

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
Федеральное государственное бюджетное образовательное учреждение высшего
образования
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Факультет математики и компьютерных наук

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Алгоритмы и структуры данных

Кафедра дискретной математики и информатики
факультета математики и компьютерных наук

Образовательная программа бакалавриата

01.03.05 Статистика

Направленность (профиль) программы
Анализ больших данных

Форма обучения

очная

Статус дисциплины:
входит в обязательную часть

Махачкала, 2023

Рабочая программа дисциплины «Алгоритмы и структуры данных» составлена в 2023 году в соответствии с требованиями ФГОС ВО по направлению подготовки 01.03.05 Статистика (уровень бакалавриата) от «14» августа 2020 г. № 1032.

Разработчик(и): кафедра дискретной математики и информатики, Раджабова Наима Шамильевна, к.ф.-м.н., доцент.

Рабочая программа дисциплины одобрена:
на заседании кафедры дискретной математики и информатики от «18»
сентября, протокол № 5 ;
зав. кафедрой AM Магомедов А. М.
(подпись)

и
на заседании Методической комиссии факультета математики и компьютерных наук
от
«25» сентября, протокол № 4 ;
председатель AM Ризаев М. К.
(подпись)

Рабочая программа дисциплины согласована с учебно-методическим управлением
«20» 02. 2023 .
/Начальник УМУ AG Гасангаджиева А. Г.
(подпись)

Аннотация рабочей программы дисциплины

Дисциплина “Алгоритмы и структуры данных” входит в обязательную часть образовательной программы бакалавриата по направлению 01.03.05 Статистика.

Дисциплина реализуется на факультете математики и компьютерных наук кафедрой дискретной математики и информатики.

Содержание дисциплины охватывает круг вопросов, связанных с современными знаниями в области основных применяемых в программировании структур данных, алгоритмов обработки данных и анализа этих алгоритмов, взаимосвязи алгоритмов и структур данных, реализации алгоритмов на языке С# (и среды MS Visual Studio).

Дисциплина нацелена на формирование следующих компетенций выпускника: профессиональной – ПК-5.

Преподавание дисциплины предусматривает проведение следующих видов учебных занятий: лекции и лабораторные занятия.

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости: текущего контроля в форме 4 контрольных работ и промежуточного – в форме зачета.

Объем дисциплины составляет 5 зачетных единиц (180 часов), в том числе в академических часах по видам учебных занятий:

Объем дисциплины в очной форме

Семестр	Учебные занятия						СРС, в том числе экзамен	Форма промежуточной аттестации (зачет, дифференцированный зачет, экзамен)	
	в том числе								
	Контактная работа обучающихся с преподавателем								
Всего	из них								
	Лекции	Лабораторные занятия	Практические занятия	КСР	Консультации				
1	180	16	16				112+36	экзамен	

1. Цели освоения дисциплины

Целью освоения дисциплины (модуля) «Алгоритмы и структуры данных» является получение фундаментальных знаний и практических навыков в области теории алгоритмов, современных структур данных и их реализации на языке программирования С# для построения математических моделей дискретных структур и разработки программного обеспечения.

Задачей изучения дисциплины является создание и развитие практических навыков использования основных структур данных в программировании, алгоритмов обработки данных и анализа этих алгоритмов, взаимосвязи алгоритмов и структур данных.

2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина «Алгоритмы и структуры данных» относится к обязательной части образовательной программы бакалавриата по направлению 01.03.05 Статистика.

Дисциплина «Алгоритмы и структуры данных» призвана содействовать знакомству студентов с языками и методами программирования и является курсом, для освоения которого необходимы теоретические знания и практические навыки, полученные по дисциплинам «Основы информатики», «Языки и методы программирования», «Дискретная математика».

Результаты освоения данной дисциплины будут востребованы на занятиях по «Объектно-ориентированным языкам и системам программирования» и при подготовке выпускной квалификационной работы.

3. Компетенции обучающегося, формируемые в результате освоения дисциплины (перечень планируемых результатов обучения).

Код компетенции из ФГОС ВО	Наименование компетенции из ФГОС ВО	Планируемые результаты обучения	Процедура освоения
ПК-5. Способе разрабатывать реализовывать в виде программного модуль алгоритм решения поставленной теоретической или прикладной задачи.	ПК-5.1. Знает разрабатывать и реализовывать в виде программного модуль алгоритм решения поставленной теоретической или прикладной задачи.	Знает: разрабатывать и реализовывать в виде программного модуль решения поставленной теоретической или прикладной задачи. Умеет: разрабатывать и реализовывать в виде программного модуль решения поставленной теоретической или	Конспектирование лекций, выполнение лабораторных работ, подготовка и представление реферата

	прикладной задачи.. Владеет: навыками разработки и реализации в виде программного модуля алгоритм решения поставленной теоретической или прикладной задачи.
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<p>ПК-5.2. Умеет разрабатывать и реализовывать в виде программного модуля алгоритм решения поставленной теоретической или прикладной задачи.</p>	<p>Знает: разрабатывать и реализовывать в виде программного модуля алгоритм решения поставленной теоретической или прикладной задачи. Умеет: разрабатывать и реализовывать в виде программного модуля алгоритм решения поставленной теоретической или прикладной задачи.. Владеет: навыками разработки и реализации в виде программного модуля алгоритм решения поставленной теоретической или прикладной задачи.</p>
--------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

	ПК-5.3. Владеет навыками разрабатывать и реализовывать в виде программного модуля алгоритм решения поставленной теоретической или прикладной задачи	Знает: разрабатывать и реализовывать в виде программного модуля алгоритм решения поставленной теоретической или прикладной задачи. Умеет: разрабатывать и реализовывать в виде программного модуля алгоритм решения поставленной теоретической или прикладной задачи. Владеет: навыками разработки и реализации в виде программного модуля алгоритм решения поставленной теоретической или прикладной задачи.	
--	-----------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

4. Объем, структура и содержание дисциплины.

4.1 Объем дисциплины составляет 5 зачетных единиц, 180 академических часов.

4.2 Структура дисциплины

Структура дисциплины в очной форме

№ п/п	Разделы и темы дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)
				Лекции	Практические занятия	Лабораторные занятия	Самостоятельная работа	
Модуль 1. Основные понятия теории алгоритмов и структур данных								
1	Алгоритм и его свойства.	1	1	2		2	14	Прием лабораторных работ, реферат

2	Структуры данных	1	2	2		2	14	Прием лабораторных работ, реферат
	Итого по модулю 1		36	4		4	28	
Модуль 2. Элементы теории сложности алгоритмов								
3	Анализ алгоритмов	1	3	2		2	14	Прием лабораторных работ
4	Алгоритмы сортировки массивов	1	4	2		2	14	Прием лабораторных работ
	Итого по модулю 2:		36	4		4	28	Контрольная работа
Модуль 3. Графовые алгоритмы и структуры данных								
5	Алгоритмы для работы с графами	1	5	2		2	14	Прием лабораторных работ
6	Кратчайшие пути.	1	6	2		2	14	Прием лабораторных работ, реферат
	Итого по модулю 3:		36	4		4	28	Контрольная работа
Модуль 4. Жадные алгоритмы и динамическое программирование								
7	Жадные алгоритмы	1	7	2		2	14	Прием лабораторных работ
8	Динамическое программирование	1	8	2		2	14	Прием лабораторных работ, реферат
	Итого по модулю 4:		36	4		4	28	Контрольная работа
	Подготовка к экзамену							36
	ИТОГО:		180	16		16	112+36	

4.3 Содержание дисциплины, структурированное по разделам и темам

4.3.1 Содержание лекционных занятий по дисциплине

Модуль 1. Основные понятия теории алгоритмов и структур данных

Тема 1. Алгоритм и его свойства. Определение и свойства алгоритма. Алгоритм как модель. Способы записи алгоритма. Основные алгоритмические конструкции: следование, ветвление, цикл. Запись основных конструкций на языке Python. Оценка сложности алгоритма. Принципы создания эффективных алгоритмов.

Тема 2. Структуры данных. Понятие структуры данных. Физический и логический уровни представления структур данных. Простые и интегрированные структуры (типы) данных. Классификация структур данных по признаку изменчивости. Линейные и нелинейные структуры данных. Структуры данных (матрица, вектор, множество). Строчные структуры данных (стек, очередь, дек). Списки как структуры данных. Нелинейные структуры данных (деревья, графы, многосвязные списки).

Модуль 2. Элементы теории сложности алгоритмов

Тема 1. Модель вычислений RAM. Анализ сложности наилучшего, наихудшего и среднего случая. Асимптотические обозначения. Скорость роста и отношения доминирования. Работа с асимптотическими обозначениями. Оценка эффективности алгоритмов. Классы P и NP.

Тема 2. Алгоритмы сортировки массивов

Алгоритмы сортировки. Алгоритмы поиска. Итеративные и рекурсивные алгоритмы.

Модуль 3. Графовые алгоритмы и структуры данных

Тема 1. Алгоритмы для работы с графами

Основные понятия теории графов. Обход графов. Построение остовного дерева. Построение дерева решений

Тема 2. Кратчайшие пути.

Нахождение кратчайших путей. Алгоритм Дейкстры, алгоритм Флойда и переборные алгоритмы.

Модуль 4. Жадные алгоритмы и динамическое программирование

Тема 1. Динамическое программирование. Одномерное динамическое программирование. Двумерное динамическое программирование.

Тема 1. Жадные алгоритмы. Принцип жадного выбора. Алгоритм Хаффмана. Метод k-ближайших соседей.

4.3.2 Содержание лабораторных занятий по дисциплине

Модуль 1. Основные понятия теории алгоритмов и структур данных

Лабораторная работа 1. Запись основных алгоритмических конструкций на языке Python.

Лабораторная работа 2. Структуры данных.

Модуль 2. Элементы теории сложности алгоритмов

Лабораторная работа 3. Модель вычислений RAM. Анализ сложности наилучшего, наихудшего и среднего случая. Асимптотические обозначения. Скорость роста и отношения доминирования. Работа с асимптотическими обозначениями. Оценка эффективности алгоритмов. Классы P и NP.

Лабораторная работа 4. Алгоритмы сортировки массивов

Алгоритмы сортировки. Алгоритмы поиска. Итеративные и рекурсивные алгоритмы.

Модуль 3. Графовые алгоритмы и структуры данных

Лабораторная работа 5. Алгоритмы для работы с графами

Основные понятия теории графов. Обход графов. Построение остовного дерева.

Построение дерева решений

Лабораторная работа 6. Кратчайшие пути.

Нахождение кратчайших путей. Алгоритм Дейкстры, алгоритм Флойда и переборные алгоритмы.

Модуль 4. Жадные алгоритмы и динамическое программирование

Лабораторная работа 7. Динамическое программирование. Одномерное динамическое программирование. Двумерное динамическое программирование.

Лабораторная работа 8. Жадные алгоритмы. Принцип жадного выбора. Алгоритм Хаффмана. Метод k-ближайших соседей.

5. Образовательные технологии

В процессе преподавания дисциплины «Алгоритмы и структуры данных» применяются следующие образовательные технологии: лекционно-зачетная система обучения, технология развития критического мышления, изложение теоретических положений в ходе лекционных занятий с применением современного презентационного оборудования, проведение групповых дискуссий. Основная литература, презентации и конспекты лекций предоставляются студентам в электронном виде.

6. Учебно-методическое обеспечение самостоятельной работы студентов

6.1 Виды самостоятельной работы и её контроля

- проработки лекционного материала (настоятельно рекомендуется самостоятельное практическое решение всех разобранных на лекциях упражнений);
- изучения рекомендованной литературы и материалов соответствующих форумов интернет;
- подготовки к отчетам по лабораторным работам;
- подготовки к сдаче промежуточных форм контроля.

Пакет заданий для самостоятельной работы выдается по истечению месяца с начала семестра, определяются предельные сроки их выполнения и сдачи.

Контрольная работа – средство проверки умений применять полученные знания для решения задач определенного типа по теме или разделу.

Перед контрольной работой по каждому модулю студент должен самостоятельно повторить и освоить соответствующий теоретический материал по данному модулю, систематизировать необходимые алгоритмы, детально анализировать ранее решенные на практических занятиях задачи и упражнения. Задания по контрольной работе составлены для проверки освоения необходимых умений и навыков решения задач по тематике данного модуля.

Критерии оценки по контрольной работе

Если студент владеет по данному модулю навыками решения типичных задач, то по этому модулю ему выставляются:

- 1) 50 баллов;
- 2) 40 баллов в случае наличия неточностей;
- 3) 20 баллов в случае наличия некоторых допустимых ошибок.

Эти баллы учитываются при выводе общего результата как интегральной оценки, складывающейся из текущего контроля и промежуточного контроля.

Примерное распределение времени самостоятельной работы студентов:

Вид самостоятельной работы	Примерная трудоёмкость, а.ч.		
	Очная	Очно-заочная	заочная
Текущая СРС			
работа с лекционным материалом, с учебной литературой	6		
опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях)	4		
самостоятельное изучение разделов дисциплины			
выполнение домашних заданий, домашних контрольных работ	5		
подготовка к лабораторным работам и практическим занятиям	5		
подготовка к контрольным работам	8		
подготовка к зачету	8		
Творческая проблемно-ориентированная СРС			
поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме	4		
исследовательская работа, участие в конференциях, семинарах, олимпиадах	4		
Итого СРС:	44		

Учебно-методическое обеспечение СРС

№	Вид самостоятельной работы	Вид контроля	Учебно-методическое обеспечение
1.	Проработка лекционного материала	Контрольный фронтальный опрос	См. разделы 7.3, 8, 9 данного документа
2.	Изучение рекомендованной литературы материалов соответствующих форумов интернет	Контрольный фронтальный опрос прием и представление рефератов	См. разделы 7.3, 8, 9 данного документа
3.	Подготовка к отчетам по лабораторным работам.	Проверка выполнения работ опрос по теме работы.	См. разделы 7.3, 8, 9 данного документа
4.	Подготовка к сдаче промежуточных форм контроля	Контрольные работы по каждому модулю.	См. разделы 7.3, 8, 9 данного документа

Типовые задания для самостоятельной работы

1. Нажатием на кнопку организовать диалог по выбору файла любых форматов, поддерживаемых WindowsMediaPlayer, и проиграть его содержимое в окне.

Указания к решению. Поместим на форму кнопку и невидимый объект openFileDialog1, затем для размещения элемента управления WindowsMediaPlayer на форму внесем предварительно его в раздел General панели инструментов: вызовем контекстное меню раздела General панели инструментов, выберем пункт "Choose Items", в открывшемся окне для выбора компонент на вкладке COM Components поставим галочку рядом с Windows Media Player и нажмем ОК. Появившийся в разделе General появится элемент Windows Media Player перетащим на форму (или дважды щелкнем по этому элементу) и растянем его по ширине формы. Во вкладке Properties изменим имя полученного объекта на «WMP1».

В обработчике щелчка по кнопке (подпишемся на щелчок по кнопке) занесем имя выбранного файла в свойство WMP1.URL.

2. XOR – логическая операция, которая принимает значение «истина» только если всего один из аргументов имеет значение «истина».

Значение выражения $a \text{ xor } 0$ равно

- a. a
- b.
- c. 0

- d. 1
 - e. $a * a$
3. XOR – логическая операция, которая принимает значение «истина» только если всего один из аргументов имеет значение «истина». Свойство $a \text{ xor } b \text{ xor } b = a$ можно использовать для:
- a. Обмена значениями двух целых переменных
 - b. Шифрования с ключом b
 - c. Шифрования документов MsOffice
 - d. Установки старшего бита числа a в 0
4. XOR – логическая операция, которая принимает значение «истина» только если всего один из аргументов имеет значение «истина». Чему равно значение выражения $a \text{ xor } a$?
5. Оператор присваивания ($=$) вычисляет выражение, заданное в его правой части, и присваивает результат переменной, идентификатор которой расположен в левой части. Например, $z := (x + y)/2$. Во избежание ошибок присваивания необходимо следить, чтобы тип выражения был совместим с типом переменной. Под совместимостью типов данных понимается возможность автоматического преобразования значений одного типа данных в значения другого типа данных. Допустимым оператором присваивания для переменной вещественного типа является:
- a. $y := 2.0 * 1000 * 2000 * 2000$
 - b. $y := 1000 * 2000 * 2000$
 - c. $y := 10 * 1000 * 2000 * 200$
 - d. $y := 4 * 1000 * 1000 * 1000$
6. Оператор присваивания ($=$) вычисляет выражение, заданное в его правой части, и присваивает результат переменной, идентификатор которой расположен в левой части. Например, $z := (x + y)/2$. Во избежание ошибок присваивания необходимо следить, чтобы тип выражения был совместим с типом переменной. Под совместимостью типов данных понимается возможность автоматического преобразования значений одного типа данных в значения другого типа данных. В каких случаях произойдет автоматическое преобразование в операторе присваивания?
- a. Переменной с типом данных Integer присвоить выражение с типом данных Byte
 - b. Выражение с типом данных Integer присвоить переменной с типом данных Real
 - c. Переменной с типом данных Integer присвоить выражение с типом данных Single
 - d. Переменной с типом данных Byte присвоить выражение с типом данных Word
7. Оператор присваивания ($:=$) вычисляет выражение, заданное в его правой части, и присваивает результат переменной, идентификатор которой расположен в левой части. Например, $z := (x + y)/2$. Во избежание ошибок присваивания необходимо следить, чтобы тип выражения был совместим с типом переменной. Под совместимостью типов данных понимается возможность автоматического преобразования значений одного типа данных в значения другого типа данных. Чему будет равно значение переменной y , равное 3, после выполнения оператора присваивания $y := y \text{ xor } 0$?

7. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

7.1 Типовые контрольные задания

7.1.1 Темы рефератов для практических занятий

1. Консольные приложения Delphi и C#. Сравнительная характеристика.
2. Компьютерный вывод рекуррентных формул (на примере задачи разбиения).
3. Длинная целочисленная арифметика.
4. Числа Фибоначчи. Вычислительные аспекты.
5. Настольные графические приложения.
6. Проекты с несколькими формами.
7. Компьютерная графика в C#-проектах.
8. OpenGL в проектах C#.
9. Проекты с применением веб-камер. Распознавание цифр.
10. Визуальные компоненты Visual C#.
11. Класс BigInteger в задачах криптографии

7.1.2 Примерные упражнения и задания к лабораторным занятиям и для самопроверки

Задание 1. Найти время вычисления операций +, -, *, /.

Указания к решению. Во-первых, необходим цикл. Во-вторых, пустой цикл или цикл с присвоением, чтобы время этого цикла (время на организацию цикла) отнять. Самое главное – нужно учитывать мультизадачность, что совсем трудно; то, что при разных запусках наша программа запускается в одном потоке – утешение малое, т.к. этому потоку будет отводиться малое время. Целесообразно программу запускать 10 раз (программным путем).

Задание 2. Составить программу с двумя кнопками «Действия с блокнотом» и «Запуск калькулятора» и текстовым контейнером.

При щелчке на первую кнопку запустить программу Блокнот, ввести в первую строку несколько нулей, во вторую - несколько единиц, затем сохранить текст в текущем каталоге в файле 1234.txt (в предположении отсутствия одноименного файла), после чего программа закроет Блокнот.

При щелчке по второй кнопке проверить, запущено ли стандартное приложение Калькулятор, в случае, если не запущено, программа должна запустить его, затем передачей в него кодов нажатых клавиш выполнить эмуляцию видимых действий по суммированию $0+1+2+ \dots + 100$. При щелчке по контейнеру текста выполнить бесконечный вывод в него буквы «А».

7.1.3 Примерный перечень заданий для текущего контроля

Упр. 1. Введите с клавиатуры n , объявите одномерный массив a из n целых элементов и выведите построчно на экран: а) с использованием индексов, б) с помощью оператора `foreach`.

Упражнение 2. Заполнить прямоугольный массив $n*m$ ($n=m=4$) случайными целыми числами $< k=20$ и просуммировать все четные элементы, сумму вывести в формате с 5 знаками. Упражнение 3. Измените решение первого упражнения на вывод отсортированного массива, добавить вывод индекса значения 0 в отсортированном массиве.

Упражнение 4 (см. в тексте лекции). Обратите внимание на аргумент функции `Main`. Аргумент `string [] args` является массивом строк. Это параметры, принимаемые из командной строки (задаются после имени программы с пробельным разделителем). Перейдем в Windows в режим командной строки, используя главное меню. Затем, используя команды `cd`, перейдем в папку, которая содержит исполняемый файл следующей программы. Пусть это `A.exe`. Наберите командной строке `A 12 13 14` и нажмите `Enter` для выполнения.

Упражнение 5. Программа должна удалить из папки `C:\SomeDir` своего компьютера все файлы с расширением `bmp`.

Упражнение 6. Введите предложение и подсчитайте количество слов в нем, учитывая всевозможные разделители.

Упражнение 7. Заданную с клавиатуры строку выведите в верхнем и нижнем регистре.

Упражнение 8. Создание коллекции - строки статуса с панелями; в первую панель выводятся текущие координаты курсора мыши, во вторую – текущее время. Для обеих панелей предусмотреть всплывающие подсказки («координаты мыши» и текущую дату соответственно).

Упражнение 9. При запуске программы на поверхности формы нарисовать $m=10$ рядов квадратных кнопок по $m=10$ кнопок в ряду, размеры кнопки – $30*30$ Затем выполнить дрожание кнопок в 10 итерациях.

Упражнение 10. Создать проект из двух форм. При нажатии на кнопку первой формы предьявляется вторая форма, а первая начинает «прогулку» по невидимой окружности с постепенным усилением прозрачности.

7.1.3 Контрольные вопросы к экзамену

1. Основные определения алгоритмизации. Свойства алгоритмов.
2. Формы записи алгоритмов.
3. Базовые алгоритмические структуры.
4. Понятие о структурах данных. Уровни структур данных.
5. Классификация структур данных.
6. Перечисляемый тип данных.
7. Стандартные типы данных.

8. Алгоритмы обработки простых структур данных.
9. Массивы. Многомерные массивы.
10. Динамические массивы.
11. Строки.
13. Алгоритмы сортировки. Сортировка выбором. Сортировка обменом (пузырек). Сортировка вставками.
14. Алгоритмы сортировки. Сортировка слиянием. Сортировка Шелла. Быстрая сортировка.
Пирамидальная сортировка.
15. Анализ сложности алгоритмов сортировки.
16. Алгоритмы поиска. Последовательный поиск. Бинарный поиск. Интерполирующий поиск. Фибоначчиев поиск.
17. Анализ сложности алгоритмов поиска.
18. Файлы. Операции с данными на внешних носителях: внешний поиск, внешняя сортировка.
19. Способы представления и определения алгоритмов.
20. Получисленные алгоритмы.
21. Комбинаторные алгоритмы.
22. Рекурсивные алгоритмы.
23. Свойства структур данных.
24. Статические структуры данных.
25. Полудинамические и динамические структуры данных.

7.2. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Общий результат выводится как интегральная оценка, складывающаяся из текущего контроля - 50% и промежуточного контроля - 50%.

Текущий контроль по дисциплине включает:

- реферат- 10 баллов,
- выполнение лабораторных или практических заданий – 40 баллов,
- выполнение домашних (аудиторных) контрольных работ - 50 баллов.

Промежуточный контроль по дисциплине включает: письменная контрольная работа – 100 баллов.

8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.

а) адрес сайта курса

<http://cathedra.dgu.ru/EducationalProcess.aspx?Value=18&id=6>

б) основная литература:

1. Гаврилов, Г.П. Задачи и упражнения по дискретной математике : учебное пособие [Электронный ресурс] / Г. П. Гаврилов ; Гаврилов Г. П. - М. : Физматлит, 2009. - 416 с. - ISBN 978-5-9221-0477-7. Российская государственная библиотека (РГБ) URL: http://нэб.пф/catalog/000199_000009_004393971/ — Режим доступа: http://нэб.пф/catalog/000199_000009_004393971/
2. Павловская Т.А. Программирование на языке высокого уровня С# [Электронный ресурс] / Т.А. Павловская. — 2-е изд. — Электрон. текстовые данные. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 245 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/73713.html>
3. Токманцев Т.Б. Алгоритмические языки и программирование: учебное пособие / Токманцев Т.Б.— Е.: Уральский федеральный университет, ЭБС АСВ, 2013. 104— с.
4. **Андреева, Татьяна Анатольевна.** Программирование на языке Pascal : учеб. пособие / Андреева, Татьяна Анатольевна. - М. : Изд-во Интернет-Ун-та Информ. Технологий: БИНОМ. Лаб. знаний, 2006. - 234,[5] с. - (Основы информационных технологий). - ISBN 5-9556-0025-6 (ИНТУИТ.РУ) : 230-00.

в) дополнительная:

5. Иванов И.П. Сборник задач по курсу «Алгоритмы и структуры данных» [Электронный ресурс] : методические указания / И.П. Иванов, А.Ю. Голубков, С.Ю. Скоробогатов. — Электрон. текстовые данные. — М. : Московский государственный технический университет имени Н.Э. Баумана, 2013. — 36 с. — 978-5-7038-3681-1. — Режим доступа: <http://www.iprbookshop.ru/31548.html>
6. Мейер Б. Инструменты, алгоритмы и структуры данных [Электронный ресурс] / Б. Мейер. — 2-е изд. — Электрон. текстовые данные. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 542 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/73680.html>

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

- 1) eLIBRARY.RU [Электронный ресурс]: электронная библиотека / Науч. электрон. б-ка. — Москва, 1999 – . Режим доступа: <http://elibrary.ru/defaultx.asp>
- 2) Moodle [Электронный ресурс]: система виртуального обучением: [база данных] / Даг. гос. ун-т. – Махачкала, г. – Доступ из сети ДГУ или, после регистрации из сети ун-та, из любой точки, имеющей доступ в интернет. – URL: <http://moodle.dgu.ru/>
- 3) Электронный каталог НБ ДГУ [Электронный ресурс]: база данных содержит сведения о всех видах лит, поступающих в фонд НБ ДГУ/Дагестанский гос. ун-т. – Махачкала, 2010 – Режим доступа: <http://elib.dgu.ru>, свободный

Видеокурсы лекций:

- 1) <https://www.coursera.org/>
- 2) <https://www.udacity.com/>

3) <https://www.intuit.ru/>

Форумы по компьютерным наукам и программированию:

1) www.stackoverflow.com

2) <http://www.cyberforum.ru/>

3) <http://citforum.ru/http://www.intuit.ru/>

10. Методические указания для обучающихся по освоению дисциплины. При решении лабораторных заданий программистский подход непременно должен присутствовать (без него решение не будет полноценным), однако, он не должен заслонять сугубо математические (доказательство и др.) и алгоритмические (построение, оптимизация, верификация и др.) аспекты.

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем.

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используются следующее программное обеспечение: Microsoft Visual Studio Express, Microsoft Windows, Ubuntu Linux, Skype.

Также студентам предоставляется доступ к российским и международным электронным библиотекам через компьютеры университета.

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.

Имеется необходимая литература в библиотеке, медиапроектор и компьютер для проведения лекций-презентаций.

Лабораторные занятия проводятся в компьютерных классах с необходимым программным обеспечением. Вся основная литература предоставляется студенту в электронном формате.