

Министерство науки и высшего образования Российской Федерации
 Федеральное государственное автономное образовательное учреждение
 высшего образования
 «СЕВЕРО-ВОСТОЧНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ ИМЕНИ М.К. АММОСОВА»
 Политехнический институт (филиал) ФГАОУ ВО «Северо-Восточный федеральный университет
 имени М.К. Аммосова» в г. Мирном.
 Кафедра фундаментальной и прикладной математики

Рабочая программа дисциплины

Б1.О.18 Алгоритмы и алгоритмические языки

для программы бакалавриата




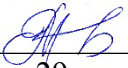
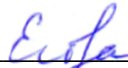
по направлению подготовки

01.03.02 Прикладная математика и информатика

Профиль подготовки: Математическое моделирование и вычислительная математика

Форма обучения: очная

Автор: Якушев Илья Анатольевич, к.ф.-м.н., доцент кафедры фундаментальной и прикладной математики, МПТИ (ф)СВФУ, Yakushevilya@mail.ru

<p>РЕКОМЕНДОВАНО</p> <p>Заведующий кафедрой фундаментальной и прикладной математики</p> <p> /Гадоев М.Г. протокол № <u>3</u> от «22» <u>февраля</u> 2019 г.</p>	<p>ОДОБРЕНО</p> <p>Заведующий кафедрой фундаментальной и прикладной математики</p> <p> /Гадоев М.Г. протокол № <u>3</u> от «22» <u>февраля</u> 2019 г.</p>	<p>ПРОВЕРЕНО</p> <p>Нормоконтроль в составе ОП пройден Ст.диспетчер УМО</p> <p> / <u>Баишева О.Ю.</u> «28» <u>марта</u> 2019 г.</p>
<p>Рекомендовано к утверждению в составе ОП</p> <p>Председатель УМС  /<u>Константинова Т.П./</u> протокол УМС № <u>3</u> от «29» <u>марта</u> 2019 г.</p>		<p>Эксперт УМС</p> <p> / <u>Егорова М.В.</u> «29» <u>марта</u> 2019 г.</p>

1. АННОТАЦИЯ
к рабочей программе дисциплины
Б1.О.18 Алгоритмы и алгоритмические языки
Трудоемкость 3 з.е.

1.1. Цель освоения и краткое содержание дисциплины

Цель освоения: Цель дисциплины – дать представление об алгоритмах и теориях программирования, в частности о базовых определениях алгоритмов программирования, функциональном подходе к программированию. Изучить общие основы языков программирования, построения алгоритмов и методов их оптимизации.

Краткое содержание дисциплины:

- Понятие алгоритмов.
- Алгоритмические языки.
- Оптимизация алгоритмов программирования.
- Модели программирования: функциональное, императивное, декларативное, структурное, логическое, объектно-ориентированное программирование.
- Функциональный подход к программированию.
- Теория типов и комбинаторная логика
- Категориальная комбинаторная логика и абстрактные машины
- Синтаксис языков программирования

1.2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Наименование категории (группы) компетенций	Планируемые результаты освоения программы (код и содержание компетенции)	Индикаторы достижения компетенций	Планируемые результаты обучения по дисциплине	Оценочные средства
Теоретические и практические основы профессиональной деятельности	ОПК-2. Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач	ОПК-1.1 Знает: математические основы программирования и языков программирования, организации баз данных и компьютерного моделирования; математические методы оценки качества, надежности и эффективности программных продуктов; математические методы организации информационной безопасности при разработке и эксплуатации	Знать общие черты и характеристики самых главных частей персонального компьютера; Уметь пользоваться операционной системой; Владеть (навыками) работы с операционной системой; Владеть (методиками) способами и средствами получения, хранения, передачи информации	Выполнение практических заданий, тест, устный опрос

		<p>программных продуктов и программных комплексов. ОПК-1.2 Умеет использовать этот аппарат в профессиональной деятельности. ОПК-1.3 Имеет навыки применения данного математического аппарата при решении конкретных задач</p>		
ОПК	<p>Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения (ОПК-5)</p>	<p>ОПК-5.1. Знает: методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий. ОПК-5.2. Умеет: применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий. ОПК-5.3. Владеет: навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач.</p>	<p>Знать: основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. Уметь: применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ. Владеть: программирования, отладки и тестирования прототипов программно-технических комплексов задач.</p>	<p>Выполнение практических заданий, тест, устный опрос</p>

1.3. Место дисциплины в структуре ОПОП

Индекс	Наименование дисциплины (модуля), практики	Семестр изучения	Индексы и наименования учебных дисциплин (модулей), практик	
			на которые опирается содержание данной дисциплины (модуля)	для которых содержание данной дисциплины (модуля) выступает опорой
Б1.О.18	Алгоритмы и алгоритмические языки	1	Базовый курс школьной программы дисциплины Информатика и ИКТ	Б1.О.27 Операционные системы Б1.О.35 Системы программирования Б1.В.ДВ.08.02 Защита информации

1.4. **Язык преподавания:** Русский

2. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Выписка из учебного плана:

Код и название дисциплины по учебному плану	Б1.О.18 Алгоритмы и алгоритмические языки	
Курс изучения	1	
Семестр(ы) изучения	1	
Форма промежуточной аттестации (зачет/экзамен)	Экзамен	
Курсовой проект/ курсовая работа (указать вид работы при наличии в учебном плане), семестр выполнения	-	
Грудоемкость (в ЗЕТ)	3	
Грудоемкость (в часах) (сумма строк №1,2,3), в т.ч.:	108	
№1. Контактная работа обучающихся с преподавателем (КР), в часах:	Объем аудиторной работы, в часах	В т.ч. с применением ДОТ или ЭО, в часах
Объем работы (в часах) (1.1.+1.2.+1.3.):	54	-
1.1. Занятия лекционного типа (лекции)	17	-
1.2. Занятия семинарского типа, всего, в т.ч.:		-
- семинары (практические занятия, коллоквиумы и т.п.)	34	-
- лабораторные работы		-
- практикумы		-
1.3. КСР (контроль самостоятельной работы, консультации)	3	-
№2. Самостоятельная работа обучающихся (СРС) (в часах)	27	
№3. Количество часов на экзамен (при наличии экзамена в учебном плане)	27	

3. Содержание дисциплины, структурированное по темам с указанием отведенного на них количества академических часов и видов учебных занятий

3.1. Распределение часов по темам и видам учебных занятий

Тема	Всего часов	Контактная работа, в часах									Часы СРС
		Лекции	из них с применением ЭО и ДОТ	Семинары (практические занятия, коллоквиумы)	из них с применением ЭО и ДОТ	Лабораторные работы	из них с применением ЭО и ДОТ	Практикумы	из них с применением ЭО и ДОТ	КСР (консультации)	
Алгоритмы. Машина Тьюринга. Нормальные алгоритмы Маркова	15	3		6							6
Модельная ЭВМ. Алгоритмические языки. Метаязыки	16	3		7							6
Подпрограммы. Процедуры и функции. Рекурсия	16	3		7					1		5
Итерация. Проектирование циклов рекуррентные вычисления	17	4		7					1		5
Последовательные таблицы. Поиск. Сортировка	17	4		7					1		5
Всего часов	81	17		34					3		27

3.2. Содержание тем программы дисциплины

Тема 1. Алгоритмы. Машина Тьюринга. Нормальные алгоритмы Маркова.

1. АЛГОРИТМЫ

1. Вводная часть
2. Интуитивное определение алгоритмов (понятие алгоритма, примеры алгоритмов, свойства алгоритмов)
3. Неформальные способы записи алгоритмов, алгоритмические конструкции: следование, ветвление, цикл (словесная запись, блок-схемы)
4. Исполнители алгоритма, программа
5. Об алгоритмических языках (достоинства и недостатки машинного языка, алгоритмические языки высокого уровня и их достоинства, разнообразие АЯ, трансляция с АЯ)

ЧИСЛОВЫЕ ТИПЫ

1. Паскаль-программа (структура программы, синтаксис и назначение заголовка, раздела описаний и раздела операторов, правила выполнения программы, правила расстановки пробелов и перевода строк).
2. Алфавит, идентификаторы (синтаксис и назначение идентификаторов, служебные слова)

3. Типы данных. Константы (их отличие от переменных, раздел описания констант, стандартные константы: maxint, true, false).

Переменные. Перечень всех типов данных Паскаля, стандартные типы; переменные и типы данных, контроль использования переменных; синтаксис раздела переменных

4. Числовые типы данных

4.1 Целый тип (запись целых чисел, описание переменных целого типа, операции над целыми числами, стандартные функции для целых чисел, maxint)

4.2 Вещественный тип (запись вещественных чисел, описание вещественных переменных, операции и стандартные функции для вещественных чисел)

5. Арифметические выражения (тип выражения, тип результата, правила записи арифметических выражений, правила вычисления выражений и определения их типа).

ОПЕРАТОР ПРИСВАИВАНИЯ. ВВОД. ВЫВОД

1. Оператор присваивания (синтаксис, семантика и прагматика оператора присваивания)

2. Процедуры ввода (процедура read: синтаксис и семантика, допустимые типы параметров ввода, readln)

3. Процедуры вывода (процедура write, вывод целых, особенности вывода вещественных чисел, процедура writeln).

МАШИНА ТЬЮРИНГА. НОРМАЛЬНЫЕ АЛГОРИТМЫ МАРКОВА

1. Необходимость и способы уточнения понятия алгоритма (символы и слова)

2. Машина Тьюринга (структура, такт работы, запись программы, правила выполнения программы, применимость и неприменимость алгоритма к слову, соглашения для сокращения записи программы)

3. Возможности МТ (композиция, разветвление и повторение МТ)

4. Тезис Тьюринга

5. Нормальные алгоритмы Маркова (операция подстановки и формулы подстановки, запись НАМ, два вида стрелок, правила выполнения, тезис Маркова)

6. Алгоритмически неразрешимые проблемы (определение алгоритмически разрешимых и неразрешимых задач, понятия записи алгоритма и самоприменимости, теорема об алгоритмической неразрешимости проблемы самоприменимости)

ЛОГИЧЕСКИЙ ТИП

1. Логический (булевский) тип. Высказывания в логике, запись условий в виде высказываний, простые логические выражения: константы, переменные и их описание, отношения, логические операции (not, and, or) и их свойства; логические выражения (синтаксис "на пальцах") и правила их вычисления; законы логики; оператор присваивания; ввод и вывод.

2. Логические операции над целыми числами; операции сдвига; стандартные функции odd, succ и pred; стандартные процедуры inc, dec.

СИМВОЛЬНЫЙ ТИП

Символьный тип (таблица ASCII, запись символов, набор символов и их упорядоченность, описание переменных, стандартные функции (ord, chr, succ, pred), символьные выражения, оператор присваивания; ввод и вывод).

Тема 2. Модельная ЭВМ. Алгоритмические языки. Метаязыки

МОДЕЛЬНАЯ ЭВМ

1. Структура ЭВМ (схема, устройства ввода и вывода, внутренняя и внешняя память, центральный процессор)

2. Оперативная память (ячейки, адреса ячеек, размер ячеек)

3. Машинное представление данных (представление целых чисел и вещественных чисел, точности представления целых и вещественных чисел, представление символьной информации)

4. Машинная программа (смысл и формат трехадресных команд, система команд)

5. Пример машинной программы

ОПЕРАТОРЫ. СТРУКТУРНОЕ ПРОГРАММИРОВАНИЕ

1. Классификация всех операторов. Пустой оператор, составной оператор. Условный оператор. Оператор перехода, метки, раздел меток. Операторы цикла: с предусловием (while), с постусловием (repeat), счетный цикл с параметром (for), ограничения на счетный цикл

2. Как сделать программу понятной: комментарии, структурная запись программы, структурное программирование

3. Датчик псевдослучайных чисел.

ПЕРЕЧИСЛИМЫЕ И ОГРАНИЧЕННЫЕ

1. Раздел типов (раздел описания типов, переименование типов, два варианта описания переменных нестандартных типов: с заданием имени для нового типа или с использованием безымянного типа)

2. Перечислимые типы (конструктор ПТ, нумерация и упорядоченность имен в ПТ, стандартные функции и операции для ПТ: ord, succ, pred, сравнение, присваивание, счетный цикл)

3. Ограниченные типы (особенности присваивание переменным ОТ, отличие этих переменных от переменных базового типа)

4. Оператор варианта (синтаксис и семантика оператора case)

АЛГОРИТМИЧЕСКИЕ ЯЗЫКИ. МЕТАЯЗЫКИ

1. Об алгоритмических языках (достоинства и недостатки машинного языка, алгоритмические языки высокого уровня и их достоинства, разнообразие АЯ, трансляция с АЯ)

2. Металингвистические формулы (БНФ) (основные понятия БНФ, рекурсивные формулы, расширение БНФ)

3. Синтаксические диаграммы (правила перевода БНФ синтаксические диаграммы)

4. Парадигмы программирования

РЕГУЛЯРНЫЕ ТИПЫ (МАССИВЫ)

1. Одномерные массивы (понятие массива, переменных с индексами, индексная арифметика, описание регулярных типов (РТ), присваивание массивов, об именной идентичности РТ)

2. Ввод и вывод массивов

3. Многомерные массивы (массив массивов, двумерный массив (матрица), доступ к строкам и элементам матриц)

СТРОКИ

1. Определение, тип строки, строки- константы, сравнение строк

2. Операции над строками

3. Ввод и вывод строк

Тема 3. Подпрограммы. Процедуры и функции. Рекурсия.

ПОДПРОГРАММЫ. ПРОЦЕДУРЫ И ФУНКЦИИ

1. Структурирование программы. Программирование сверху вниз (достоинства и недостатки)

2. Назначение процедур (формальные и фактические параметры, параметры-значения и параметры-переменные, передача параметров по значению и по ссылке)

3. Принцип локализации (понятие о вложенных блоках и области видимости имени/метки; принцип локализации, глобальные и локальные имена процедуры)

4. Побочные эффекты функций

ПРОГРАММИРОВАНИЕ СВЕРХУ ВНИЗ. ТЕСТИРОВАНИЕ. ОТЛАДКА

1. Описание процедур (синтаксис, назначение заголовка и тела, возможные сокращения в описании формальных параметров)

2. Функции (отличие функций от процедур, синтаксис описания функции и указателя функции)

3. Тестирование (правильная программа, понятие теста и тестирования программы, цель тестирования, методы тестирования)

3. Отладка

КОМБИНИРОВАННЫЕ ТИПЫ

1. Комбинированные типы (отличие записей от массивов, описание КТ и переменных-записей, присваивание записей, обозначение поля)

2. Оператор присоединения (простейший и общий вариант оператора, конфликт имен)

Лекция 6. РЕКУРСИЯ

1. Рекурсивные функции и процедуры (особенности вычисления рекурсивных функций и их описания)

2. Алгоритмы поиска (перебора) с возвратами и их рекурсивная реализация (задачи искусственного интеллекта)

3. Параметры-функции и параметры-

Процедуры

МНОЖЕСТВЕННЫЕ ТИПЫ

1. Множественные типы (особенности множеств в Паскале: конечность, однотипность,

описание МТ и переменных-множеств, оператор присваивания для множеств, операции над множествами: отношения, объединение, пересечение, разность)

2. Моделирование множества на базе массива

ФАЙЛОВЫЕ ТИПЫ

1. Файлы (базовый тип, маркер, буферная переменная, отличие от массивов, описание ФТ и переменных-файлов, запрет присваивания файлов, режимы работы с файлами, операции в режиме чтения: reset, eof, get, read и в режиме записи: rewrite, put, write)

2. Текстовые файлы (стандартный тип text, деление текстов на строки, понятие "конец строки", операции для работы со строками: eoln, readln, writeln)

3. Двоичные файлы

Тема 4. Итерация. Проектирование циклов и рекуррентные вычисления

1. Рекуррентные соотношения 1-го и 2-го порядка

2. Инвариант цикла

3. Итерация

ССЫЛОЧНЫЕ ТИПЫ.

1. Указатели (понятие ссылки и динамической переменной, графическое изображение ссылок, описание ссылочных типов и ссылочных переменных, пустая ссылка nil, операции над ссылками)

ЛИНЕЙНЫЕ ОДНОСВЯЗНЫЕ СПИСКИ.

1. Динамические структуры данных

2. Списки (графическое изображение, описание типа списков, проблема косвенной рекурсии в описании типов, линейный односвязный список)

3. Варианты списков (однонаправленные списки без заглавного и с заглавным звеном, циклические списки, двунаправленные списки): достоинства и недостатки

Тема 5. Последовательные таблицы. Поиск. Сортировка

ПОСЛЕДОВАТЕЛЬНЫЕ ТАБЛИЦЫ. ПОИСК

1. Таблицы (набор записей, операция поиска по ключу)

2. Оценка сложности алгоритмов (оценка в худшем случае и в среднем; оптимальный алгоритм поиска и его оценка в худшем случае)

3. Последовательные таблицы (неупорядоченные таблицы, поиск и вставка, оценки сложности) (упорядоченные таблицы, бинарный поиск и вставка, оценки сложности)

4. Прямой доступ и хеширование

СТЕК. ОЧЕРЕДЬ

1. Стек (определение, векторное представление стека и реализация операций, списковое представление и реализация операций)

2. Очередь (определение, списковое представление и реализация операций векторное представление и реализация операций)

ДВОИЧНЫЕ (БИНАРНЫЕ) ДЕРЕВЬЯ. ТАБЛИЦЫ -ДЕРЕВЬЯ

1. Определение, представление (понятия вершины, ветви, корня, листьев двоичные деревья и их описание в Паскале)

2. Обход дерева

3. Деревья поиска (сравнений) (определение, реализация операций построения дерева по заданной последовательности элементов, поиск, вставка и удаление)

4. AVL-деревья (понятия уровня вершины и высоты (глубины) дерева, определение AVL-дерева, дерева и высотой дерева Фибоначчи, алгоритм вставки в AVL-дерево).

ПОСЛЕДОВАТЕЛЬНЫЕ ТАБЛИЦЫ. СОРТИРОВКА

1. Сортировка: постановка задачи

2. Алгоритмы сортировки. Быстрая сортировка. Оценки методов сортировки по числу сравнений и перестановок

Практика 17. ПЕРЕМЕШАННЫЕ ТАБЛИЦЫ (ХЭШ-ТАБЛИЦЫ)

1. Основная идея (ключ --> адрес, функция расстановки (ФР), вставка в таблицу с учетом ФР)

2. Закрытое хеширование (метод линейных проб) (основной принцип, первичная и вторичная ФР, реализация операций вставки и поиска)

3. Открытое хеширование (метод цепочек) (основная идея, операции поиска и вставки, оценки)

4. Функции расстановки

3.3. Формы и методы проведения занятий, применяемые учебные технологии

При проведении занятий применяется игровое проектирование, компьютерная симуляция, дискуссия.

4. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

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

1. Подготовка к лабораторным занятиям.
2. Самостоятельное изучение отдельных вопросов в соответствии со структурой дисциплины, составление конспектов.
3. Самостоятельное выполнение лабораторных работ.
4. Подготовка к тестированию, аудиторной контрольной работе.
5. Выполнение домашних заданий.
6. Подготовка к промежуточной аттестации.

Содержание СРС

№	Наименование раздела (темы) дисциплины	Вид СРС	Трудоемкость (в часах)	Формы и методы контроля
1.	Алгоритмы. Машина Тьюринга. Нормальные алгоритмы Маркова	Проработка материала. Выполнение индивидуальных заданий	6	Оценка по БРС
2.	Модельная ЭВМ. Алгоритмические языки. Метаязыки	Проработка материала. Выполнение индивидуальных заданий	6	Оценка по БРС
3.	Подпрограммы. Процедуры и функции. Рекурсия	Проработка материала. Выполнение индивидуальных заданий	5	Оценка по БРС
4.	Итерация. Проектирование циклов и рекуррентные вычисления	Проработка материала. Выполнение индивидуальных заданий	5	Оценка по БРС
5.	Последовательные таблицы. Поиск. Сортировка	Проработка материала. Выполнение индивидуальных заданий	5	Оценка по БРС
	Всего часов		27	

5. Методические указания для обучающихся, по освоению дисциплины

Рейтинговый регламент по дисциплине:

Вид выполняемой учебной работы (контролирующие мероприятия)	Количество баллов (min)	Количество баллов (max)
Алгоритмы. Машина Тьюринга. Нормальные алгоритмы Маркова	9	14
Модельная ЭВМ. Алгоритмические языки. Метаязыки	9	14
Подпрограммы. Процедуры и функции. Рекурсия	9	14
Итерация. Проектирование циклов и рекуррентные вычисления	9	14
Последовательные таблицы. Поиск. Сортировка	9	14
Количество баллов для допуска к экзамену (min-max)	45	70

6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

6.1. Показатели, критерии и шкала оценивания

Коды оцениваемых компетенций	Индикаторы достижения компетенций	Показатель Оценивания (дескриптор) (по П.1.2.РПД)	Уровни освоения	Критерий оценивания	Оценка
<p>ОПК-2. Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач</p> <p>ОПК-5. Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения</p>	<p>ОПК-1.1 Знает: математические основы программирования и языков программирования, организации баз данных и компьютерного моделирования; математические методы оценки качества, надежности и эффективности программных продуктов; математические методы организации информационной безопасности при разработке и эксплуатации программных продуктов и программных комплексов.</p> <p>ОПК-1.2 Умеет использовать этот аппарат в профессиональной деятельности.</p> <p>ОПК-1.3 Имеет навыки</p>	<p>Знать общие черты и характеристики самых главных частей персонального компьютера;</p> <p>Уметь пользоваться операционной системой;</p> <p>Владеть (навыками) работы с операционной системой;</p> <p>Владеть (методиками) способами и средствами получения, хранения, передачи информации</p>	Высокий	Освоены все компетенции. Студент показывает отличные теоретические и практические знания по дисциплине.	отлично
			Базовый	Студент показывает хорошие знания по дисциплине. Может применять полученные знания.	хорошо

<p>применения данного математического аппарата при решении конкретных задач ОПК-5.1.</p> <p>Знает: методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий.</p> <p>ОПК-5.2.</p> <p>Умеет: применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий.</p> <p>ОПК-5.3.</p> <p>Владеет: навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач</p>			
	Минимальный	Студент показывает хорошие теоретические знания.	удовлетворительно
	Не освоены	Знания студента по дисциплине минимальны.	неудовлетворительно

6.2. Типовые контрольные задания (вопросы) для промежуточной аттестации

Коды оцениваемых компетенций	Оцениваемый показатель (ЗУВ)	Тема	Образец типового (тестового или практического) задания (вопроса)
ОПК-2. Способен использовать и адаптировать существующ	Знать общие черты и характеристики самых главных частей персонального компьютера;	Алгоритмы. Машина Тьюринга. Нормальные алгоритмы Маркова	Вариант письменного теста (образец) 1. Указать, что будет выведено на экран в результате выполнения следующих операторов программы (программа на языке Паскаль):

<p>ие математическ ие методы и системы программиро вания для разработки и реализации алгоритмов решения прикладных задач</p> <p>ОПК-5 Способен разрабатыват ь алгоритмы и компьютерны е программы, пригодные для практическог о применения</p>	<p>Уметь пользоваться операционной системой;</p> <p>Владеть (навыками) работы с операционной системой;</p> <p>Владеть (методиками) способами и средствами получения, хранения, передачи информации</p>	<p>Модельная ЭВМ. Алгоритмические языки. Метаязыки Подпрограммы. Процедуры и функции. Рекурсия</p> <p>Итерация. Проектирование циклов и рекуррентные вычисления</p> <p>Последовательные таблицы. Поиск. Сортировка</p>	<pre>procedure P(a: integer; var b: integer); begin a:=a+b; b:=b+c; c:=c+a end; var a, b, c: integer; a:=1; b:=2; c:=3; P(c,a); write(a:3, b:3, c:3);</pre> <p>2. type T=file of integer; Описать целочисленную функцию Nmax(f), которая за один просмотр непустого файла f типа T определяет, сколько раз в этот файл входит его максимальный элемент.</p> <p>3. Нарисовать AVL-дерево со следующими свойствами: а) высота дерева равна $h = 4$ (пояснение: высота пустого дерева равна 0); б) у всех вершин, кроме листьев, высота правого поддерева меньше высоты левого поддерева; в) ключи в вершинах – это целые числа от 1 до n, где n – число вершин в дереве.</p> <p>4. Описать алгоритм Маркова, применимый только к тем словам в алфавите {a,b}, которые задаются следующим определением <прим>: <прим> ::= a {a} a <прим> b и неприменимый к другим словам в данном алфавите. При этом алгоритм должен содержать не более четырех формул подстановки.</p> <p>Решения и критерии оценки задач для тематических задач варианта – образца письменного тестирования</p> <p>Каждое задание оценивается из 100 баллов. Всего можно получить 600 баллов. Оценка «Зачтено» выставляется за работу, набравшую более 300 баллов.</p> <p>1. Ответ: 1 5 6 Критерии: за каждое неправильное число, за непонимание параметра вывода вида x:n – снижать на 1/3</p> <p>2. function Nmax(var f: T): integer; var max, k, x: integer; begin reset(f); read(f,max); k:=1;</p>
--	---	--	---

			<pre> while not eof(f) do begin read(f,x); if x>max then begin max:=x; k:=1 end else if x=max then k:=k+1 end; Nmax:=k end; Критерии: ошибки в алгоритме поиска максимального значения (не умеет искать максимум), более одного просмотра файла – сразу минус за каждую другую ошибку при работе с параметрами и файлами (нет var перед f, нет reset и т.п.) – снижать на 1/3 </pre>
--	--	--	---

6.3. Методические материалы, определяющие процедуру оценивания

Промежуточный контроль является заключительным занятием по основным разделам программы в виде контрольной работы в виде практических задач.

Итоговый контроль проводится в виде экзамена. Студенты получают билеты с теоретическими и практическими заданиями.

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

№	Автор, название, место издания, издательство, год издания учебной литературы, вид и характеристика иных информационных ресурсов	Наличие грифа, вид грифа	МПТИ СВФУ, кафедральная библиотека и кол-во экземпляров	Электронные издания: точка доступа к ресурсу (наименование ЭБС, ЭБ СВФУ)
Основная литература				
1	Вирт Никлаус, Алгоритмы и структура данных, учебник, М.: ДМК Пресс, 2012		18	18
2	Токманцев Т.Б., Алгоритмические языки и программирование, учебное пособие, Екатеринбург: УФУ, 2013		18	http://www.iprbookshop.ru/68220.html
Дополнительная литература				
1	Игошин В.В., Задачи и упражнения по математической логике и теории алгоритмов, учебное пособие, М.: Академия, 2008	МО	18	
2	Канцедал С.А., Алгоритмизация и программирование, учебное пособие, М.: Форум: Инфра-м, 2013	МО	18	
3	Кнут Д.Э., Искусство программирования. Т.1 Основные алгоритмы, учебное пособие, М.: ИД Вильямс, 2012		18	
4	Златопольский Д.М., Программирование: типовые задачи, алгоритмы, методы, учебное пособие, М. : Лаборатория знаний, 2020		18	https://biblioclub.ru/index.php?page=book_red&id=222873&sr=1

8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее сеть-Интернет), необходимых для освоения дисциплины

1. ЭБС «Университетская библиотека онлайн» www.biblioclub.ru
2. ЭБС IPRbooks <http://www.iprbookshop.ru/>
3. Научная электронная библиотека <http://elibrary.ru>

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

Занятия проводятся в компьютерном классе, оборудованным ПК, интерактивной доской, специальным оборудованием для создания и воспроизведения мультимедиа.

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

10.1. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

При осуществлении образовательного процесса по дисциплине используются следующие информационные технологии:

- использование на занятиях электронных изданий;
- использование специализированных и офисных программ.

10.2. Перечень программного обеспечения

1. Borland Delphi 7.0;
2. Borland Pascal 7.0;
3. FreePascal 2.2.4;
4. Lazarus 0.9.22;
5. Эмулятор машины Тьюринга.

Не обязательно все из перечисленного. Версии могут быть старше. Возможны аналоги.

10.3. Перечень информационных справочных систем

Консультант, Гарант

