимптотическую временную сложность решения задачи в зависимости от N .

Пример: SEND MORE MONEY соответствует $9567+1085=10652$.

4. Шашки

Имеется набор из 24 шашек, образованных следующим образом:

- двойной набор из 10 шашек с числами от 1 до 10;

- четыре шашки с числами 25, 50, 75 и 100.

Из этого набора случайным образом выбирается N шашек. Случайным образом выбирается трехзначное число K (первая цифра которого – не ноль). Задача заключается в том, чтобы соединить значения шашек между собой с помощью арифметических операций (сложение, вычитание, умножение, целочисленное деление), чтобы получить число K . Не обязательно использовать все арифметические операции и все N шашек.

Исследовать асимптотическую временную сложность решения задачи в зависимости от N .

5. Амазонки

Есть много вариантов шахматной игры: играют на досках различной формы, добавляют необычные фигуры и т. д. Новые фигуры можно получить, соединив возможности нескольких известных фигур. Например, амазонка – это фигура, которая может ходить как ферзь и как конь. Найти все способы расстановки n амазонок на шахматной доске размером $n \times n$ полей. Амазонки должны быть расставлены так, чтобы они не угрожали друг другу.

Исследовать асимптотическую временную сложность решения задачи в зависимости от n .

8. Учебно-методическое и информационное обеспечение дисциплины

Электронный каталог и электронные информационные ресурсы, предоставляемые научной библиотекой ФГБОУ ВО «Чувашский государственный университет имени И.Н. Ульянова» доступны по ссылке <http://library.chuvsu.ru/>

8.1. Рекомендуемая основная литература

№ п/п	Наименование
1.	Павлов, Л.А. Структуры и алгоритмы обработки данных: учеб. пособие / Л.А. Павлов.– Чебоксары: Изд-во Чуваш. ун-та, 2008.– 252 с.
2.	Никлаус Вирт Алгоритмы и структуры данных [Электронный ресурс] / Вирт Никлаус. — Электрон. текстовые данные. — Саратов: Профобразование, 2017. — 272 с. — 978-5-4488-0101-3. — Режим доступа: http://www.iprbookshop.ru/63821.html

8.2. Рекомендуемая дополнительная литература

№ п/п	Наименование
3.	Структуры и алгоритмы обработки данных: метод. указания к выполнению расчетно-графической работы / сост. Л.А. Павлов.– Чебоксары: Изд-во Чуваш. ун-та, 2014. – 24 с.
4.	Структуры и алгоритмы обработки данных: метод. указания к лабораторным занятиям / сост. Л.А. Павлов.– Чебоксары: Чуваш. ун-т, 2002. – 52 с.
5.	Комлева, Н.В. Структуры и алгоритмы компьютерной обработки данных [Электронный ресурс]: учебное пособие / Н.В. Комлева. – Электрон. текстовые данные. – М : Евразийский открытый институт, Московский государственный университет экономики, статистики и информатики, 2004. – 140 с. Режим доступа: http://www.iprbookshop.ru/10898.html
6.	Самуйлов, С.В. Алгоритмы и структуры обработки данных [Электронный ресурс]: учебное пособие / С.В. Самуйлов. – Электрон. текстовые данные. – Саратов: Вузовское образование, 2016. — 132 с. Режим доступа: http://www.iprbookshop.ru/47275.html

8.3. Рекомендуемые методические разработки по дисциплине

№ п/п	Наименование	Условия доступа
7.	Павлов, Л.А. Структуры и алгоритмы обработки данных: учеб. пособие	Сервер кафедры
8.	Структуры и алгоритмы обработки данных: метод. указания к лабораторным занятиям	Сервер кафедры
9.	Структуры и алгоритмы обработки данных: метод. указания к выполнению расчетно-графической работы	Сервер кафедры

8.4. Программное обеспечение, профессиональные базы данных, информационно-справочные системы.

Программное обеспечение, профессиональные базы данных, информационно-справочные системы, предоставляемые управлением информатизации ФГБОУ ВО «Чувашский государственный университет имени И.Н. Ульянова» доступны по ссылке <http://ui.chuvsu.ru/>

8.4.1. Программное обеспечение

№ п/п	Наименование	Условия доступа/скачивания
1.	Visual Studio Community	http://www.visualstudio.com/ru/vs/community
2.	MS Office/ LibreOffice	лицензия университета/ свободное лицензионное соглашение (https://ru.libreoffice.org/)
3.	MS Windows/Linux (Ubuntu)	лицензия университета/ свободное лицензионное соглашение (http://ubuntu.ru/)

8.4.2. Базы данных, информационно-справочные системы

№ п/п	Наименование программного обеспечения	Условия доступа/скачивания
1.	Гарант	из внутренней сети университета (договор)*
2.	Консультант +	

8.4.3. Рекомендуемые интернет-ресурсы и открытые он-лайн курсы

№ п/п	Наименование интернет ресурса	Режим доступа
1.	Российская Государственная Библиотека	http://www.rsl.ru
2.	Государственная публичная научно-техническая библиотека России	http://www.gpntb.ru
3.	Фундаментальная библиотека Нижегородского государственного университета	http://www.unn.ru/library

4.	Научная библиотека Казанского государственного университета	http://lsl.ksu.ru
5.	Научная электронная библиотека	http://elibrary.ru
6.	Полнотекстовая библиотека учебных и учебно-методических материалов	http://window.edu.ru
7.	Электронно-библиотечная система IPRbooks	http://www.iprbookshop.ru
8.	Сайт алгоритмов и методов вычислений	URL: http://www.algolist.manual.ru/
9.	Ванькина, Г.В. Структуры и алгоритмы компьютерной обработки данных [Электронный ресурс] /: Г.В. Ванькина, Т.О. Сундукова. – М.: Интернет-Университет Информационных Технологий (ИНТУИТ)	URL: http://www.intuit.ru/department/algorithms/staldata/

9. Материально-техническое обеспечение дисциплины

Учебные аудитории для лекционных занятий по дисциплине оснащены автоматизированным рабочим местом (АРМ) преподавателя, обеспечивающим тематические иллюстрации и демонстрации, соответствующие программе дисциплины в составе:

- ПЭВМ с доступом в интернет (операционная система, офисные программы, антивирусные программы);
- мультимедийный проектор с дистанционным управлением;
- настенный экран.

Учебные аудитории для лабораторных и самостоятельных занятий по дисциплине оснащены АРМ преподавателя и пользовательскими АРМ по числу обучающихся, объединенных локальной сетью («компьютерный» класс), с возможностью подключения к сети Интернет и доступом к электронной информационно-образовательной среде ФГБОУ ВО «Чувашский государственный университет имени И.Н. Ульянова».

10. Средства адаптации преподавания дисциплины к потребностям лиц с ограниченными возможностями

В случае необходимости, обучающимся из числа лиц с ограниченными возможностями здоровья (по заявлению обучающегося) могут предлагаться одни из следующих вариантов восприятия информации с учетом их индивидуальных психофизических особенностей, в том числе с применением электронного обучения и дистанционных технологий:

- для лиц с нарушениями зрения: в печатной форме увеличенным шрифтом; в форме электронного документа; в форме аудиофайла (перевод учебных материалов в аудиоформат); в печатной форме на языке Брайля; индивидуальные консультации с привлечением тифлосурдопереводчика; индивидуальные задания и консультации.

- для лиц с нарушениями слуха: в печатной форме; в форме электронного документа; видеоматериалы с субтитрами; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания и консультации.

- для лиц с нарушениями опорно-двигательного аппарата: в печатной форме; в форме электронного документа; в форме аудиофайла; индивидуальные задания и консультации.

11. Методические рекомендации по освоению дисциплины

В ходе лекционных занятий студенту рекомендуется вести конспектирование учебного материала. Следует обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации, положительный опыт в ораторском искусстве. При составлении конспекта желательно оставлять в рабочих конспектах поля, на которых в дальнейшем можно делать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений. В ходе лекционных занятий рекомендуется задавать преподавателю уточняющие вопросы с целью выяснения теоретических положений, разрешения спорных ситуаций.

В ходе подготовки к лабораторным работам рекомендуется изучить основную литературу, ознакомиться с дополнительной литературой, новыми публикациями в периодических изданиях. Основой для выполнения лабораторной работы являются разработанные кафедрой методические указания. Рекомендуется дорабатывать свой конспект лекций, делая в нем соответствующие записи из литературы, рекомендованной преподавателем и предусмотренной рабочей программой дисциплины. В процессе подготовки студент может дополнить список использованной литературы современными источниками, не представленными в списке рекомендованной литературы, и в дальнейшем использовать собственные подготовленные учебные материалы при написании расчетно-графической работы.

Форма организации студентов на лабораторных работах: фронтально-индивидуальная. Все студенты выполняют одновременно одну и ту же работу по индивидуальному заданию в соответствии со порядковым номером студента в списке группы.

В результате выполнения лабораторной работы запланирована подготовка письменного отчета в соответствии с требованиями методических указаний. Качество выполнения лабораторных работ является важной составляющей оценки текущей успеваемости обучающегося.

Изменения и (или) дополнения от 01.09.2018 г (протокол №1 МК факультета ИВТ) к рабочей программе дисциплины «Структуры и алгоритмы обработки данных» (направление подготовки 09.03.01 «Информатика и вычислительная техника», профиль «Программное обеспечение средств вычислительной техники и автоматизированных систем»):


к перечню учебной литературы и ресурсов сети «Интернет»

№ п/п	Рекомендуемая основная литература
1	Медведев Д.М. Структуры и алгоритмы обработки данных в системах автоматизации и управления [Электронный ресурс] : учебное пособие / Д.М. Медведев. — Электрон. текстовые данные. — Саратов: Ай Пи Эр Медиа, 2018. — 100 с. — 978-5-4486-0192-7. — Режим доступа: http://www.iprbookshop.ru/71591.html
2	Никлаус Вирт Алгоритмы и структуры данных [Электронный ресурс] / Вирт Никлаус. — Электрон. текстовые данные. — Саратов: Профобразование, 2017. — 272 с. — 978-5-4488-0101-3. — Режим доступа: http://www.iprbookshop.ru/63821.html
3	Павлов, Л.А. Структуры и алгоритмы обработки данных: учеб. пособие / Л.А. Павлов.— Чебоксары: Изд-во Чуваш. ун-та, 2008.— 252 с.
Рекомендуемая дополнительная литература	
4	Структуры и алгоритмы обработки данных: метод. указания к выполнению расчетно-графической работы / сост. Л.А. Павлов.— Чебоксары: Изд-во Чуваш. ун-та, 2014. — 24 с.
5	Структуры и алгоритмы обработки данных: метод. указания к лабораторным занятиям / сост. Л.А. Павлов.— Чебоксары: Чуваш. ун-т, 2002. — 52 с.
6	Батищев Р.В. Структуры и алгоритмы обработки данных. Часть 1 [Электронный ресурс] : учебное пособие / Р.В. Батищев. — Электрон. текстовые данные. — Липецк: Липецкий государственный технический университет, ЭБС АСВ, 2014. — 90 с. — 5-88247-716-6. — Режим доступа: http://www.iprbookshop.ru/55658.html
7	Комлева Н.В. Структуры и алгоритмы компьютерной обработки данных [Электронный ресурс] : учебное пособие / Н.В. Комлева. — Электрон. текстовые данные. — М. : Евразийский открытый институт, Московский государственный университет экономики, статистики и информатики, 2004. — 140 с. — 5-7764-0400-2. — Режим доступа: http://www.iprbookshop.ru/10898.html
8	Самуйлов С.В. Алгоритмы и структуры обработки данных [Электронный ресурс] : учебное пособие / С.В. Самуйлов. — Электрон. текстовые данные. — Саратов: Вузовское образование, 2016. — 132 с. — 2227-8397. — Режим доступа: http://www.iprbookshop.ru/47275.html

к перечню информационных технологий, включая перечень программного обеспечения и информационных справочных систем

№ п/п	Наименование Рекомендуемого ПО	Условия доступа/скачивания
1.	Microsoft Visual Studio	свободное лицензионное соглашение: https://visualstudio.microsoft.com/ru/free-developer-offers/
2.	Microsoft Windows	из внутренней сети университета (договор)
3.	Microsoft Office	

Декан факультета

 — А.В. Щипцова