Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 06.07.2026 15:38:44
Уникальный программный ключ:
8db180d1a3f02ac9e60521a5672742735c18b1d6

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

(МОСКОВСКИЙ ПОЛИТЕХ)

Факультет информационных технологий

УТВЕРЖДАЮ
Декан факультета
«Информационных технологий»
_____ / Д.Г. Демидов /
«26» _____февраля 2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Алгоритмы и структуры данных»

Направление подготовки/специальность
09.03.03 Прикладная информатика

Профиль/специализация
«Разработка и интеграция бизнес-приложений»

Квалификация
бакалавр

Формы обучения
очная

Год начала обучения – 2026 г.

Москва, 2026 г.

Разработчик(и):

доцент  / А.Н. Варнавский /

Согласовано:
Заведующий кафедрой «Инфокогнитивные технологии»,
к.т.н., доцент  / Е.А. Пухова /

Руководитель образовательной программы

ст преподаватель  / М.В. Даньшина /

Содержание

1 Цели, задачи и планируемые результаты обучения по дисциплине 4
2 Место дисциплины в структуре образовательной программы 4
3 Структура и содержание дисциплины 4
3.1 Виды учебной работы и трудоемкость для очной формы обучения 5
3.2 Тематический план изучения дисциплины для очной формы обучения 5
3.3 Содержание дисциплины 6
3.4 Тематика лабораторных занятий 6
4 Учебно-методическое и информационное обеспечение 6
4.1 Нормативные документы и ГОСТы 6
4.2 Основная литература 7
4.3 Дополнительная литература 7
4.4 Электронные образовательные ресурсы 7
4.5 Лицензионное и свободно распространяемое программное обеспечение 7
4.6 Современные профессиональные базы данных и информационные справочные системы 7
5 Материально-техническое обеспечение 7
6 Методические рекомендации 8
6.1 Методические рекомендации для преподавателя по организации обучения 8
6.2 Методические рекомендации для обучающихся по освоению дисциплины 8
7 Фонд оценочных средств 8
7.1 Методы контроля и оценивания результатов обучения 8
7.2 Шкала и критерии оценивания результатов обучения 9
7.3 Оценочные средства 10

1 Цели, задачи и планируемые результаты обучения по дисциплине

Цель формирование знаний и умений в области методов представления данных в памяти компьютера, основных алгоритмов, оперирующих с ними, в том числе с использованием современных интеллектуальных технологий, а также освоение методов решения задач и создание теоретической основы для изучения ряда специальных дисциплин.

Задачи дисциплины ознакомление с теорией о структурах данных и базовых алгоритмах; определение понятия временной сложности и сложности по памяти для алгоритмов; определение требований к эффективному применению структур и алгоритмов; определение основных фундаментальных и абстрактных структур

Обучение по дисциплине «Алгоритмы и структуры данных» направлено на формирование у обучающихся следующих компетенций:

Код и наименование компетенций	Индикаторы достижения компетенции
ОПК-7. Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ИОПК-7.1. Знает основные языки программирования, операционные системы и оболочки, современные среды разработки программного обеспечения. ИОПК-7.2. Умеет составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы, интегрировать программные модули. ИОПК-7.3. Владеет языком программирования, методами отладки и тестирования работоспособности программы
ПК-4**.** Способен проводить работы по интеграции программных модулей и компонент и проверку работоспособности выпусков программных продуктов	ИПК-4.1. Знает: методы описания алгоритмов, основные абстрактные типы данных и их реализации; ИПК-4.2. Умеет: разрабатывать алгоритмы для конкретных задач, учитывая различия в представлении информационных объектов современными браузерами.

2 Место дисциплины в структуре образовательной программы

Дисциплина относится к части, формируемой участниками образовательных отношений блока Б1.2 и междисциплинарно связана с поддерживающими дисциплинами: основы программирования и последующими дисциплинами: Проектирование баз данных, Шаблоны проектирования приложений, Математические методы анализа данных.

3 Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 4 зачетные единицы (144 часа).

3.1 Виды учебной работы и трудоемкость для очной формы обучения

№ п/п	Вид учебной работы	Количество часов	Семестры	
			1 семестр	
1	**Аудиторные занятия**	**48**	48	
	В том числе:			
1.1	Лекции	16	16	
1.2	Семинарские/практические занятия	-	-	
1.3	Лабораторные занятия	32	32	
2	**Самостоятельная работа**	**96**	96	
3	**Промежуточная аттестация**			
	Зачет/диф.зачет/экзамен		экзамен	
	Итого:	**144**	144	

3.2 Тематический план изучения дисциплины для очной формы обучения

№ п/п	Разделы/темы дисциплины	Трудоемкость, час					
		Всего	Аудиторная работа				Самостоятельная работа
			Лекции	Семинарские/ практические занятия	Лабораторные занятия	Практическая подготовка	
1	Языки программирования, алгоритмы, структуры данных	13	1		2		10
2	Введение в анализ сложности алгоритмов	13	1		2		10
3	Поиск и сортировка	13	1		2		10
4	Структуры данных: массивы, стеки, очереди, списки	13	1		2		10
5	Анализ сложности алгоритмов	14	2		2		10
6	Асимптотическая оценка эффективности операций с массивом	16	2		4		10
7	Алгоритмы поиска	15	2		4		9
8	Алгоритмы сортировки	15	2		4		9
9	Программная реализация структур данных	15	2		4		9
10	Некоторые алгоритмы	17	2		6		9
Итого		**144**	**16**		**32**		**96**

3.3 Содержание дисциплины

Языки программирования, алгоритмы, структуры данных. Рассматриваемы вопросы: Языки программирования. Алгоритмы. Итеративные и рекурсивные алгоритмы. Структуры данных.

Введение в анализ сложности алгоритмов. Рассматриваемы вопросы: Сложности экспериментального анализа эффективности алгоритма. Базовые операции. Подсчет базовых операций. Асимптотическая оценка сложности алгоритма. Асимптотическая оценка эффективности операций с массивом, списком, кортежем.

Поиск и сортировка. Рассматриваемы вопросы: Поиск. Последовательный поиск в неотсортированном массиве. Последовательный поиск в отсортированном массиве. Бинарный поиск. Сортировки. Обменные сортировки. Сортировка выбором. Сортировка вставками. Сортировка Шелла. Быстрая сортировка. Сортировка слиянием. Сравнение сортировок.

Структуры данных: массивы, стеки, очереди, списки. Рассматриваемы вопросы: Массивы, списки, кортежи. Библиотеки, функции, методы для работы с массивом, списками, кортежами. Статические и динамические массивы. Стеки. Программная реализация стека. Очереди. Программная реализация очереди. Связные списки. Однонаправленные связные списки. Циклические связные списки. Двунаправленные связные списки.

3.4 Тематика лабораторных занятий

Языки программирования, алгоритмы, структуры данных
Введение в анализ сложности алгоритмов
Поиск и сортировка
Структуры данных: массивы, стеки, очереди, списки
Анализ сложности алгоритмов
Асимптотическая оценка эффективности операций с массивом
Алгоритмы поиска
Алгоритмы сортировки
Программная реализация структур данных
Некоторые алгоритмы

4 Учебно-методическое и информационное обеспечение

4.1 Нормативные документы и ГОСТы

1. Федеральный закон от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации» (с изменениями и дополнениями);

2. Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.03 Прикладная информатика, утвержденный приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 922.

3. Приказ Министерства образования и науки РФ от 05 апреля 2017 г. № 301 «Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры;

4. Порядок проведения государственной итоговой аттестации по образовательным программам высшего образования – программам бакалавриата, программам специалитета и программам магистратуры, утвержденный приказом Минобрнауки России от 29 июня 2015 г. № 636;

5. Положение о практической подготовке обучающихся, утвержденное приказом Министерства науки и высшего образования Российской Федерации и Министерства просвещения Российской Федерации от 5 августа 2020 г. № 885/390.

4.2 Основная литература

Черпаков, И. В. Алгоритмизация и программирование на Python : учебник для вузов / И. В. Черпаков. — Москва : Издательство Юрайт, 2026. — 159 с. — (Высшее образование). — ISBN 978-5-534-21910-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/582412 (дата обращения: 11.02.2026).

4.3 Дополнительная литература

Павлов, Л. А. Структуры и алгоритмы обработки данных: учебник для вузов / Л. А. Павлов, Н. В. Первова. — 3-е изд., стер. — Санкт-Петербург: Лань, 2021. — 256 с. — ISBN 978-5-8114-7259-8. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/156929 (дата обращения: 11.02.2026). — Режим доступа: для авториз. пользователей.

Сыромятников, В. П. Структуры и алгоритмы обработки данных: Практикум: учебное пособие / В. П. Сыромятников. — Москва: РТУ МИРЭА, 2020. — 244 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/163915 (дата обращения: 11.02.2026). — Режим доступа: для авториз. пользователей.

4.4 Электронные образовательные ресурсы

1. ЭОР курс в lms Алгоритмы и структуры данных
https://online.mospolytech.ru/course/view.php?id=13265
2. ЭБС IPR SMART (https://www.iprbookshop.ru/)
3. Образовательная платформа Юрайт. Для вузов и ссузов (https://urait.ru/)
4. ЭБС Лань (https://e.lanbook.com/)

4.5 Лицензионное и свободно распространяемое программное обеспечение

1. Visual Studio Code
2. Google Colab
3. Браузеры Chrome, Edge, Firefox
4. OpenVPN с правами для запуска у студентов
5. FileZilla
6. PuTTy
7. Git
8. Node.js 18
9. Python 3.10
10. Wireshark

4.6 Современные профессиональные базы данных и информационные справочные системы

https://habr.com/ru/articles/

5 Материально-техническое обеспечение

Для проведения лабораторных работ и самостоятельной работы студентов подходят аудитории, оснащенные компьютерами с программным обеспечением в соответствии со списком в пункте 4.5 и подключенные к интернету.

Число рабочих мест в аудитории должно быть достаточным для обеспечения индивидуальной работы студентов.

Рабочее место преподавателя должно быть оснащено компьютером с подключенным к нему проектором или иным аналогичным по функциональному назначению оборудованием.

6 Методические рекомендации

6.1 Методические рекомендации для преподавателя по организации обучения

1. При подготовке к занятиям следует предварительно проработать материал занятия, предусмотрев его подачу точно в отведенное для этого время занятия. Следует подготовить необходимые материалы – теоретические сведения, задачи и др. При проведении занятия следует контролировать подачу материала и решение заданий с учетом учебного времени, отведенного для занятия.

2. При проверке работ и отчетов следует учитывать не только правильность выполнения заданий, но и оптимальность выбранных методов решения, правильность выполнения всех его шагов.

6.2 Методические рекомендации для обучающихся по освоению дисциплины

Изучение дисциплины осуществляется в строгом соответствии с целевой установкой в тесной взаимосвязи с учебным планом. Основой теоретической подготовки студентов являются лекции и самостоятельная работа.

В процессе самостоятельной работы студенты закрепляют и углубляют знания, полученные во время аудиторных занятий, готовятся к промежуточной аттестации, а также самостоятельно изучают отдельные темы учебной программы.

На занятиях студентов, в том числе предполагающих практическую деятельность, осуществляется закрепление полученных, в том числе и в процессе самостоятельной работы, знаний. Особое внимание обращается на развитие умений и навыков установления связи положений теории с профессиональной деятельностью будущего специалиста.

Самостоятельная работа осуществляется индивидуально. Контроль самостоятельной работы организуется в двух формах:

- самоконтроль и самооценка студента;
- контроль со стороны преподавателей (текущий и промежуточный).

Текущий контроль осуществляется на аудиторных занятиях.

Критериями оценки результатов самостоятельной работы студента являются:

- уровень освоения студентом учебного материала;
- умения студента использовать теоретические знания при выполнении практических задач;
- сформированность компетенций;
- оформление материала в соответствии с требованиями.

Приветствуется обсуждение самих заданий с другими студентами: можно как давать, так и получать советы по общей стратегии выполнения и изучения материала, давать и получать помощь в отладке. Однако писать код студент должен самостоятельно. Делиться кодом или писать его совместно запрещено.

7 Фонд оценочных средств

7.1 Методы контроля и оценивания результатов обучения

Приведенные ниже правила выставления оценок и опозданий могут быть изменены, если преподаватель сочтет это необходимым. Важно, чтобы студенты регулярно просматривали план курса, выложенный в СДО, на предмет его обновления или изменения.

Достижение компетенций оценивается с помощью лабораторных работ и рубежных контролей. Индикаторы ИОПК-7.1, ИОПК-7.2, ИОПК-7.3, ИПК-4.1, ИПК-4.2 заложены в темах 1-12.

В соответствии с планом на дисциплины студентам выдаются задания на лабораторные работы. Помимо требований и описания функционала в работе указан крайний срок сдачи. Для сдачи лабораторной работы студенту необходимо прислать ссылку на репозиторий в GitHub Classroom и на хостинг, где размещен результат работы с реализованным функционалом, описанным в задании. Работа считается сданной если в ней реализовано 80% и более требований и функционала, описанного в задании.

Каждый студент имеет право на 6 дней опоздания, которые могут быть потрачены на любые задания в течение семестра. Опоздания предназначены для решения особых ситуаций, таких как болезнь или чрезвычайные семейные обстоятельства.

Когда использованы все дни опоздания за каждый день просрочки начисляется штраф в размере 25% от максимального результата за задание. Задания, присланные позже, чем 4 дня, не будут оцениваться. В связи с зависимостью между работами студентам может потребоваться все равно выполнить предыдущие работы, даже если они не оцениваются.

После сдачи лабораторной работы студент должен ее защитить. Во время защиты лабораторной работы преподаватель проверяет репозиторий, хостинг и выполнение критериев и требований задания, а студент отвечает на вопросы преподавателя по его коду, а также теоретических вопросов, приведенных после текста задания лабораторной работы. Если студент отказывается отвечать на вопросы, или дает полностью неверные ответы, или ответы не по теме, то работа может считаться сданной, но при этом она не оценивается.

Работа должна быть выполнена студентом самостоятельно: в репозитории в системе контроля версий студента содержатся коммиты только за его авторством, по этим коммитам можно проследить как велась работа, студент может объяснить свой код и ход выполнения работы, если эти правила не соблюдаются, то работа не считается сданной и не оценивается.

Рубежные контроли пишутся в аудитории индивидуально по варианту задания, выданному преподавателем в назначенные дни. При отсутствии студента в день написания контрольной работы ему дается еще один шанс ее написать на последнем занятии в семестре, но обязательно очно.

Студенты должны заранее сообщать о том, что у них могут возникнуть трудности со своевременной сдачей задания или проекта. При наличии реальных причин задержки студентам следует как можно скорее связаться с преподавателем и обсудить возможные условия.

7.2 Шкала и критерии оценивания результатов обучения

Итоговый балл Rитог состоит из 5-и составляющих и рассчитывается по формуле:

$$R_{итог} = R_{контрольные} + R_{лабораторные} + R_{задание1} + R_{задание2} + R_{итог.тест}.$$

Rконтрольные – баллы, полученный за выполнение контрольных работ;

Rлабораторные – баллы, полученные за работу на лабораторных занятиях и выполнение лабораторных работ;

Rзадание1 – балл, полученный за самостоятельное выполнение задания1 по работе с закодированными строчками программного кода;

Rзадание2 - балл, полученный за выполнение задания2 по работе с закодированными строчками программного кода на последнем лабораторном занятии;

Rитог.тест – балл за итоговый тест.

За контрольные работы можно набрать до 45 баллов, за лабораторные занятия – до 10 баллов, за задание1 – до 10 баллов, за задание2 – до 10 баллов, за итоговый тест – до 15 баллов. Таким образом, максимально возможный балл составляет 90.

Контрольные и лабораторные работы оцениваются в процентах степени выполнения следующих критериев и для выставления оценки суммируются проценты за каждый из четырех критериев:

1. Полнота выполнения практического задания (30%): соответствует ли функциональность заданным требованиям и целям, насколько точно и без ошибок код выполняет поставленные задачи, насколько эффективно задание отвечает требованиям целевой аудитории и обеспечивает приятное восприятие.

2. Качество и структура кода (10%): качество, читаемость и организация кода, рациональность выполнения задания, последовательность именования и соблюдение лучших практик.

3. Творчество и инновации (10%): творческий подход студентов к выполнению заданий, насколько студенты вышли за рамки основных требований и реализовали дополнительные возможности или использовали уникальные решения.

4. Ответы на вопросы по коду студента и теории (50%):

Дает краткий ответ, содержащий ошибки или неточности. На наводящие вопросы отвечает неправильно (10% из 50%).

Дает развернутый ответ, содержащий ошибки или неточности. На наводящие вопросы отвечает неверно (20% из 50%).

Дает развернутый ответ, содержащий ошибки или неточности. На наводящие вопросы отвечает правильно (30% из 50%).

Дает правильные и развернутые ответы на вопросы (50% из 50%).

Rлабораторные рассчитывается как сумма результатов за все лабораторные работы, отмасштабированная к 45 баллам. Rконтрольные рассчитывается как сумма результатов за все лабораторные работы, отмасштабированная к 10 баллам.

За задание1 и задание2 выставляется оценка 10 баллов в случае раскодирования всех полученных студентом строчек программного кода. В противном случае балл за задания снижается пропорционально числу не раскодированных строчек кода.

Итоговый тест выполняется в ЛМС, проверка ответов осуществляется автоматически.

Итоговый балл пересчитывается по шкале ниже и на основании полученной оценки фиксируется результат промежуточной аттестации.

Соответствие набранных баллов Rитог оценке по 4-бальной шкале:

0-44 - неудовлетворительно

45-59 - удовлетворительно

60-74 - хорошо

75-90 – отлично

7.3 Оценочные средства

7.3.1 Текущий контроль

Примеры вопросов для устного опроса

К теме 1. Структуры данных. 1. Классификация структур данных. 2. Операции над структурами данных. 3. Типы данных линейной структуры. 4. Линейные структуры данных с прямым и последовательным доступом. 5. Стеки, очереди, очереди с приоритетом, деки, связанные списки. 6. Односвязный линейный список, циклические списки, двусвязный линейный список. 7. Мультисписки.

К теме 2. Алгоритмы обработки данных. 1. Алгоритмы обработки данных линейной структуры – сортировка. 2. Сортировка выбором. 3. Сортировка обменом (пузырек). 4. Сортировка вставками. 5. Сортировка слиянием. 6. Анализ сложности алгоритмов. 7. Сортировка Шелла. 8. Быстрая сортировка. 9. Пирамидальная сортировка. 10.Анализ сложности алгоритмов. 11.Алгоритмы обработки данных линейной структуры – поиск. 12.Методы поиска. 13.Последовательный поиск. 14.Бинарный поиск. 15.Интерполирующий поиск. 16.Фибоначчиев поиск.

7.3.2 Промежуточная аттестация

Вопросы к экзамену

1. Процесс создания компьютерной программы.
2. Псевдоязык и пошаговая «кристаллизация» алгоритмов.
3. Эвристические алгоритмы.
4. Понятия типа данных, структуры данных, абстрактного типа данных.
5. Понятие указателя.
6. Понятие курсора.
7. Массивы.
8. Записи.
9. Файлы.
10. Понятие абстрактного типа данных список.
11. Связные списки.
12. Дважды связные списки.
13. Понятие стека.
14. Понятие очереди.
15. Рекурсии и рекурсивные процедуры.
16. Дерево как иерархическая структура узлов.
17. Неупорядоченные и упорядоченные деревья.
18. Способы обхода узлов дерева.
19. Помеченные деревья и деревья выражений.
20. Вычисление «наследственных» данных.
21. Коды Хаффмана.
22. Создание дерева Хаффмана.
23. Алгоритм Хаффмана и его реализация.
24. Линейно упорядоченное множество.
25. Основные операции, выполняемые над множествами.
26. Операторы абстрактных типов данных, основанных на множествах.
27. Отображения множеств.
28. Словарь как абстрактный типа множества.
29. Хеширование как метод реализации словарей.
30. Открытое хеширование.
31. Закрытое хеширование.
32. Оценка эффективности хеш-функций.
33. Реструктуризация хеш-таблиц.
34. Очередь с приоритетами как модель множества.
35. Реализация очереди с приоритетами посредством частично упорядоченных деревьев.
36. Сложные множества.
37. Отношения «многие-ко-многим».
38. Структуры мультисписков.
39. Деревья двоичного поиска.
40. Понятия ориентированного и неориентированного графов.
41. Матрица смежности.
42. Абстрактные типы данных для ориентированных графов.
43. Задача нахождения кратчайшего пути с одним источником.
44. «Жадный» алгоритм Дейкстры.

45.Нахождение кратчайших путей между парами вершин.
46.Алгоритм Флойда и его реализация.
47.Нахождение центра ориентированного графа.
48.Поиск ориентированного графа в глубину.
49.Ориентированный ациклический граф.
50.Топологическая сортировка.
51.Остовные деревья минимальной стоимости.
52.Алгоритм Прима.
53.Алгоритм Крускала.
54.Обход неориентированного графа методом поиска в глубину.
55.Обход неориентированного графа методом поиска в ширину.
56.Последовательный поиск в массивах.
57.Последовательный поиск в связных списках.
58.Бинарный поиск в массивах.
59.Бинарный поиск в связных списках.
60.Задача сортировки.
61.Сортировка методом «пузырька».
62.Сортировка вставками.
63.Сортировка посредством выбора.
64.Временная сложность методов сортировки.
65.Процедура быстрой сортировки.
66.Временная сложность быстрой сортировки.
67.Абстрактный алгоритм пирамидальной сортировки.
68.Двухэтапная «карманная» сортировка.
69.Общая поразрядная сортировка.
70.Сортировка Шелла.
71.Определение эффективности алгоритма.
72.Подходы к решению рекуррентных соотношений.
73.Общее решение рекуррентных соотношений.
74.Однородные и частные решения рекуррентных уравнений.
75.Метод декомпозиции при разработке алгоритмов.
76.Таблицы решений в динамическом программировании.
77.Задача триангуляции многоугольника.
78.«Жадные» алгоритмы как эвристики.
79.Задача коммивояжера и гамильтонов цикл.
80.Метод поиска с возвратом.
81.Метод альфа-бета отсечения.
82.Ограничения эвристических алгоритмов.
83.Эвристический поиск.
84.Поиск на игровых деревьях (графах).