

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
СТАРООСКОЛЬСКИЙ ТЕХНОЛОГИЧЕСКИЙ ИНСТИТУТ ИМ. А.А. УГАРОВА
(филиал) федерального государственного автономного образовательного учреждения
высшего образования
«Национальный исследовательский технологический университет «МИСИС»
СТИ НИТУ «МИСИС»

Рабочая программа утверждена
решением Ученого совета
СТИ НИТУ «МИСИС»
от «23» июня 2026 г.
протокол № 38

Рабочая программа дисциплины

Системное программное обеспечение

Закреплена за кафедрой	<u>Кафедра автоматизированных и информационных систем управления</u>
Направление подготовки	<u>им. Ю.И. Ерёмченко</u>
Профиль	15.04.04 Автоматизация технологических процессов и производств
	Системы автоматизированного анализа, обработки данных и управления
Квалификация	<u>Магистратура</u>
Форма обучения	<u>Очная</u>

Общая трудоемкость 3 ЗЕТ

Часов по учебному плану	<u>108</u>
в том числе:	
аудиторные занятия	<u>17</u>
самостоятельная работа	<u>91</u>
часов на контроль	<u>-</u>

Формы контроля в семестре: 1
Зачет,

Семестр	1		Итого	
Вид занятий	УП	РП	УП	РП
Лекции	0	0	0	0
Лабораторные	0	0	0	0
Практические занятия	17	17	17	17
Сам. работа	91	91	91	91
Часов на контроль	0	0	0	0
Итого:	108	108	108	108

Год набора 2026 г.

Программу составил: ассистент кафедры. АИСУ им.
Ю.И. Ерёменко, кандидат технических наук,
Жуков Пётр Игоревич

Должность, уч. ст., уч. зв. ФИО полностью


подпись

Рабочая программа дисциплины

Системное программное обеспечение

наименование

Разработана в соответствии с ОС ВО НИТУ МИСИС:

Самостоятельно устанавливаемый образовательный стандарт высшего образования – бакалавриат
Федеральное государственное автономное образовательное учреждение высшего образования
«Национальный исследовательский технологический университет «МИСИС» по направлению подготовки
15.04.04 Автоматизация технологических процессов и производств (приказ от 2.04.21 г. № 119 о.в.)

Составлена на основании учебного плана 2026 года набора.

15.04.04 Автоматизация технологических процессов и производств,

Профиль: Системы автоматизированного анализа, обработки данных и управления, утвержденного Ученым
советом СТИ НИТУ «МИСИС» 23.06.2026 г., протокол № 38.

Рабочая программа одобрена на заседании кафедры

Автоматизированные и информационные системы управления им. Ю.И. Ерёменко

Протокол от «22» июня 2026 г. № 6.

и.о. зав.
кафедрой

АИСУ им. Ю.И.
Ерёменко
аббревиатура наименования кафедры


подпись

Д.А. Полещенко

И.О. Фамилия

«22» июня 2026 г.

Руководитель ОПОП ВО
и.о. зав. кафедрой АИСУ им. Ю.И.
Ерёменко, кандидат технических наук,
доцент

должность, уч. ст., уч. зв.


подпись

Д.А. Полещенко

И.О. Фамилия

«22» июня 2026 г.

1. ЦЕЛИ ОСВОЕНИЯ	
Цель дисциплины – получить теоретические знания по теории алгоритмов и структур данных. Освоить практические навыки работы с наиболее популярными, на текущий момент, структурами данных.	
Задачи дисциплины:	
1. Познакомить обучающихся со сложными не типичными структурами данных (хэш-таблицы, древовидные структуры, коллекции); 2. Научить обучающихся базовым алгоритмам обработки таких структур данных; 3. Научить обучающихся применять базовые алгоритмы в реальных задачах со сложными структурами данных и получать на их основе собственные алгоритмы обработки.	

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО	
Цикл (раздел) ОП:	Б1.В.
2.1	Требования к предварительной подготовке обучающихся:
2.1.1	Программирование не языках высокого уровня
2.1.2	Объектно-ориентированное программирование
2.1.3	Системный анализ
2.2	Дисциплины и практики, для которых освоение данной дисциплины необходимо как предшествующее:
2.2.1	Научно-исследовательская работа

3. ИНДИКАТОРЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ, СОВМЕЩЕННЫЕ С РЕЗУЛЬТАТАМИ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ	
ОПК-11: Способен разрабатывать современные методы исследования автоматизированного оборудования в машиностроении	
Знать:	ОПК-11-31 – Базовые алгоритмы для обработки пространственных и временных структур данных.
Уметь:	ОПК-11-У1 – Использовать базовые алгоритмы в реальных задачах.
Владеть:	ОПК-11-В1 – Программированием на языке высокого уровня
ОПК-12: Способен:	
• проектировать и разрабатывать продукцию, процессы и системы в условиях неопределенности и альтернативных решений в междисциплинарных областях, соответствующих профилю подготовки; • разрабатывать и оптимизировать алгоритмы и современные цифровые системы автоматизированного проектирования технологических процессов; • создавать программы изготовления деталей и узлов различной сложности на станках с числовым программным управлением; • проектировать алгоритмы функционирования гибких производственных систем	
Знать:	ОПК-12-31 – Теорию построения алгоритмов систем с дискретным временем жизни.
Уметь:	ОПК-12-У1 – Применять базовые алгоритмы для исследования различных структур данных.
Владеть:	ОПК-12-В1 – Инструментами описания алгоритмов и алгоритмических структур.
ПК-1: Способен:	
• проводить анализ, синтез и оптимизацию процессов автоматизации, управления производством, жизненным циклом продукции и ее качеством на основе проблемно-ориентированных методов, в том числе, методов искусственного интеллекта; • проводить математическое моделирование процессов, оборудования, средств и систем автоматизации, контроля, диагностики, испытаний и управления с использованием современных технологий научных исследований; • разрабатывать структуру АСУТП (АСУП), организационное, информационное, алгоритмическое и программное обеспечение средств и систем автоматизации и управления; • выполнять анализ состояния и динамики функционирования средств и систем автоматизации, контроля, диагностики, испытаний и управления качеством продукции, метрологического и нормативного обеспечения производства.	
Знать:	ПК-1-31 – Теорию алгоритмической и пространственной сложности систем
Уметь:	ПК-1-У1 – Проводить анализ алгоритмической сложности систем
Владеть:	ПК-1-В1 – Нотацией асимптотической сложности дискретных систем (Big-O).
ПК-2: Способен:	
• аккумулировать научно-техническую информацию, отечественный и зарубежный опыт в области автоматизации технологических процессов и производств, искусственного интеллекта, инжиниринга знаний; • составлять научные отчеты по выполненному заданию и участвовать во внедрении результатов исследований и разработок в области автоматизации технологических процессов и производств; • способность участвовать в работах по моделированию продукции, технологических процессов, средств и систем автоматизации, контроля и управления процессами с использованием современных средств автоматизированного проектирования, по разработке алгоритмического и программного обеспечения средств и систем автоматизации и управления процессами.	
Знать:	ПК-2-31 – Интегрированные среды разработки и проектирования ПО на языках высокого уровня
Уметь:	ПК-2-У1 – Реализовывать спроектированные алгоритмы на языках высокого уровня.
Владеть:	ПК-2-В1 – Навыками чтения программного кода на языках высокого уровня

4. СТРУКТУРА И СОДЕРЖАНИЕ						
Код занятия	Наименование разделов и тем /вид занятия/	Семестр	Часов	Компетенции	Литература и эл. ресурсы	Примечание
Раздел №1: Структуры данных. Алгоритмы поиска и сортировки.						
1.1.	Введение в теорию алгоритмов и структур данных. Введение в нотацию Big-O /Пр/	1	1	ОПК-11-31 ОПК-12-31 ПК-1-31 ПК-2-31	Л1.1, Л1.2, Л3.1, Э1, Э2, Э3	
1.2	Массивы и списки. Базовые структуры данных /Пр/	1	2	ОПК-11-У1 ОПК-12-У1 ОПК-11-В1 ОПК-12-В1	Л1.1, Л1.2, Л3.1, Э1, Э2, Э3	
1.3	Алгоритмы сортировки и поиска. Применение наиболее эффективных алгоритмов сортировки к массивам и спискам /Пр/	1	2	ПК-2-У1 ПК-2-31 ПК-1-У1 ПК-1-В1 ОПК-11-В1	Л1.1, Л1.2, Л3.1, Э1, Э2, Э3	
1.4	Оптимизация базовых алгоритмов сортировки при работе с двумерными массивами. Приведение $O(n^2)$ к $O(n \log n)$ Подготовка к практическим работам /Ср/	1	30	ПК-1-У1 ПК-1-В1 ПК-2-У1 ПК-2-В1 ОПК-11-В1 ОПК-12-В1	Л1.1, Л3.2, Э1, Э2, Э3	
Раздел №2: Древовидные структуры данных. Алгоритмы обхода деревьев.						
2.1	Графы и красно-черные деревья. Алгоритмы обхода графов /Пр/	1	4	ОПК-11-31 ОПК-12-31 ПК-1-31 ПК-2-31	Л1.1, Л1.2, Л3.1, Э1, Э2, Э3	
2.2	Программная реализация алгоритма обхода в глубину и алгоритма обхода в ширину /Пр/	1	2	ОПК-11-У1 ОПК-12-У1 ОПК-11-В1 ОПК-12-В1	Л1.1, Л1.2, Л3.1, Э1, Э2, Э3	
2.3	Алгоритмы Дейкстры и Флойда-Уоршелла для поиска оптимального пути /Пр/	1	2	ПК-2-У1 ПК-2-31 ПК-1-У1 ПК-1-В1 ОПК-11-В1	Л1.1, Л1.2, Л3.1, Э1, Э2, Э3	
2.4	Наиболее оптимальные способы хранения древовидных структур. Перенос двумерного массива в дерево не дольше, чем за $O(n^2)$. Точечный поиск в дереве. Подготовка к практическим работам / Ср/	1	31	ПК-1-У1 ПК-1-В1 ПК-2-У1 ПК-2-В1 ОПК-11-В1 ОПК-12-В1	Л1.1, Л3.2, Э1, Э2, Э3	
Раздел №3: Хэш-таблицы						
3.1	Хэш-таблицы, как наиболее универсальная структура данных/Пр/	1	2	ОПК-11-31 ОПК-12-31 ПК-1-31 ПК-2-31	Л1.1, Л1.2, Л3.1, Э1, Э2, Э3	
3.2	Примеры применения хэш-таблиц. Очереди и стеки /Пр/	1	2	ОПК-11-У1 ОПК-12-У1 ОПК-11-В1 ОПК-12-В1	Л1.1, Л1.2, Л3.1, Э1, Э2, Э3	
3.3	Описание структуры искусственной нейронной сети при помощи хэш-таблиц на языке высокого уровня. Подготовка к практическим работам. Подготовка к зачету по дисциплине/ Ср/	1	30	ПК-1-У1 ПК-1-В1 ПК-2-У1 ПК-2-В1 ОПК-11-В1 ОПК-12-В1	Л1.1, Л3.2, Э1, Э2, Э3	

5. ФОНД ОЦЕНОЧНЫХ МАТЕРИАЛОВ	
5.1. Вопросы для самостоятельной подготовки к экзамену (зачёту с оценкой)	
Экзамен не предусмотрен	
5.2. Перечень работ, выполняемых по дисциплине	
В рамках данного курса предполагается выполнение следующей учебной нагрузки. Практические работы: - Практическая работа №1. Введение в теорию алгоритмов и структур данных. Введение в нотацию Big-O (ОПК-11-31, ОПК-12-31, ПК-1-31, ПК-2-31); - Практическая работа №2. Массивы и списки. Базовые структуры данных (ОПК-11-У1, ОПК-12-У1, ОПК-11-В1, ОПК-12-В1); - Практическая работа №3 Алгоритмы сортировки. Применение наиболее эффективных алгоритмов сортировки к массивам и спискам (ПК-2-У1, ПК-2-31, ПК-1-У1, ПК-1-В1, ОПК-11-В1); - Практическая работа №4 Графы и красно-черные деревья. Алгоритмы обхода графов (ОПК-11-31, ОПК-12-31, ПК-1-31, ПК-2-31); - Практическая работа №5 Программная реализация алгоритма обхода в глубину и алгоритма обхода в ширину (ОПК-11-У1, ОПК-12-У1, ОПК-11-В1, ОПК-12-В1); - Практическая работа №6 Алгоритмы Дейкстры и Флойда-Уоршелла для поиска оптимального пути (ПК-2-У1, ПК-2-31, ПК-1-У1, ПК-1-В1, ОПК-11-В1); - Практическая работа №7 Хэш-таблицы, как наиболее универсальная структура данных (ОПК-11-31, ОПК-12-31, ПК-1-31, ПК-2-31); - Практическая работа №8 Примеры применения хэш-таблиц (ОПК-11-У1, ОПК-12-У1, ОПК-11-В1, ОПК-12-В1). Домашние задания (ПК-1-У1, ПК-1-В1, ПК-2-У1, ПК-2-В1, ОПК-11-В1, ОПК-12-В1): - Домашнее задание №1. Оптимизация базовых алгоритмов сортировки при работе с двумерными массивами; - Домашнее задание №2. Наиболее оптимальные способы хранения древовидных структур. Перенос двумерного массива в дерево не дольше, чем за $O(n^2)$. Точечный поиск в дереве; - Домашнее задание №3. Описание структуры искусственной нейронной сети при помощи хэш-таблиц на языке высокого уровня	
5.2.1 Перечень вопросов к защите практических работ	
Практическая работа №1. Введение в теорию алгоритмов и структур данных. Введение в нотацию Big-O (ОПК-11-31, ОПК-12-31, ПК-1-31, ПК-2-31) - Что такое асимптотический класс сложности? - Везде ли применим асимптотический класс сложности? Приведите пример. - Чем анализ асимптотической сложности отличается от практического анализа вычислительных алгоритмов? - Дайте определения NP-задачи. Дайте определение NP-полной задачи. - Дайте определения термину «разрешимость» задачи. Практическая работа №2. Массивы и списки. Базовые структуры данных (ОПК-11-У1, ОПК-12-У1, ОПК-11-В1, ОПК-12-В1) - Чем одномерный массив отличается от списка. - Дайте понятие «многомерности» у массивов. - Какова сложность обхода двумерного массива и почему? - Какие структуры данных называются базовыми? - Какие структуры данных называются статическими, а какие относят к динамическим? - Является ли файловая структура структурой данных? Практическая работа №3 Алгоритмы сортировки и поиска. Применение наиболее эффективных алгоритмов сортировки к массивам и спискам (ПК-2-У1, ПК-2-31, ПК-1-У1, ПК-1-В1, ОПК-11-В1) - Какие задачи решает Bogosort? - Какова сложность QuickSort в худшем случае? - Приведите пример классической задачи сортировки. - Какие существуют два способа реализации алгоритмов сортировки? - Опишите алгоритм Кнута-Мориса-Пратта. - Опишите алгоритм Боуера-Мура. Практическая работа №4 Графы и красно-черные деревья. Алгоритмы обхода графов (ОПК-11-31, ОПК-12-31, ПК-1-31, ПК-2-31) - Дайте определение графу. - Какие задачи решаются при помощи графовых моделей? - Что такое бинарное дерево? - Какие задачи решаются при помощи бинарных деревьев? - Какие существуют алгоритмы обхода графов и деревьев? - Приведите пример построения элементарного нисходящего дерева. - Что такое рекурсия и для чего она применяется?	

Практическая работа №5 Программная реализация алгоритма обхода в глубину и алгоритма обхода в ширину
(ОПК-11-У1, ОПК-12-У1, ОПК-11-В1, ОПК-12-В1)

- Каков класс сложности DFS? Докажите.
- Каков класс сложности BFS? Докажите.
- Чем принципиально отличается алгоритм обхода в ширину от алгоритма обхода в глубину?
- Какой структурой данных лучше всего описать граф?

Практическая работа №6 Алгоритмы Дейкстры и Флойда-Уоршелла для поиска оптимального пути
(ПК-2-У1, ПК-2-В1, ПК-1-У1, ПК-1-В1, ОПК-11-В1)

- Каков класс сложности алгоритма Дейкстры? Почему?
- Назовите задачи, которые могут быть решены при помощи алгоритма Дейкстры.
- Каков класс сложности алгоритма Флойда-Уоршелла?
- Назовите ограничения применения алгоритма Флойда-Уоршелла?
- Приведите пример задачи, которую предпочтительней решить алгоритмом Флойда-Уоршелла, чем алгоритмом Дейкстры. Дайте пояснения почему.

Практическая работа №7 Хэш-таблицы, как наиболее универсальная структура данных
(ОПК-11-В1, ОПК-12-В1, ПК-1-В1, ПК-2-В1)

- Что такое хэш-функция? Какие её основные характеристики?
- Что такое ассоциативный массив? Чем он отличается от обычного массива?
- Как в хэш-функциях избегают коллизий?
- Как отсортировать хэш-таблицу?

Практическая работа №8 Примеры применения хэш-таблиц. Очереди и стеки
(ОПК-11-У1, ОПК-12-У1, ОПК-11-В1, ОПК-12-В1)

- Что такое стек? Для чего его можно использовать?
- Чем очередь отличается от стека?
- Назовите задачу, для которой хэш-таблица является более оправданным решением, чем двумерный массив.
- Назовите способ реализации очереди.

5.2.3 Перечень тестовых вопросов

Разделы №1-3 (ПК-1-У1, ПК-1-В1, ПК-2-У1, ПК-2-В1, ОПК-11-В1, ОПК-12-В1)

- 1) Какая сортировка из следующих является самой эффективной?
a) Шелла
b) Быстрая
c) Отбором
d) Вставками
e) Пузырьковая
- 2) Каким выражением определяется количество перестановок для пузырьковой сортировки в лучшем случае?
a) $(N-1)/2$
b) 0
c) $N-1$
d) $N(N-1)/4$
e) N^2
- 3) Какое минимальное количество полей может содержать каждое звено односвязного списка, при условии, что в нём содержится полезная информация?
a) 1
b) 3
c) 5
d) 2
e) 4
- 4) Имеется двоичное дерево (не являющееся деревом поиска), содержащее произвольные символы. Восходящий просмотр дерева даёт следующий результат: A, a, +, *, 1, \$, x. Какой узел является корнем дерева?
a) +
b) A
c) *
d) 1
e) x
- 5) Некоторый массив размером N был отсортирован за время, пропорциональное N^2 . По какому алгоритму выполнялась сортировка?
a) Шелла
b) Замыканием
c) Пузырьковая

- d) Ветвлением
e) Хоара
- 6) Какие основные операции над элементами характерны для списков?
a) Создание ведущего звена, вставка нового звена, удаление звена, поиск.
b) Занесение нового элемента в список, удаление элемента из списка, просмотр списка, поиск элемента в списке, сортировка списка.
c) Занесение нового элемента в список и извлечение элемента из списка.
d) Сортировка элементов списка, занесение элемента в список, извлечение элемента из списка и удаление списка.
e) Просмотр списка, поиск элемента в списке и сортировка списка.
- 7) Какое минимальное количество полей может содержать каждое звено двусвязного списка, при условии, что в нём содержится полезная информация?
a) 3
b) 2
c) 5
d) 4
e) 1
- 8) Имеется двоичное дерево поиска, содержащее целые числа от 1 до 7. Каким будет результат последовательного просмотра?
a) 7,6,5,4,3,2,1
b) 1,3,2,5,7,6,4
c) 4,2,1,3,6,5,7
d) 1,2,3,4,5,6,7
e) 4,2,6,1,3,5,7
- 9) В список с выделенным ведущим звеном подряд заносились целые числа, упорядоченные по возрастанию. Просмотр списка даёт следующий результат: 2, 4, 6, 8, 10. В какую позицию заносились данные?
a) В ведущее звено
b) Сразу после ведущего звена
c) Перед ведущим звеном
d) В середину списка
e) За последним звеном
- 10) Каким выражением определяется количество перестановок для пузырьковой сортировки в среднем случае?
a) N^2
b) $N(N-1)/4$
c) $N-1$
d) 0
e) $N(N-1)/2$
- 11) Имеется упорядоченный массив целых чисел из 15 элементов. Сколько операций сравнения потребуется при двоичном поиске для установления факта отсутствия искомых данных в этом массиве?
a) 14
b) 1
c) 5
d) 10
e) $\log_2(15)$
- 12) Имеется некоторая структура данных, в которую заносятся упорядоченные по убыванию символы. Считывание данных из этой структуры даёт результат: F, E, D, C, B, A. Чем является эта структура данных?
a) Очередь
b) Дерево
c) Стек
d) Связный список
e) Граф
- 13) Производится пузырьковая сортировка массива из 6 элементов, причём массив уже упорядочен в требуемом порядке. Сколько будет выполнено перестановок?
a) 0
b) 6
c) 30
d) 15
e) 7
- 14) Производится пузырьковая сортировка массива из 6 элементов, причём массив упорядочен в обратном порядке. Сколько будет выполнено перестановок?
a) 0
b) 6

- c) 30
- d) 15
- e) 7

15) Каков порядок алгоритмической сложности пузырьковой сортировки?

- a) N^2
- b) $N(N-1)/4$
- c) $N-1$
- d) 0
- e) $N(N-1)/2$

16) Имеется неупорядоченный массив целых чисел из 8 элементов. Сколько операций сравнения потребуется для нахождения искомого ключа, если он находится в конце массива?

- a) 8
- b) 1
- c) 7
- d) 0
- e) 4

17) Основное требование, предъявляемое к массиву для возможности выполнения двоичного поиска:

- a) Малый размер
- b) Нет особых требований
- c) Упорядоченность
- d) Неупорядоченность
- e) Большой размер

18) Чему равно значение указателя в последнем звене кольцевого односвязного списка?

- a) 1
- b) Случайному числу
- c) 0
- d) Адресу ведущего звена
- e) -1

19) Имеется идеально сбалансированное двоичное дерево поиска, содержащее целые числа. Просмотр дерева даёт следующий результат: 40, 20, 10, 30, 60, 50, 70. Какой способ просмотра дерева использовался?

- a) Пузловой
- b) Нисходящий
- c) Последовательный
- d) Восходящий
- e) Поуровневый

20) Какие позиции очереди доступны для занесения новых элементов?

- a) Все позиции, кроме начала очереди
- b) Начало очереди
- c) Начало и конец очереди
- d) Конец очереди
- e) Все позиции

21) Какая структура данных используется для сохранения и восстановления содержимого регистров общего назначения центрального процессора при вызове процедур?

- a) Очередь
- b) Список
- c) Стек
- d) Таблица
- e) Двоичное дерево

22) Какое из следующих высказываний наилучшим образом характеризует сортировку Хоара?

- a) Ищет наименьший или наибольший элемент
- b) Относится к самым быстрым в среднем случае из универсальных алгоритмов
- c) Считается самой простой
- d) Не подходит для 1-мерных массивов
- e) Выполняет наименьшее число операций

23) Имеется упорядоченный массив целых чисел. Для нахождения ключа используется двоичный поиск. Гарантируется ли в этом случае истинность результата поиска?

- a) Да
- b) Гарантируется при условии, что значение ключа не превышает размера массива
- c) Нет
- d) Гарантируется при условии, что в процедуре поиска используется цикл while
- e) Гарантируется при условии, что в процедуре поиска используется цикл for

- 24) Какое из следующих высказываний наилучшим образом характеризует сортировку отбором?
- a) Считается самой простой
 - b) Выполняет наименьшее число операций
 - c) Ищет наименьший или наибольший элемент
 - d) Считается самой быстрой
 - e) Не подходит для 1-мерных массивов
- 25) Имеется двоичное дерево поиска, содержащее целые числа. Последовательный просмотр дерева даёт следующий результат: 2, 4, 6, 8, 10, 12, 14. Какой узел является корнем дерева?
- a) 4
 - b) 2
 - c) 8
 - d) 6
 - e) 14
- 26) В процессе сортировки весь сортируемый массив и каждая его часть делятся на две части. По какому алгоритму выполняется эта сортировка?
- a) Быстрая
 - b) Шелла
 - c) Вставками
 - d) Отбором
 - e) Пузырьковая
- 27) Имеются 2 одинаковых ЭВМ - А и Б, в памяти каждой из которых содержится упорядоченный массив из 2000 целых неповторяющихся чисел от 1 до 2000. ЭВМ А ищет число 24 при помощи последовательного поиска, а ЭВМ Б ищет число 24 при помощи двоичного поиска. Какая из машин раньше закончит поиск?
- a) А
 - b) Обе одновременно
 - c) В обоих случаях будет выявлена ошибка
 - d) Б
 - e) Обе зайдутся
- 28) Какая следующая структура данных имеет наибольший объем служебной информации?
- a) Кольцевой односвязный список
 - b) Массив
 - c) Линейный односвязный список
 - d) Кольцевой односвязный список с ведущим звеном
 - e) Линейный двусвязный список
- 29) Сколько сравнений выполнится при пузырьковой сортировке массива из 10 элементов?
- a) 100
 - b) 90
 - c) 24
 - d) 45
 - e) 10
- 30) Имеется идеально сбалансированное двоичное дерево, узлы которого размещены на 6-и уровнях. Какое максимальное число узлов может быть в этом дереве?
- a) 127
 - b) 6
 - c) 64
 - d) 63
 - e) 7
- 31) Какое из следующих высказываний наилучшим образом характеризует пузырьковую сортировку?
- a) Не подходит для 1-мерных массивов
 - b) Выполняет наименьшее число операций
 - c) Считается самой простой
 - d) Считается самой быстрой
 - e) Ищет наименьший или наибольший элемент
- 32) Какая следующая структура данных имеет наименьший объем служебной информации?
- a) Массив
 - b) Кольцевой односвязный список с ведущим звеном
 - c) Линейный двусвязный список
 - d) Линейный односвязный список
 - e) Кольцевой односвязный список
- 33) Некоторый массив размером N был отсортирован за время, пропорциональное $n \log n$. По какому алгоритму

выполнялась сортировка?

- a) Случайная
- b) Замыканием
- c) Пузырьковая
- d) Шелла
- e) Ветвлением

34) По совету с Habr'a магистр принялся сортировать массив из 100 элементов при помощи алгоритма Хоара. В конце сортировки, на последних 15 элементах, из-за неудачных входных данных алгоритм сильно деградировал и замедлился. Сколько потребовалось времени, чтобы сортировать последний участок?

- a) $10t$
- b) $15t$
- c) $225t$
- d) $3375t$
- e) $150t$

35) Какие существуют метрики, отображающие эффективность алгоритма?

- a) процессорное время, память
- b) надежность, масштабируемость
- c) адаптивность, простота реализации

36) Считается ли компьютерная память важным ресурсом, учитываемым при разработке эффективного алгоритма?

- a) да
- b) нет

37) Зависит ли время работы алгоритма от размера входных данных N ?

- a) нет
- b) да

38) Считается ли процессорное время важным ресурсом, учитываемым при разработке эффективного алгоритма?

- a) да
- b) нет

39) Big-O дает приближенную оценку. Что нужно сделать, чтобы найти оценку точнее?

- a) выполнить большее количество тестов
- b) нужно для начала определиться, нас интересует оценка на фиксированный алгоритм или на задачу и выполнять оценку исходя из этого
- c) изменить входные данные

40) Что означает найти оценку для фиксированного алгоритма?

- a) нужно указать такую оценку, которая справедлива для всех мысленных алгоритмов
- b) нужно найти оценку снизу, сверху. Если оценки совпали, то оценка равна $\Theta(N)$. И как правило оценка сводится к наихудшему случаю
- c) означает что нужно найти среднюю оценку для алгоритма

41) Что означает найти оценку снизу на задачу?

- a) нужно указать такую оценку, которая справедлива для всех мысленных алгоритмов. То есть понять какие время и память точно понадобятся
- b) нужно найти оценки снизу, сверху. Если оценки совпали, то оценка равна $\Theta(N)$. И как правило оценка сводится к наихудшему случаю
- c) означает что нужно найти среднюю оценку для алгоритма

42) В чем состоит отличие в работе алгоритма для модели "разрешающие деревья" от RAM - модели и модели машины Тьюринга?

- a) алгоритм неограничен в своих действиях
- b) разрешено действие только одного типа
- c) в такой модели можно программировать

43) 11. К каким структурам данных в общем случае относится дерево?

- a) К статическим нелинейным
- b) К динамическим нелинейным
- c) К динамическим линейным
- d) К статическим линейным
- e) К кольцевым

44) Имеется идеально сбалансированное двоичное дерево, содержащее 31 узел. Какова высота этого дерева?

- a) 1 уровень
- b) 3 уровня

- c) 3 уровня
 - d) 5 уровней
 - e) 4 уровня
- 45) Основное достоинство последовательного поиска:
- a) Алгоритмическая простота
 - b) Малое время работы
 - c) Возможность работать с упорядоченными данными
 - d) Высокая скорость
 - e) Возможность эффективной работы с большими объемами данных
- 46) Какие существуют метрики, отображающие эффективность алгоритма?
- a) процессорное время, память
 - b) надежность, масштабируемость
 - c) адаптивность, простота реализации
- 47) В функциональной парадигме при проектировании алгоритма, какой оценкой на время работы интересуются?
- a) оценкой в худшем случае
 - b) оценкой в среднем
 - c) оценкой в лучшем случае
- 48) При размере входных данных N , как рассчитывается время работы алгоритма?
- a) не зависимо от N
 - b) в сравнении с N
 - c) как функция от параметра N
- 49) Возможна ли такая ситуация при проектировании алгоритма, когда можно сэкономить на одном ресурсе в ущерб другому (процессорное время / память)?
- c) нет
 - b) да
- 50) Если T - время работы алгоритма, N - размер входных данных, что отображает функция $\max T(I)$ для $N(I) = N$?
- a) время работы алгоритма в худшем случае для конкретного входа I
 - b) время работы алгоритма в лучшем случае при рассмотрении всех входов (I) размера N
 - c) время работы алгоритма в худшем случае при рассмотрении всех входов (I) размера N
- 51) При рассмотрении времени работы $T(M)$ и памяти $M(N)$ что нас интересует?
- a) точный вид функций $T(N)$ и $M(N)$
 - b) приближенный до константы вид функций. Используется O -символика
 - c) приближенный вид функций. Используется Θ -символика
- 52) При оценивании функций какая оценка соответствует символическому $f = O(g)$?
- a) оценка снизу
 - b) оценка сверху
 - c) асимптотическое равенство
- 53) При оценивании функций символическому $f = \Theta(g)$ соответствует:
- a) оценка снизу
 - b) оценка сверху
 - c) асимптотическое равенство
- 54) Если при оценивании фиксированного алгоритма оценки сверху и снизу совпали, то какие действия предпринимаются?
- a) время оценивается как $\Theta(N)$ и оценивание сводится к придумыванию наихудшего случая для алгоритма
 - b) берется сумма оценок сверху и снизу и делится на 2
 - c) это означает что оценка произведена неверно
- 55) Что означает найти оценку для фиксированного алгоритма?
- a) нужно указать такую оценку, которая справедлива для всех мысленных алгоритмов
 - b) нужно найти оценку снизу, сверху. Если оценки совпали, то оценка равна $\Theta(N)$. И как правило оценка сводится к наихудшему случаю
 - c) означает что нужно найти среднюю оценку для алгоритма
- 56) Какие из перечисленных ниже утверждений относятся к параметру машинное слово W в стандартной модели оперативной памяти (RAM - model)?
- a) w это количество ячеек в памяти
 - b) w это число бит в одной ячейке памяти
 - c) w это максимально допустимый размер переменной
 - d) w хранит числа ограниченной битности

е) w это число бит, необходимых для представления одной буквы или цифры

57) Какие характеристики НЕ относятся к стандартной модели оперативной памяти (RAM - model)?

- a) каждая ячейка памяти имеет динамический размер
- b) память это набор ячеек
- c) каждая ячейка это число ограниченной битности
- d) манипуляции с числами, хранящимися в ячейке, выполняются за константное время
- e) ячеек в теоретической модели памяти бесконечно много, как в машине Тьюринга

58) Что представляет собой программа для модели "разрешающие деревья"?

- a) программа на языке, похожем на Assembler, C
- b) структура в виде дерева
- c) это некоторая таблица, в которой записано, что нужно делать в зависимости от состояния

59) Какая нижняя оценка справедлива для задачи сортировки?

- a) $O(\log N)$
- b) $\Omega(N \cdot \log N)$
- c) $O(N^2)$

60) В алгоритмической модели "разрешающее дерево" в каком случае работа алгоритма завершается?

- a) если алгоритм дошел до корня
- b) если алгоритм дошел до листа
- c) если алгоритм перебрал все листья
- d) если алгоритм перебрал все ключи

61) С какого элемента начинается работа в разрешающем дереве в стандартном случае?

- a) с листьев
- b) с корня
- c) с любого возможного ключа

62) Что называется бинарным деревом?

- a) у которого ключи представлены в двоичном виде
- b) у каждой вершины которого, кроме листьев, есть ровно два сына
- c) в вершинах которого хранятся двоичные значения

63) Что называется правильным разрешающим деревом?

- a) так еще называют бинарное дерево, то есть имеющее для каждого родителя не более двух потомков
- b) которое приводит к требуемому результату, если идти по алгоритму вниз
- c) на предпоследнем уровне которого у всех родителей есть по два сына
- d) которое приводит к какому-либо результату, если идти по алгоритму вниз

64) Что называется сложностью для алгоритма, заданного разрешающим деревом?

- a) это количество всех возможных путей в дереве
- b) это высота дерева, то есть максимальная длина пути от корня дерева до вершины
- c) это количество листьев в дереве, то есть элементов на нижних уровнях

65) Как оценивается сложность правильного дерева сортировки (в худшем случае)?

- a) $T = \Omega(\log N)$
- b) $T = \Omega(N \cdot \log N)$
- c) $T = \Omega(N^2)$
- d) $T = O(N)$

66) Можно ли сортировать быстрее чем за $T = \Omega(N \cdot \log N)$, если разрешить дополнительные операции с ключами?

- a) нет
- b) да, за $T = \Omega(N \cdot \log(\log N))$, но это нереализуемо на практике
- c) да, за $T = \Omega(N \cdot \log(\log N))$ и это реализуемо на практике

67) Сколько листьев должно быть в правильном дереве для множества из N элементов?

- a) N^2
- b) $N!$
- c) $N \cdot \log N$
- d) $2N$

68) Какое из перечисленных ниже высказываний не характеризует разрешающие деревья?

- a) разрешающее дерево не является алгоритмом в общем понимании этого слова
- b) для решения алгоритмической задачи всегда строится одно разрешающее дерево
- c) модель не строит единую инструкцию для всевозможных входов в задачу

69) Почему модель алгоритма "разрешающее дерево" не очень типична для практики?

- a) дерево не всегда решает задачу корректно

- b) конкретное дерево годится для данного конкретного числа элементов
 c) задача выполняется гораздо дольше, чем на других алгоритмических моделях
- 70) Пусть имеется вектор, состоящий из битов, представляющий двоичное число. Изначально все биты равны 0. Какова сложность в худшем случае алгоритма инкрементации этого числа?
 a) $O(1)$
 b) $\Theta(N)$
 c) $O(N^2)$
 d) $O(\log N)$
- 71) Пусть имеется вектор, состоящий из битов, представляющий двоичное число. Изначально все биты равны 0. Какова сложность в худшем случае алгоритма тройной инкрементации этого числа?
 a) $O(3 + N)$
 b) $O(3N)$
 c) $O(N)$
 d) $O(3)$
- 72) Пусть имеется двоичный счетчик. Изначально все биты равны 0. Для M операций инкрементации в каком случае справедлива оценка $O(M \cdot N)$?
 a) в лучшем случае
 b) в худшем случае
 c) в среднем
- 73) По какому принципу выбирается размер *reallocation* для аддитивного метода? Если C - старый размер массива.
 a) $C' = C \cdot d$, d - константа > 1
 b) $C' = C + d$, d - число добавляемых элементов
 c) $C' = C^2/d$, d - константа > 1
- 74) По какому принципу выбирается размер *reallocation* для мультипликативного метода? Если C - старый размер массива.
 a) $C' = C \cdot d$, d - константа > 1
 b) $C' = C + d$, d - число добавляемых элементов
 c) $C' = C^2/d$, d - константа > 1
- 75) Какое время будет затрачено на выполнение последовательности из M операций для аддитивного метода увеличения размера массива?
 a) $O(M \cdot \log M)$
 b) $O(M^2)$
 c) $O(M)$
- 76) Какое время будет затрачено на выполнение последовательности из M операций для мультипликативного метода увеличения размера массива?
 a) $O(M \cdot \log M)$
 b) $O(M)$
 c) $O(M^2)$
- 77) При анализе учетных стоимостей операций $C(a_i)$ с каждым из состояний S_i связано некоторое вещественное значение ϕ_i , называемое потенциалом. Тогда чему равняется приведенная стоимость $C'(a_i)$?
 a) $C'(a_i) = C(a_i) + \phi_i - \phi_{i-1}$
 b) $C'(a_i) = C(a_i) - C(a_{i-1})$
 c) $C'(a_i) = C(a_i) + \phi_{i-1} + \phi_i$
- 78) Для задачи о бинарном поиске, какую нужно использовать функцию потенциала, чтобы получить приведенную стоимость $C'(a_i) = 2$?
 a) $\phi(S) = \#(1 \text{ in } S)$ - количество единиц в состоянии S
 b) $\phi(S) = -\#(1 \text{ in } S)$ - количество единиц в состоянии S
 c) $\phi(S) = k + 1$ - количество единиц справа до 0 в состоянии S
 d) $\phi(S) = k$ - количество единиц слева до 0 в состоянии S
- 79) Если подобрать такую функцию потенциала ϕ , что приведенная стоимость будет ограничена каким-то числом M : $C'(a_i) \leq M$. Тогда какая будет линейная оценка для суммы стоимостей?
 a) $\sum_{i=1..n} C(a_i) = \sum C'(a_i) - \phi(S_n)$
 b) $\sum_{i=1..n} C(a_i) = \sum C'(a_i) - \phi(S_1) + \phi(S_n)$
 c) $\sum_{i=1..n} C(a_i) = \sum C'(a_i) + \phi(S_n)$
- 80) Какие две операции должен выполнять хороший стек?
 a) push, get
 b) push, pop
 c) insert, get
 d) enqueue, dequeue

- 81) Какова учетная стоимость операций в стэке, реализованном с помощью вектора?
- $O(N)$
 - $Oa)$
 - $O(\log N)$
- 82) Что называется гистерезисом с точки зрения структур данных?
- если в структуре данных реализованы дополнительные свойства (поддержка минимума, максимума, сортировка)
 - если структура данных может не только увеличивать свой размер, но и уменьшать его в зависимости от заполненности
 - если в структуре данных хранятся все предыдущие ее модификации
 - если структура данных может только увеличивать свой размер, но не уменьшать
- 83) Как будет называться свойство структуры данных, для которой выполняется следующее: если коэффициент заполнения становится больше 1, тогда размер структуры увеличивается (например в 2 раза), если коэффициент заполнения падает до $1/4$ раза, тогда размер структуры уменьшается в два раза.
- амортизация
 - гистерезис
 - persistant* (версионирование)
- 84) Как эффективно реализовать стэк с поддержкой минимума?
- использовать один стэк и переменную для хранения текущего минимума, которую нужно обновлять
 - использовать два стэка: один основной для значений, второй для хранения ответов для текущего минимума
 - использовать один стэк и функцию для вычисления минимума
- 85) Как (с помощью каких структур данных) можно эффективно реализовать очередь с поддержкой минимума?
- с помощью очереди и стэка
 - с помощью двух стэков
 - очередь с дополнительной переменной
 - с помощью очереди и функции для вычисления минимума
- 86) Какая будет стоимость операций *enqueue* и *dequeue* в учетном смысле, если очередь реализована с помощью двух стэков?
- $O(N)$
 - $Oa)$
 - $O(\log N)$
 - $O(N * \log N)$
- 87) С помощью каких структур данных, перечисленных ниже, нельзя реализовать очередь?
- linked lists*
 - один стэк
 - chunked vector*
 - два стэка
- 88) Что такое циклическая очередь?
- очередь, реализованная с помощью структуры данных *linked lists*
 - очередь, в которой элементы хранятся по индексам, вычисляемым по некоторому модулю
 - очередь, реализованная с помощью структуры данных *chunked vector*
 - очередь, динамически изменяющая свой размер
- 89) Что означает свойство *persistent* (версионирование) для структуры данных?
- структура данных изменяет свои свойства в зависимости от текущей задачи
 - структура данных хранит историю своего развития и модификаций
 - структура данных динамически изменяет свой размер в зависимости от заполненности
- 90) Какая существует главная проблема, мешающая реализации *immutable* очереди с помощью двух стэков?
- большой объем памяти для хранения структуры
 - трудность анализа учетных стоимостей
 - сложность реализации
 - низкая производительность
- 91) Какое время выполнения операции *Push* у *persistent* стэка? Если N - длина стэка
- $O(\log N)$
 - $Oa)$
 - $O(N)$
- 92) Существует подход для освобождения памяти для *persistent stack*, называемый подсчет ссылок (*ref-counting*). Как его можно описать?
- для каждой вершины (узла) мы храним указатели на все ссылающиеся вершины
 - для каждой вершины (узла) мы помним сколько стрелок на нее ссылается (число)

- с) помечаются все элементы, достижимые из корней
- 93) Выберите, чем характеризуется подход для освобождения памяти для *persistent stack*, называемый подсчет ссылок (*ref-counting*)?
- он корректен
 - необходимо хранить счетчик на каждую вершину
 - структура при этом не является неизменяемой (*immutable*)
 - помечаются все элементы, достижимые из корней
 - структура эффективна в многопоточном режиме
- 94) В чем особенности очереди?
- открыта с обеих сторон;
 - открыта с одной стороны на вставку и удаление;
 - доступен любой элемент.
- 95) Какую дисциплину обслуживания принято называть FIFO?
- стек;
 - очередь;
 - дек.
- 96) Какую дисциплину обслуживания принято называть LIFO?
- стек;
 - очередь;
 - дек.
- 97) При удалении элемента из кольцевого списка...
- список разрывается;
 - в списке образуется дыра;
 - список становится короче на один элемент.
- 98) Для чего используется указатель в кольцевых списках?
- для ссылки на следующий элемент;
 - для запоминания номера сегмента расположения элемента;
 - для ссылки на предыдущий элемент;
 - для расположения элемента в списке памяти.
- 99) Чем отличается кольцевой список от линейного?
- в кольцевом списке последний элемент является одновременно и первым;
 - в кольцевом списке указатель последнего элемента пустой;
 - в кольцевых списках последнего элемента нет;
 - в кольцевом списке указатель последнего элемента не пустой.
- 100) В каких направлениях можно перемещаться в кольцевом двунаправленном списке?
- в обоих;
 - влево;
 - вправо.

5.3. Оценочные материалы, используемые для экзамена (описание билетов, тестов и т.п.)

Экзамен не предусмотрен

5.4. Методика оценки освоения дисциплины

№ п/п	Вид оценочного средства	Критерий	Оценка
1	Выполнение и защита домашних заданий	Все задачи домашнего задания выполнены без ошибок, либо с не принципиальными ошибками, не влияющими на конечный результат (или оказывающие минимальное влияние)	«Зачтено»
		Задание не выполнено полностью, либо выполнены не все задачи, либо в решении допущены существенные ошибки, не исправленные после собеседования с преподавателем	«Не зачтено»
2	Устный опрос по материалам лабораторных и	Обучающийся уверенно и правильно отвечает на вопросы	«Зачтено»
		Обучающийся не верно отвечает на вопросы	«Не зачтено»

		практических занятий			
	3	Оценка тестовых заданий	Обучающийся дал 90% и более правильных ответов	«Отлично»	
			Обучающийся дал 65-89% правильных ответов	«Хорошо»	
			Обучающийся дал 50-64% правильных ответов	«Удовлетворительно»	
			Обучающийся дал менее 50% правильных ответов	«Неудовлетворительно»	

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ				
6.1. Рекомендуемая литература				
6.1.1. Основная литература				
Обозначение	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л 1.1	Круз Р.Л.	Структуры данных и проектирование программ: учебное пособие	ЭБС «Университетская библиотека онлайн» URL: https://biblioclub.ru/index.php?page=book_red&id=446128	Москва: Лаборатория знаний, 2021
Л 1.2	Мунтян Е. Р. Сергеев Н. Е.	Математические и алгоритмические основы построения интеллектуальных систем: учебное пособие	ЭБС «Университетская библиотека онлайн» URL: https://biblioclub.ru/index.php?page=book_red&id=698713	Ростов-на-Дону, Таганрог: Южный федеральный университет, 2022
6.1.2. Дополнительная литература				
Обозначение	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л 2.1	Вялый М., Подольский В., Рубцов А., Шварц Д., Шень А.	Лекции по дискретной математике: учебник	ЭБС «Университетская библиотека онлайн» URL: https://biblioclub.ru/index.php?page=book_red&id=615644	Москва: Издательский дом Высшей школы экономики, 2021
6.1.3. Методические разработки				
Обозначение	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л 3.1	Жуков П.И.	Методические указания по выполнению практических работ		
Л 3.2	Жуков П.И.	Методическое указание по выполнению домашних заданий		
6.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»				
Э 1	Открытое образование [Электронный ресурс]: https://openedu.ru/			
Э 2	LMS Moodle https://open.misis.ru/			
Э 3	ЭБС «Университетская библиотека онлайн» [Электронный ресурс]: https://biblioclub.ru			
6.3. Перечень программного обеспечения				
П 1	Microsoft Windows			
П 2	Microsoft Office			
П 3	Microsoft Visual Studio 2022			
6.4. Перечень информационных справочных систем и профессиональных баз данных				
И 1	eLIBRARY.RU - НАУЧНАЯ ЭЛЕКТРОННАЯ БИБЛИОТЕКА [Электронный ресурс]: https://elibrary.ru/			

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ	
7.1	Аудитория №406 «Лаборатория прикладного программирования» Перечень основного оборудования, учебно-наглядных пособий: • монитор - 9шт.; • персональный компьютер - 9шт.; • проектор; • экран настенный; • усилитель-распределитель; • комплект учебной мебели на 25 посадочных мест. В помещении имеется подключение к сети «Интернет» и доступ в электронную информационно-образовательную среду организации.
7.2	Аудитория №416 «Лаборатория искусственного интеллекта» Перечень основного оборудования, учебно-наглядных пособий: • Персональный компьютер ASUS H81M с монитором- 9 шт. • Экран ELV240 / ACP-DINON-ELV240 / 180x240 MW 3:4 - 1шт • Мультимедиа проектор Epson EB-485W - 1шт • Комплект учебной мебели на 25 посадочных мест. • Лабораторный стенд «Исследование сервоприводов» - 1 шт., ввод в эксплуатацию: 2011 год. В помещении имеется подключение к сети «Интернет» и доступ в электронную информационно-образовательную среду организации.

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ИЗУЧЕНИЮ ДИСЦИПЛИНЫ
Для успешного освоения дисциплины обучающемуся необходимо: • Посещать все виды занятий. • Своевременно регистрироваться на рекомендованные электронные ресурсы. • При возникновении любых вопросов по содержанию дисциплины и организации работы своевременно обращаться к преподавателю (в часы очных консультаций, через LMS Moodle). • Отчеты по практическим и лабораторным работам рекомендуется выполнять с использованием MS Office. • Активно работать с научными базами в сети Интернет. • Качественное освоение дисциплины возможно только при систематической самостоятельной работе, что поддерживается системой текущей аттестации. КОНТРОЛЬ КАЧЕСТВА ОСВОЕНИЯ ДИСЦИПЛИНЫ Промежуточная аттестация по дисциплине предусмотрена в виде зачета в первом семестре. Обучение проводится в один семестр и организуется в соответствии с настоящей программой. Самостоятельная работа студентов осуществляется и контролируется с помощью: – индивидуального опроса студентов при защите практических работ и лабораторных работ; – сдачи домашних заданий; – выполнения тестов. Оценочные средства позволяют оценить компетенции (части компетенций), сформированные у обучающихся в процессе освоения и по завершению изучения дисциплины. По результатам выполнения ДЗ обучающиеся оформляют отчеты (по ГОСТ 7.32–2017 Отчет о научно-исследовательской работе).