

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кубанский государственный университет»
Факультет компьютерных технологий и прикладной математики

УТВЕРЖДАЮ:

Проректор по учебной работе,
качеству образования, первый
проректор


подпись
Хагуров Т.А.
« 27 » 04 2018



**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.В.ДВ.04.01 «АЛГОРИТМЫ И АНАЛИЗ СЛОЖНОСТИ»**

Направление подготовки 02.03.03 Математическое обеспечение и
администрирование информационных систем

Профиль Технология программирования

Программа подготовки Академическая

Форма обучения Очная

Квалификация выпускника Бакалавр

Краснодар 2018

Рабочая программа дисциплины «Алгоритмы и анализ сложности» составлена в соответствии с Федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки (профиль) 02.03.03 Математическое обеспечение и администрирование информационных систем.

Программу составила: О.Н. Лапина доцент кафедры вычислительных

технологий, канд. физ.-мат. наук



Рабочая программа дисциплины утверждена на заседании кафедры вычислительных технологий от «3» апреля 2018 г., протокол № 7

Заведующий кафедрой (разработчика) А.И. Миков



подпись

Рабочая программа обсуждена на заседании кафедры интеллектуальных информационных систем от «12» апреля 2018 г., протокол № 5.

Заведующий кафедрой (выпускающей) К.И. Костенко



Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики протокол № 1 от «20» апреля 2018 г.

Председатель УМК факультета Малыхин К.В.



Рецензенты:

Зайков В.П., ректор НЧОУ ВО «Кубанский институт информзащиты», доктор экономических наук., к.т.н., доцент.

Гаркуша О.В., доцент кафедры информационных технологий ФБГОУ ВО «Кубанский государственный университет», кандидат физико-математических наук, доцент.

1. Цели и задачи изучения дисциплины.

1.1. Цели освоения дисциплины.

Целью преподавания и изучения дисциплины «Алгоритмы и анализ сложности» является ознакомление студентов с фундаментальными алгоритмами обработки информации, с современными методами исследования алгоритмов и оценки их алгоритмической сложности.

1.2. Задачи дисциплины.

Студент должен **получить** знания об основных алгоритмах и методах оценки сложности алгоритмов; **уметь** разрабатывать оптимальные алгоритмы для решения поставленных задач; **изучить** основные структуры представления данных и алгоритмы обработки данных.

1.3. Место дисциплины в структуре образовательной программы

Дисциплина «Алгоритмы и анализ сложности» относится к вариативной части цикла профессиональных дисциплин. Для изучения дисциплины необходимо знание основ программирования, языков программирования. Знания, получаемые при изучении теории сложности алгоритмов, используются при изучении других дисциплин профессионального и вариативного цикла учебного плана бакалавра, а также при работе над курсовыми проектами.

1.4 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы.

Изучение данной учебной дисциплины направлено на формирование у обучающихся общепрофессиональных/профессиональных компетенций (ОПК/ПК)

№ п.п.	Индекс компетенции	Содержание компетенции (или её части)	В результате изучения учебной дисциплины обучающиеся должны		
			знать	уметь	владеть
1.	ОПК-8	способностью использовать знания методов проектирования и производства программного продукта, принципов построения, структуры и приемов	основные фундаментальные алгоритмы обработки информации, методы оптимизации алгоритмов	эффективно использовать базовые алгоритмы для решения научных и проектно-технологических задач	математическим аппаратом для написания алгоритмов обработки информации

2.	ПК-3	готовностью к разработке моделирующих алгоритмов и реализации их на базе языков и пакетов прикладных программ моделирования	Основные алгоритмы моделирования систем и процессов; языки программирования	Разрабатывать алгоритмы и программы с использованием пакетов прикладных программ	Методами математического моделирования
----	------	---	---	--	--

2. Структура и содержание дисциплины.

2.1 Распределение трудоёмкости дисциплины по видам работ.

Общая трудоёмкость дисциплины составляет 3 зач.ед. (108 часов), их распределение по видам работ представлено в таблице

Вид учебной работы		Всего часов	Семестры (часы)			
			3			
Контактная работа, в том числе:						
Аудиторные занятия (всего):		72	72			
Занятия лекционного типа		36	36	-	—	—
Лабораторные занятия		36	36	-	—	—
Занятия семинарского типа (семинары, практические занятия)		—	—	—	—	—
		—	—	—	—	—
Иная контактная работа:						
Контроль самостоятельной работы (КСР)		4	4	-	—	—
Промежуточная аттестация (ИКР)		0,2	0,2	-	—	—
Самостоятельная работа, в том числе:		31,8	31,8			
Курсовая работа		—	—	—	—	—
Проработка учебного (теоретического) материала		15,8	15,8	-	—	—
Выполнение индивидуальных заданий (подготовка сообщений, презентаций)		16	16	—	—	—
Реферат		—	—	—	—	—
Подготовка к текущему контролю		-	-	-	—	—
Контроль:						
Подготовка к экзамену		-	-	-	—	—
Общая трудоёмкость	час.	108	108	-	—	—
	в том числе контактная работа	76,2	76,2	-	—	—
	зач. ед.	3	3		—	—

2.2 Структура дисциплины:

Распределение видов учебной работы и их трудоёмкости по разделам дисциплины. Разделы дисциплины, изучаемые в 3 семестре (очная форма)

№ раздела	Наименование разделов	Количество часов				
		Всего	Аудиторная работа			Самостоятельная работа
			Л	ЛР	КСР	
1	2	3	4	5	6	7
1	Основы анализа алгоритмов.	24	8	10	2	6

2	Оптимизация алгоритмов	4	2	2		2
3	Основные алгоритмы обработки информации	46	12	14	2	12
4	Стратегии алгоритмов	24	8	6		8
5	Алгоритмы с р-исполнителями	6	2	2		2
6	Основы теории вычислимости	8	4	2		1,8
	ИКР	0,2				
	Итого:	108	36	36	4	31,8

2.3 Содержание разделов дисциплины:

2.3.1 Занятия лекционного типа.

№ раз-дела	Наименование раздела	Содержание раздела	Форма текущего контроля
1	2	3	4
1	Основы анализа алгоритмов.	Понятие алгоритма. Основные свойства алгоритма. Основные методы и приемы анализа сложности алгоритма. Асимптотический анализ верхней и средней оценок сложности алгоритмов; сравнение наилучших, средних и наихудших оценок; стандартные классы сложности; эмпирические измерения эффективности алгоритмов; накладные расходы алгоритмов по времени и памяти; рекуррентные соотношения и анализ рекурсивных алгоритмов.	ЛР
2	Оптимизация алгоритмов	Понятие оптимизации алгоритмов. Алгоритм Штрассена перемножения матриц; алгоритмы перемножения длинных целых чисел и возведения в целую степень.	ЛР
3	Основные алгоритмы обработки информации	Основные алгоритмы над числами; алгоритмы последовательного и бинарного поиска; алгоритмы сортировки сложности $O(N*N)$ и $O(N*\log N)$; деревья бинарного поиска; представление графов (списки и матрицы смежности); поиск в глубину и поиск в ширину; алгоритмы поиска кратчайших путей (алгоритмы Дейкстры и Флойда); транзитивное замыкание (алгоритм Флойда);	ЛР, РГЗ

		алгоритмы построения минимального покрывающего дерева (алгоритмы Прима и Крускала); топологическая сортировка.	
4	Стратегии алгоритмов	Полный перебор; метод “разделяй и властвуй”; “жадные” алгоритмы; бэктрекинг (перебор с возвратами); метод ветвей и границ; эвристический поиск; поиск по образцу, алгоритмы обработки строк; алгоритмы аппроксимации числовых функций.	ЛР, РГЗ
5	Алгоритмы с р-исполнителями	Основы сложности параллельных и распределенных алгоритмов.	ЛР
6	Основы теории вычислимости	Конечные автоматы; контекстно-свободные грамматики; разрешимые и неразрешимые проблемы; невычислимые функции; применение невычислимости.	ЛР

2.3.2 Занятия семинарского типа.

Занятия семинарского типа – не предусмотрены.

2.3.3 Лабораторные занятия.

№ работы	Наименование раздела дисциплины	Наименование лабораторных работ
1	Основы анализа алгоритмов.	Понятие сложности алгоритма, анализ сложности линейных алгоритмов, алгоритмов содержащих циклы. Написать процедуры: 1) скалярного перемножения векторов; 2) сложения и перемножения матриц.
2	Основы анализа алгоритмов.	Оценка сложности ветвящихся алгоритмов. Написать процедуры: Нахождение максимального и минимального элемента в массиве.
3	Основы анализа алгоритмов.	Оценка сложности простейших рекурсивных алгоритмов. Рекуррентные соотношения. Написать процедуры и оценить сложность алгоритма: 1) Рекурсивный алгоритм вычисления функции факториал. 1) Рекурсивная функция для поиска корня уравнения $f(x) = 0$ с заданной точностью методом деления отрезка пополам.

4	Основы анализа алгоритмов.	Оценка сложности рекурсивных алгоритмов. Рекуррентные соотношения. Линейная рекурсия, нелинейная рекурсия. Написать процедуры: 1) Написать процедуру, которая формирует все $n!$ перестановок n элементов $a_1, a_2, \dots, a_n$ без использования другого массива. Указание: Рассматривайте задачу получения всех перестановок m элементов как состоящую из m подзадач, строящих все перестановки $a_1, a_2, \dots, a_{m-1}$, за которыми следует a_m. В i-той подзадаче сначала меняются местами a_i и a_m. 2) Вычисление чисел Фибоначчи (рекурсивные и не рекурсивные алгоритмы).
5	Основы анализа алгоритмов.	Асимптотический анализ верхней и средней оценок сложности алгоритмов. Анализ сложности взаимно рекурсивных алгоритмов. Написать алгоритм: взаимно-рекурсивные функции $g_0(n)$ и $g_1(n)$ для целого неотрицательного n, подсчитывают и возвращают логическое значение, равное 1 (истина), если в двоичном представлении n имеется четное число цифр "1", и 0 (ложь) в противном случае.
6	Оптимизация алгоритмов	Алгоритмы перемножения длинных целых чисел. Возведение чисел в целую степень.
7	Основные алгоритмы обработки информации	Создание двоичного дерева поиска. Операции с двоичными деревьями.: Добавление элемента, удаление элемента. Подсчет количества элементов.
8	Основные алгоритмы обработки информации	Алгоритмы последовательного и бинарного поиска. Сложность алгоритма поиска элемента в бинарном дереве.
9	Основные алгоритмы обработки информации	Представление графов (матрицы смежности). Процедуры с матричным представлением графа.
10	Основные алгоритмы обработки информации	Представление графов (списковое представление). Обход графа, создание графа.

11	Основные алгоритмы обработки информации	Представление графов (списковое представление). Операции изменения (добавление элемента, удаление элемента) и удаления графа.
12	Основные алгоритмы обработки информации	Алгоритмы поиска в глубину и ширину. Алгоритмы поиска кратчайших путей.
13	Основные алгоритмы обработки информации	Сложность алгоритмов сортировки. Максимальная, минимальная, средняя сложность. Сложность быстрой сортировки.
14	Стратегии алгоритмов	Стратегии алгоритмов. Жадные алгоритмы (задача коммивояжера).
15	Стратегии алгоритмов	Стратегии алгоритмов. Алгоритмы обработки строк.
16	Стратегии алгоритмов	Стратегии алгоритмов. Задача раскраски графа. Метод «Перебор с возвратом».
17	Алгоритмы с r-исполнителями	Сложность параллельных алгоритмов перемножения матриц
18	Основы теории вычислимости	Разработка простейших программ машины Тьюринга. Программы для вычисления $Z(x) = 0$ в алфавитах A_1 и A_2 . Вычисление $x_1 \sqcap x_2$ в алфавите A_1 .

2.3.4 Примерная тематика курсовых работ (проектов)

Курсовые работы – не предусмотрены.

2.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

№	Вид СРС	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	2	3
1	Проработка учебного материала, выполнение индивидуальных заданий.	Миков А.И., Лапина О.Н. Вычислимость и сложность алгоритмов. М-во образования и науки Рос. Федерации, Кубанский гос. ун-т, Каф. вычислительных технологий. - Краснодар: 2013. - 78 с.

Учебно-методические материалы для самостоятельной работы обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ) предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа,

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа,

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

3. Образовательные технологии.

Используемые интерактивные образовательные технологии:

- Компьютерные презентации и обсуждение.
- Разбор конкретных ситуаций (задач), тренинги по решению задач, компьютерные симуляции (программирование алгоритмов).

4. Оценочные средства для текущего контроля успеваемости и промежуточной аттестации.

4.1 Фонд оценочных средств для проведения текущего контроля.

Фонд оценочных средств дисциплины состоит из средств текущего контроля выполнения индивидуальных расчетно-графических заданий, лабораторных работ, средств итоговой аттестации (зачета в 3 семестре).

Оценка успеваемости осуществляется по результатам:

- выполнения лабораторных работ;
- оценки, выставляемой при сдаче индивидуальных расчетно-графических заданий – разработка, реализация и оценка сложности алгоритма;
- ответа на зачете (для выявления знания и понимания теоретического материала дисциплины).

4.2 Фонд оценочных средств для проведения промежуточной аттестации.

По дисциплине студентом выполняется два индивидуальных расчетно-графических заданий – разработка, реализация и оценка сложности алгоритма. Темы заданий для каждого студента различны. Задача РГЗ состоит в проверке умений студента и проверки эффективности его самостоятельной работы.

Общая тематика соответствует тематике лабораторных работ.

Образец РГЗ – задания на разработку алгоритма и компьютерной программы

Задание: Написать процедуру. Найти вид функции сложности алгоритма (программы).

Произвести экспериментальные расчеты, подтверждающие правильность аналитических результатов.

Отчет по выполнению РГЗ должен содержать:

- постановку задачи;
- краткое описание алгоритма;
- листинг программы на одном из языков программирования;
- примеры работы программы;
- результаты экспериментальных расчетов сложности алгоритма;
- аналитический анализ сложности алгоритма;
- список использованной литературы.

Список индивидуальных заданий РГЗ для выбора:

Задание: Написать процедуру. Найти вид функции сложности алгоритма (программы). Произвести экспериментальные расчеты, подтверждающие правильность аналитических результатов.

Задача 1.

1. Написать процедуру, отыскивающую в бинарном дереве ключ, следующий за данным (в смысле отношения порядка);
2. Написать процедуру, подсчитывающую количество вершин, имеющих двух потомков; Перестроить дерево так, чтобы значения ключей возрастали по ярусам, а внутри яруса – слева направо (при сохранении строения дерева);
3. Написать процедуру определения того, являются ли два данных дерева изоморфными в теоретико-графовом смысле (без учета полей *inf* и того, является ли потомок левым или правым);
4. Написать процедуру отыскания в заданном дереве поддеревьев определенного вида (без учета полей *inf*);
5. Написать процедуру, удаляющую из дерева все вершины, имеющие только левых потомков. Все остальные вершины должны остаться в дереве;
6. Написать процедуру определения того, являются ли два данных дерева изоморфными в теоретико-графовом смысле (без учета полей *inf* и того, является ли потомок левым или правым, а также без учета направлений дуг и корней);
7. Написать процедуру, строящую дерево разбора для арифметических выражений;
8. Написать процедуру балансировки дерева;
9. Написать процедуру, подсчитывающую количество висячих вершин в дереве;
10. Написать процедуру, подсчитывающую количество вершин, имеющих одного потомка;
11. Написать процедуру, отыскивающую в бинарном дереве ключ, предыдущий данному (в смысле отношения порядка);
12. Написать процедуру, определяющую, является ли бинарное дерево вырожденным (т.е. списком);
13. Перестроить дерево так, чтобы направления дуг инвертировались через один ярус (в 1-м ярусе направления сохраняются);
14. Написать процедуру, изменяющую направления всех дуг в бинарном дереве на обратные;
15. Написать процедуру, вычисляющую количество ветвей бинарного дерева;
16. Написать процедуру, вычисляющую высоту дерева;
17. Написать процедуру включения информации в тернарное дерево. При этом в дерево

- могут включаться многократно одинаковые значения информации;
18. Написать процедуру, вычисляющую количество вершин на заданном уровне;
 19. Написать процедуру, вычисляющую «ширину» дерева;
 20. Написать процедуру, вычисляющую разность между количествами вершин, имеющих только левых потомков и вершин, имеющих только правых потомков;
 21. Написать процедуру, вычисляющую длину самой короткой ветви дерева. Задача 2.
 1. Написать процедуру, отыскивающую наибольшее независимое множество ребер в графе по его списковому представлению;
 2. Написать процедуру проверки того, является ли неориентированный граф самодополнительным, по его матричному представлению;
 3. Написать процедуру, реализующую операцию произведения двух неориентированных графов по их списковому представлению;
 4. Написать процедуру, определяющую, является ли неориентированный граф G_1 подграфом графа G_2 , по их списковым представлениям;
 5. Написать процедуру, решающую задачу изоморфизма ориентированных графов, заданных списковыми представлениями;
 6. Написать процедуру вычисления дополнения неориентированного графа по его списковому представлению;
 7. Написать процедуру вычисления хроматического многочлена неориентированного графа по его матрице смежностей;
 8. Написать процедуру определения связанных компонент в графе;
 9. Написать процедуру, определяющую, является ли граф эйлеровым.; 10. Написать процедуру, отыскивающую число маршрутов длины n из i -й вершины в j -ю по матричному представлению графа;
 11. Написать процедуру, вычисляющую граф - n -мерный куб в списковом представлении;
 12. Написать процедуру укладки непланарного графа на торе;
 13. Написать процедуру, реализующую операцию композиции двух неориентированных графов по их списковому представлению;
 14. Написать процедуру, отыскивающую наибольшее независимое множество вершин в графе по его списковому представлению.;
 15. Написать процедуру, решающую задачу изоморфизма орграфов по их матричному представлению;
 16. Написать процедуру топологической сортировки неориентированного ациклического графа по его списковому представлению;
 17. Написать процедуру, вычисляющую число остовов в неориентированном графе по его списковому представлению;
 18. Написать процедуру отыскания кратчайших путей между всеми парами вершин ориентированного графа по его списковому представлению; 19. Написать процедуру, вычисляющую соединение двух графов по их списковым представлениям;
 20. Написать процедуру отыскания транзитивного замыкания графа по его списковому представлению;
 21. Написать процедуру, отыскивающую наименьшее вершинное покрытие графа по его списковому представлению;
 22. Написать процедуру, вычисляющую пересечение двух графов по их списковым представлениям.

1. Понятие алгоритма. Свойства алгоритма. Рекурсивные алгоритмы.
2. Анализ сложности алгоритмов. Основные понятия и определения. Функции сложности. Сложность данных.
3. Основные методы и приемы анализа сложности алгоритмов. Линейные и ветвящиеся алгоритмы. Примеры.
4. Основные методы и приемы анализа сложности алгоритмов. Циклические (for) алгоритмы. Примеры. Вычисление скалярного произведения векторов. Сложение матриц. Перемножение матриц.
5. Основные методы и приемы анализа сложности алгоритмов. Циклические (while, repeat) алгоритмы. Примеры. Алгоритм вычисления корня p -й степени из вещественного числа.
6. Анализ сложности рекурсивных алгоритмов. Рекуррентные уравнения. Линейная рекурсия. Нелинейная рекурсия. Перенос Ханойской башни.
7. Анализ сложности рекурсивных алгоритмов. Вычисление наибольшего общего делителя.
8. Анализ сложности взаимно рекурсивных алгоритмов. Рекуррентные уравнения. Решение рекуррентных уравнений. Примеры.
9. Алгоритмы «Разделяй и властвуй». Алгоритм умножения длинных целых чисел. Алгоритм возведения целого числа в большую целую степень.
10. Алгоритмы «Разделяй и властвуй». Алгоритм Ф.Штрассена перемножения матриц.
11. Алгоритмы «Разделяй и властвуй». Алгоритм перемножения длинных целых чисел.
12. Стратегии алгоритмов. Полный перебор, «жадные» алгоритмы.
13. «Жадные» алгоритмы. Задача о светофорах.
14. Задачи поиска: исчерпывающий поиск: перебор с возвратом, метод ветвей и границ.
15. Основные алгоритмы обработки строк.
16. Бинарные деревья. Типы данных для построения бинарных деревьев. Процедуры с бинарными деревьями.
17. Представление графов матрицами. Матрицы для типовых графов K_n , P_n , C_n , $K_{m,n}$. Процедуры построения матричных представлений.
18. Операции над графами. Процедуры выполнения операций для матричных представлений.
19. Списковое представление графов. Примеры построения. 20. Процедура просмотра спискового представления графа. 21. Процедуры поиска вершин и дуг в списковом представлении графа.
22. Процедуры добавления вершин и дуг в списковом представлении графа.
23. Процедуры удаления дуг и вершин из спискового представления графа.
24. Процедура уничтожения спискового представления графа.
25. Алгоритмы отыскания множества достижимых вершин в графе (поиск в глубину и в ширину).
26. Алгоритм поиска кратчайшего пути в графе (алгоритм Дейкстры).
27. Алгоритм поиска кратчайшего пути в графе (алгоритм Флойда).
28. Алгоритм отыскания транзитивного замыкания графа. Топологическая сортировка.
29. Алгоритмы поиска минимального покрывающего дерева (алгоритмы Прима и Крускала).
30. Процедуры сортировки. Минимальная, максимальная и средняя оценки сложности.

31. Сложность быстрой сортировки Хоара.
32. Файлы: организация и обработка, представление деревьями: В-деревья. 33. Методы внешней сортировки.
34. Сложность параллельных алгоритмов. Сложность распределенных алгоритмов.
35. Формализация понятия алгоритма. Машины Тьюринга. Табличное представление программ. Представление программ машин Тьюринга диаграммами. Программы для вычисления $Z(x) = 0$ в алфавитах A_1 и A_2 . Вычисление $x_1 + x_2$ в алфавите A_1 .
36. Нормальные алгорифмы Маркова. Правила подстановки. Схема алгоритма. Примеры.
37. Разрешимые и неразрешимые задачи. Проблема остановки.

4.2.1 Критерии оценивания к зачету

Оценка “зачтено” - практические задания выполнены в срок в объеме не менее 80%. студент демонстрирует правильные, уверенные действия по применению полученных знаний на практике, грамотное и логически стройное изложение материала при аргументации ответов на вопросы при защите лабораторных.

Оценка «не зачтено» - практические задания не выполнены либо предоставлены не в срок в объеме менее 60%, студент демонстрирует наличие грубых ошибок в ответе, непонимание сущности излагаемого вопроса, неумение применять знания на практике, неуверенность и неточность ответов на дополнительные и наводящие вопросы.

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

— при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;

— при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;

— при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине (модулю) предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа. Для

лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

5. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля).

5.1 Основная литература:

1. Тюкачев, Н.А. С#. Алгоритмы и структуры данных [Электронный ресурс] / Н.А. Тюкачев, В.Г. Хлебостроев. — Электрон. дан. — Санкт-Петербург: Лань, 2017. — 232 с. — Режим доступа: <https://e.lanbook.com/book/94748>.
2. Конова, Е.А. Алгоритмы и программы. Язык С++ [Электронный ресурс] : учебное пособие / Е.А. Конова, Г.А. Поллак. — Электрон. дан. — Санкт-Петербург : Лань, 2017. — 384 с. — Режим доступа: <https://e.lanbook.com/book/90158>.
3. Седжвик, Р. Алгоритмы на С++ / Р. Седжвик. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 1773 с. : ил. - Библиогр. в кн. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429164>
4. Королев Л.Н., Миков А.И. Информатика. Введение в компьютерные науки. – М.: Абрис, 2012.
5. Миков А.И., Лапина О.Н. Вычислимость и сложность алгоритмов. М- во образования и науки Рос. Федерации, Кубанский гос. ун-т, Каф. вычислительных технологий. - Краснодар: 2013. - 78 с.

Для освоения дисциплины инвалидами и лицами с ограниченными возможностями здоровья имеются издания в электронном виде в электронно-библиотечных системах «Лань» и «Юрайт».

5.2 Дополнительная литература:

1. Мейер, Б. Инструменты, алгоритмы и структуры данных / Б. Мейер. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 543 с. : схем., ил. - Библиогр. в кн. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429033>
2. Математические методы и модели исследования операций : учебник / ред. В.А. Колемаева. - Москва : Юнити-Дана, 2015. - 592 с. : ил., табл., граф. - Библиогр. в кн. - ISBN 978-5-238-01325-1 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=114719>

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

1. Российское образование, федеральный портал [Официальный сайт] — URL: <http://www.edu.ru>
2. <http://www.microsoft.com/en-us/default.aspx>.
3. <http://www.microsoft.com/visualstudio/en-us/products/2010-editions>

Программное обеспечение

1. С++.
2. Pascal.

7. Методические указания для обучающихся по освоению дисциплины

Лабораторные работы выполняются, как правило, в компьютерном классе. Отдельные работы при определении функции сложности алгоритма могут выполняться в аудитории.

На лабораторных работах изучаются основные методы оценки сложности алгоритма. Студент должен создать программу, реализующую заданный алгоритм и вычислить сложность алгоритма.

Расчетно-графическое задание по дисциплине состоит в разработке алгоритма, его реализации на языке программирования C++ или Pascal и оценке сложности алгоритма.

Задания являются индивидуальными, т.е. формулируются для каждого студента отдельно.

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

8.1 Перечень информационных технологий.

- Проверка домашних заданий и консультирование посредством электронной почты.
- Использование электронных презентаций при проведении практических занятий.

8.2 Перечень необходимого программного обеспечения.

- Программы для демонстрации и создания презентаций («Microsoft Power Point»).

8.3 Перечень информационных справочных систем:

Электронные библиотечные источники:

1. ЭБС Издательства «Лань» <http://e.lanbook.com> ,
2. ЭБС «Университетская библиотека онлайн» www.biblioclub.ru ,
3. ЭБС «Юрайт» <http://www.biblio-online.ru> ,
4. ЭБС «ZNANIUM.COM» www.znanium.com ,
5. ЭБС «BOOK.ru» <https://www.book.ru>.

9. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине (модулю).

№	Вид работ	Материально-техническое обеспечение дисциплины (модуля) и оснащенность
1.	Лекционные занятия	Лекционная аудитория, оснащенная презентационной техникой (проектор, экран, компьютер/ноутбук) и соответствующим программным обеспечением (ПО)
2.	Лабораторные занятия	Лаборатория, укомплектованная специализированной мебелью и техническими средствами обучения
3.	Групповые (индивидуальные) консультации	Аудитория с учебной мебелью (доски, столы, стулья)
4.	Текущий контроль, промежуточная аттестация	Аудитория с учебной мебелью

5.	Самостоятельная работа	Кабинет для самостоятельной работы, оснащенный компьютерной техникой с возможностью подключения к сети «Интернет», программой экранного увеличения и обеспеченный доступом в электронную информационно-образовательную среду университета.
----	------------------------	--