



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«Дальневосточный федеральный университет»
(ДВФУ)

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ (ШКОЛА)

«СОГЛАСОВАНО»

Руководитель ОП

 Пак Т.В.

«УТВЕРЖДАЮ»

Директор департамента Математического
и компьютерного моделирования
Сущенко А.А.

« 26 » января 2022 г.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Дополнительные главы теории алгоритмов

Направление подготовки 01.03.02 Прикладная математика и информатика

(Математические и компьютерные технологии)

Форма подготовки очная

курс 3 семестр б

лекции 00 час.

практические занятия 18 час.

лабораторные работы 00 час.

в том числе с использованием МАО

всего часов аудиторной нагрузки 18 час.

самостоятельная работа 18 час.

в том числе на подготовку к экзамену 00 час.

контрольные работы (количество) не предусмотрены

курсовая работа / курсовой проект не предусмотрены

зачет б семестр

экзамен не предусмотрен

Рабочая программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта по направлению подготовки 01.03.02 Прикладная математика и информатика, утвержденного приказом Министерства образования и науки Российской Федерации от 10 января 2018 г. № 9 (с изменениями и дополнениями)

Рабочая программа обсуждена на заседании департамента математического и компьютерного моделирования, протокол № 6 от «05» марта 2022 г.

Директор департамента математического и компьютерного моделирования Сущенко А. А.

Составитель (ли): ст.п. А.С. Кленин

Владивосток
2022

Оборотная сторона титульного листа РПД

I. Рабочая программа пересмотрена на заседании департамента:

Протокол от «_____» _____ 20__ г. № _____

Директор департамента _____
(подпись) (И.О. Фамилия)

II. Рабочая программа пересмотрена на заседании департамента:

Протокол от «_____» _____ 20__ г. № _____

Директор департамента _____
(подпись) (И.О. Фамилия)

III. Рабочая программа пересмотрена на заседании департамента:

Протокол от «_____» _____ 20__ г. № _____

Директор департамента _____
(подпись) (И.О. Фамилия)

IV. Рабочая программа пересмотрена на заседании департамента:

Протокол от «_____» _____ 20__ г. № _____

Директор департамента _____
(подпись) (И.О. Фамилия)

I. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ:

Цель: Получение дополнительных навыков программирования. Знакомство с технологическим циклом создания программного продукта и подготовка к решению прикладных задач программирования из любой предметной области с использованием любого подходящего языка программирования.

Задачи:

- получить представление об основах программирования и этапах решения задачи программирования;
- владеть языками программирования;
- знать стандартные алгоритмы, лежащие в основе решения задач программирования и уметь применять их на практике;
- овладеть практическими навыками решения задач, начиная от ее постановки и формализации и заканчивая отладкой и тестированием.
- научиться методам практической реализации программ на примере языков программирования PascalABC, C++.

В результате изучения данной дисциплины у обучающихся формируются следующие общепрофессиональные компетенции:

Наименование категории (группы) общепрофессиональных компетенций	Код и наименование общепрофессиональной компетенции (результат освоения)	Код и наименование индикатора достижения компетенции
Информационно-коммуникационные технологии для профессиональной деятельности	ОПК-5 Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности	ОПК-5.1 использует основные положения и концепции прикладного и системного программирования, архитектуры компьютеров и сетей (в том числе и глобальных)
		ОПК-5.2 решает стандартные задачи профессиональной деятельности с использованием современных информационных систем, сред и комплексов
		ОПК-5.3 использует технологии создания и эксплуатации программных продуктов и программных комплексов на основе информационной и цифровой культуры
	ОПК-6 Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК-6.1 выбирает современные технологии разработки алгоритмов и компьютерных программ для решения поставленных задач
ОПК-6.2 использует современные средства и языки программирования, современные программные среды разработки для решения прикладных задач различных классов		
ОПК-6.3 разрабатывает программное обеспечение, реализует мобильные, серверные приложения и другие информационно-коммуникационные сервисы с учетом основных требований информационной безопасности		

Код и наименование индикатора достижения компетенции	Наименование показателя оценивания (результата обучения по дисциплине)
	Знает современные информационно-коммуникационные технологии

Код и наименование индикатора достижения компетенции	Наименование показателя оценивания (результата обучения по дисциплине)
ОПК-5.1 использует основные положения и концепции прикладного и системного программирования, архитектуры компьютеров и сетей (в том числе и глобальных)	Умеет использовать современные информационно-коммуникационные технологии
	Владеет навыками использования современных информационно-коммуникационных технологий
ОПК-5.2 решает стандартные задачи профессиональной деятельности с использованием современных информационных систем, сред и комплексов	Знает новые решения в области наукоемких технологий и пакетов программ для решения прикладных задач профессиональной деятельности
	Умеет анализировать рынок новых решений в области наукоемких технологий и эксплуатации программных продуктов и программных комплексов на основе информационной и библиографической культуры
	Владеет навыками анализа рынка новых решений в области наукоемких технологий и пакетов программ для решения прикладных задач.
ОПК-5.3 использует технологии создания и эксплуатации программных продуктов и программных комплексов на основе информационной и цифровой культуры	Знает техническую и отчетную документацию разработки ПО
	Умеет формировать техническую и отчетную документацию и разрабатывать технические документы с учетом основных требований информационной безопасности
	Владеет навыками формирования технической отчетной документации мобильных, серверных приложений и другие информационно-коммуникационных сервисов
ОПК-6.1 выбирает современные технологии разработки алгоритмов и компьютерных программ для решения поставленных задач	Знает современные информационно-коммуникационные технологии
	Умеет использовать современные информационно-коммуникационные технологии
	Владеет навыками использования современных информационно-коммуникационных технологий
ОПК-6.2 использует современные средства и языки программирования, современные программные среды разработки для решения прикладных задач различных классов	Знает новые решения в области наукоемких технологий и пакетов программ для решения прикладных задач профессиональной деятельности
	Умеет анализировать рынок новых решений в области наукоемких технологий и эксплуатации программных продуктов и программных комплексов на основе информационной и библиографической культуры
	Владеет навыками анализа рынка новых решений в области наукоемких технологий и пакетов программ для решения прикладных задач
ОПК-6.3 разрабатывает программное обеспечение, реализует мобильные, серверные приложения и другие информационно-коммуникационные сервисы с учетом основных требований информационной безопасности.	Знает техническую и отчетную документацию разработки ПО
	Умеет формировать техническую и отчетную документацию и разрабатывать технические документы с учетом основных требований информационной безопасности
	Владеет навыками формирования технической отчетной документации мобильных, серверных приложений и другие информационно-коммуникационных сервисов

Для формирования вышеуказанных компетенций в рамках дисциплины «Алгоритмы и структуры данных» применяются следующие методы активного/ интерактивного обучения:

- презентации с использованием доски, книг, видео, слайдов, компьютеров и т.п., с последующим обсуждением материалов,
- обратная связь с формированием общего представления об уровне владения знаниями студентов, актуальными для занятия,
- разминка с вопросами, ориентированными на выстраивание логической цепочки из полученных знаний (конструирование нового знания),

- работа в малых группах (дает всем студентам возможность участвовать в работе, практиковать навыки сотрудничества, межличностного общения).

II. ТРУДОЁМКОСТЬ ДИСЦИПЛИНЫ И ВИДОВ УЧЕБНЫХ ЗАНЯТИЙ ПО ДИСЦИПЛИНЕ

Общая трудоемкость дисциплины составляет 10 зачётных единиц (360 академических часа).

(1 зачетная единица соответствует 36 академическим часам)

Видами учебных занятий и работы обучающегося по дисциплине могут являться:

Обозначение	Виды учебных занятий и работы обучающегося
Лек	Лекции
Лаб	Лабораторные работы
Пр	Практические занятия
ОК	Онлайн курс
СР	Самостоятельная работа обучающегося в период теоретического обучения
Контроль	Самостоятельная работа обучающегося и контактная работа обучающегося с преподавателем в период промежуточной аттестации

Структура дисциплины:

Форма обучения – очная.

№	Наименование раздела дисциплины	Семестр	Количество часов по видам учебных занятий и работы обучающегося					Формы промежуточной аттестации	
			Лек	Лаб	Пр	ОК	СР		Контроль
1	Раздел 1. Конечные автоматы на примере разбора текстовых цепочек	6			1				Зачет
2	Раздел 2. Длинная арифметика				2				
3	Раздел 3. Алгоритмы сжатия данных				2				
4	Раздел 4. Передача данных				1		18		
5	Раздел 5. Шифрование				2				
6	Раздел 6. Динамические структуры				2				
7	Раздел 7. Деревья				2				
8	Раздел 8. Графы				2				
9	Раздел 9. Хеширование				2				
10	Раздел 10. Введение в ООП				2				
Итого:					18		18		

III. СТРУКТУРА И СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА

Раздел 1. Конечные автоматы на примере разбора текстовых цепочек.

Раздел 2. Длинная арифметика.

- Ввод и вывод длинных чисел
- Хранение
- Операции сложения, сравнения, вычитания, умножения

Раздел 3. Алгоритмы сжатия данных.

- Равномерное кодирование
- RLE
- Код Шеннона-Фано
- Код Хаффмана
- Алгоритм Jpeg, Png

Раздел 4. Передача данных.

- Помехоустойчивый код, расстояние Хэмминга
- Бит четности, утроение бита
- Код Хэмминга

Раздел 5. Шифрование.

- Симметричный шифр
- Шифр Цезаря, Виженера
- Хэширование и пароли
- Алгоритм RSA
- Стеганография

Раздел 6. Динамические структуры.

- Связные списки
- Стек
- Очередь
- Кольцо
- Дек

Раздел 7. Деревья.

- Двоичные деревья. Представление.
- Прямой, обратный и симметричный обходы.
- Деревья двоичного поиска.
- Дерево арифметических выражений
- Сбалансированное дерево
- Добавление и удаление узлов

Раздел 8. Графы.

- Представление графов.
- Остовное дерево наименьшей стоимости.

- Метод поиска в глубину.
- Задача Штейнера.
- Алгоритм Беллмана-Форда.
- Алгоритм Дейкстры
- Алгоритм Прима Крускала
- Задача коммивояжера
- Паросочетания. Задачи о наибольшем паросочетании.
- Плоская укладка графа

Раздел 9. Хеширование.

Раздел 10. Введение в ООП.

IV. СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА И САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Лабораторная работа №1. Программа как единство алгоритма и структуры данных. Данные базового типа (целые, вещественные, символьные, логические). Структура хранения данных в ЭВМ. Определение типов и выражения с данными базового типа в языке Паскаль. Механизмы приведения типов.

Лабораторная работа №2. Операции и их приоритет в языке Паскаль. Преобразование типов. Перечислимый и диапазонный типы. Введение в систему конструируемых типов языка. Основные алгоритмические структуры. Операторы ветвления и цикла в языке Паскаль.

Лабораторная работа №3. Вычислительные задачи.

Точность вычислений. Погрешности. Решение уравнений методом перебора, деления отрезка пополам. Дискретизация. Вычисление длины кривой. Вычисление площади фигуры методами прямоугольников и трапеций.

Лабораторная работа №4. Массивы. Строковые структуры данных. Хранение и алгоритмы их обработки.

Лабораторная работа №5. Ввод\вывод данных. Файловая система. Логические и физические файлы. Общие процедуры работы с файлами. Типизированные и не типизированные файлы.

Лабораторная работа №6. Процедуры и функции. Механизм передачи параметров. Глобальные и локальные описания. Области объявления и области действия имен.

Лабораторная работа №7. Данные типа "запись". Механизм хранения и организации. Массив записей. Массив как компонента записи.

Лабораторная работа №8. Алгоритмы сортировки: включением, выбором, обменом, трехленточное слияние и др. Алгоритм двоичного поиска. Алгоритм

Ктута, Мориса, Пратта поиска подстроки в строке. Организация таблиц расстановки и функции расстановки (метод хеширования).

Лабораторная работа №9. Длинная арифметика

1. Ввод и вывод длинных чисел
2. Операции сложения, сравнения, вычитания, деления

Лабораторная работа №10. Динамическая память. Адреса и указатели. Объявление указателей. Выделение и освобождение динамической памяти. Анализ и управление состоянием кучи. Организация структур.

Лабораторная работа №11. Создание и просмотр динамического списка. Добавление и удаление элементов. Списки типа стек, очередь и основные операции с ними.

Лабораторная работа №12. Деревья. Основные понятия. Двоичные деревья. Идеально сбалансированные деревья. Упорядоченные деревья. Основные операции (обход дерева, добавление и удаление элемента). Сортировка с помощью дерева.

Лабораторная работа №13. Графы. Основные понятия. Ориентированные и неориентированные графы. Способы представления графов.

Лабораторная работа №14. Задачи на графах (Прима-Краскала, Штейнера, Дейкстры, коммивояжера).

Работа с памятью, в том числе с внешней. Алгоритм Дойча. Метод близнецов. Выделение памяти разного размера. Уплотнение памяти.

V. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ (И ОНЛАЙН КУРСА ПРИ НАЛИЧИИ)

Материалы лекций (включая авторские видеоматериалы), перечень заданий с описанием требований и формы выполнения, график работ и рейтинг студентов опубликованы на платформе **GOGLE DRIVE**. Для сдачи работ, их комментирования и оценивания используется функционал ресурса.

При необходимости, могут быть использованы платформы онлайн конференций **MS TEAMS** или **ZOOM**

Для изучения дисциплины приводится перечень рекомендуемой литературы, методические указания и вопросы к контрольным заданиям и экзамену.

В качестве основы для изучения дисциплины можно взять учебники, учебные пособия, электронные материалы и методические указания, приведенные в списке литературы.

Самостоятельная работа студентов состоит из подготовки к лабораторным работам в компьютерном классе, работы над рекомендованной литературой и текстами лекций в процессе изучения теоретического материала, выполнения заданий для самостоятельной работы.

План-график выполнения самостоятельной работы по дисциплине

№ п/п	Контролируемые разделы / темы дисциплины	Коды и этапы формирования компетенций		Оценочные средства	
				текущий контроль	промежуточная аттестация
1	Алгоритмизация и программирование	ОПК-5 ОПК-6	знает	Контрольная работа (КР-1)	Зачет
			умеет	Контрольная работа (КР-1)	
			владеет	Лабораторная работа (ЛР-1)	
2	Алгоритмы поиска и сортировки	ОПК-5 ОПК-6	знает	Коллоквиум (КР-1)	Зачет
			умеет	Коллоквиум (КР-2)	
			владеет	Лабораторная работа (ЛР-3)	
3	Динамические структуры	ОПК-5 ОПК-6	знает	Самостоятельная работа (ЛР-4)	Зачет
			умеет	Самостоятельная работа (КР-4)	
			владеет	Лабораторная работа (ЛР-4)	
4	Алгоритмы на графах	ОПК-5 ОПК-6	знает	Самостоятельная работа (ЛР-5)	Зачет
			умеет	Самостоятельная работа (ЛР-6)	
			владеет	Лабораторная работа (ЛР-6)	

Рекомендации по самостоятельной работе студентов

Самостоятельная работа включает изучение теоретического материала дисциплины и выполнение индивидуальных работ.

Материалы лекций (включая авторские видеоматериалы), перечень заданий с описанием требований и формы выполнения, график работ и рейтинг студентов опубликованы на платформе **GOGLE DRIVE**. Для сдачи работ, их комментирования и оценивания используется функционал ресурса, автоматические системы тестирования **CATS-DVFU, Informatics.msk.ru**

При необходимости, могут быть использованы платформы онлайн конференций **MS TEAMS** или **ZOOM**.

Для изучения дисциплины приводится перечень рекомендуемой литературы, методические указания и вопросы к контрольным заданиям и экзамену.

В качестве основы для изучения дисциплины можно взять учебники, учебные пособия, электронные материалы и методические указания, приведенные в списке литературы.

При изучении теоретического материала следует по методическим указаниям ознакомиться с планом темы. Освоив теоретический материал, необходимо самостоятельно, без помощи литературы, сделать попытку выполнить практическую работу. С каждой темой связан перечень ключевых

понятий. После изучения темы необходимо уметь самостоятельно давать определение понятий, иметь навыки и умения выполнять практические задания.

Требования к представлению и оформлению результатов самостоятельной работы

Результатом самостоятельной работы являются отчеты по лабораторным работам.

В процессе подготовки отчетов к лабораторным работам у студентов развиваются навыки составления письменной документации и систематизации имеющихся знаний. При составлении отчетов рекомендуется придерживаться следующей структуры:

1. Постановка задачи;
2. Метод решения;
3. Алгоритм метода;
4. Спецификация используемых функций и типов данных;
5. Описание тестов, на которых программа проходила проверку;

Критерии оценки выполнения самостоятельной работы

Отчет по лабораторной работе должен полностью удовлетворять условию задачи. В случае некачественно выполненных отчетов (не соответствующих заявленным требованиям) результирующий балл за работу может быть снижен. Студент должен продемонстрировать отчетливое и свободное владение концептуально-понятийным аппаратом, научным языком и терминологией. Наличие всех отчетов является допуском к экзамену.

На экзамене оценка «отлично» выставляется студенту, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал монографической литературы, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.

Оценка «хорошо» выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения.

Оценка «удовлетворительно» выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает

неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ.

Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.

Оценка по дисциплине может быть выставлена по результатам коллоквиумов и контрольных работ. При этом критерии оценки те же, что и на экзамене

VI. КОНТРОЛЬ ДОСТИЖЕНИЯ ЦЕЛЕЙ КУРСА

№ п/п	Контролируемые разделы / темы дисциплины	Код и наименование индикатора достижения	Оценочные средства		
				текущий контроль	промежуточная аттестация
1.	Алгоритмизация и программирование	ОПК-5 ОПК-6	знает	демонстрирует теоретическую подготовку	2 из 3 ответов правильны
			умеет	применяет знания на практике	решает в основном правильно практическое задание
			владеет	быстро выбирает метод решения	качественно выполняет задание аналитического характера
2.	Алгоритмы поиска и сортировки	ОПК-5 ОПК-6	знает	демонстрирует теоретическую подготовку	2 из 3 ответов правильны
			умеет	применяет знания на практике	решает в основном правильно практическое задание
			владеет	быстро выбирает метод решения	качественно выполняет задание аналитического характера
3.	Динамические структуры	ОПК-5 ОПК-6	знает	демонстрирует теоретическую подготовку	2 из 3 ответов правильны
			умеет	применяет знания на практике	решает в основном правильно практическое задание
			владеет	быстро выбирает метод решения	качественно выполняет задание аналитического характера
4.	Алгоритмы на графах	ОПК-5 ОПК-6	знает	демонстрирует теоретическую подготовку	2 из 3 ответов правильны
			умеет	применяет знания на практике	решает в основном правильно практическое задание
			владеет	быстро выбирает метод решения	качественно выполняет задание аналитического характера

Текущая аттестация студентов по дисциплине проводится в соответствии с локальными нормативными актами ДВФУ в форме коллоквиумов, контрольных и лабораторных работ по оцениванию

фактических результатов обучения студентов. Объектами оценивания выступают:

- учебная дисциплина (активность на занятиях, своевременность выполнения различных видов заданий, посещаемость всех видов занятий по аттестуемой дисциплине);
- степень усвоения теоретических знаний;
- уровень овладения практическими умениями и навыками;
- результаты самостоятельной работы.

Промежуточная аттестация студентов по дисциплине проводится в соответствии с локальными нормативными актами ДВФУ в виде экзамена в устной форме.

VII. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература

(электронные и печатные издания)

1. Семакин И.Г. Основы алгоритмизации и программирования: учебник для студ. учреждений сред. проф. образования М. Издательский центр «Академия», 2018. 304 с.
2. Основы алгоритмизации и программирования: Учебное пособие / В.Д. Колдаев; Под ред. Л.Г. Гагариной. - М.: ИД ФОРУМ: ИНФРА-М, 2015. - 416 с.: ил.; 60x90 1/16. - (Профессиональное образование). (переплет) ISBN 978-5-8199-0279-0 - Режим доступа: <http://znanium.com/catalog/product/484837>
3. Т.Кормен, Ч.Лейзерсон, Р.Ривест, К.Штайн - Алгоритмы. Построение и анализ. Издание 3-е (2013)
4. С.М. Окулов -Программирование в алгоритмах-БИНОМ. Лаборатория знаний (2014)
5. Искусство программирования для ЭВМ : пер. с англ. . т. 1 . Основные алгоритмы / Д. Кнут ; пер. Г. П. Бабенко, Ю. М. Баяковский. Кнут, Дональд Эрвин. Издание; 2-е изд. Место публикации; Москва Издатель; Вильямс Год; 2007. Физическое описание; 735 с.
6. [Искусство программирования : пер. с англ. . т. 3 . Сортировка и поиск / Д. Э. Кнут ; под общ. ред. Ю. В. Козаченко. Издание; 2-е изд. Место публикации; Москва Издатель; Вильямс Год; 2012. Физическое описание; 822 с.](#)
7. [С/С++. Программирование на языке высокого уровня :учебник / Т. А. Павловская. Санкт-Петербург : Питер, 2011. 461 с.](#)
8. [Паскаль : Программирование на языке высокого уровня : учебник для вузов / Т. А. Павловская. Санкт-Петербург : Питер, 2010.](#)

9. [C/C++. Структурное программирование. Практикум : учебное пособие / Т. А. Павловская, Ю. А. Щупак. Санкт-Петербург : Питер, 2010.](#)
10. Построение и анализ вычислительных алгоритмов : [монография] / А. Ахо, Дж. Хопкрофт, Дж. Ульман ; пер. с англ. А. О. Слисенко. Ахо, Альфред. Место публикации; Москва Издатель; Мир Год; 2012. Физическое описание; 536 с.
11. Кормен, Лейзерсон, Ривест: Алгоритмы. Построение и анализ / Издательство: Диалектика, 2019 г.
12. Алгоритмы и структуры данных с примерами на Паскале / Никлаус Вирт ; [пер. с англ. Д. Б. Подшивалова]. Вирт, Никлаус. Издание; [2-е изд.]. Место публикации; Санкт-Петербург Издатель; [Невский Диалект] Год; 2008. Физическое описание; 351 с.
13. Основы программирования : [учебник] / С. Окулов. Окулов, Станислав Михайлович. Издание; 4-е изд. Место публикации; Москва Издатель; Лаборатория знаний Год; 2008. Физическое описание; 440 с.

Дополнительная литература

1. Дж. Ахо, Алгоритмы и структуры данных
2. Н. Вирт, Алгоритмы и структуры данных
3. Н. Вирт, Систематическое программирование
4. Н. Вирт, Алгоритмы + структуры данных = программа
5. П. Холл, Вычислительные структуры
6. В. Бондарев, Основы программирования
7. [Ахо А., Хопкрофт Д., Ульман Д., Построение и анализ вычислительных алгоритмов.](#)
8. Дейкстра Э., Дисциплина программирования.
9. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ.
10. Скопин И.Н., Основы конструирования программ и языки программирования.

Перечень ресурсов информационно-телекоммуникационной сети

«Интернет»

1. Язык PascalABC.NET, Современное программирование на языке Паскаль, <http://pascalabc.net/>
2. Programming Taskbook, Электронный задачник по программированию, <http://www.ptaskbook.com/ru/>
3. Система автоматического тестирования программ, <http://imcs.dvfu.ru/cats/>

Перечень информационных технологий и программного обеспечения

1. PascalABC.NET + Microsoft .NET Framework v4.0 (Setup, 66 Mb)
2. Система программирования PascalABC.NET
3. Интегрированная среда разработки Visual Studio
4. Задачник РТ4
5. Платформа Microsoft .NET Framework v4.0
6. Russian Language Pack for .NET v4.0 (русификация сообщений об ошибках времени выполнения)
7. Framework Class Library Help (документация для всплывающей подсказки)
8. Автоматическая тестирующая система Московского центра непрерывного математического образования www.informatics.mccme.ru
9. Автоматическая тестирующая система ДВФУ <https://imcs.dvfu.ru/cats>.

VIII. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Все методические указания по типам задач размещены на ресурсе дисциплины и включают теоретическое описание проблемы и решений, содержание задач, пояснения к их выполнению и требования. Примеры заданий:

Задание **Моделирование стека.**

Цель работы: Отработка практических навыков по разработке основных этапов решения задачи моделирования стека.

Рекомендации к самостоятельной работе:

Изучить материал лекции «Понятие о структурах данных. Моделирование ряда структур данных: стека, очереди, списка».

Содержание работы:

1. Заполнить стек 10 случайными числами из интервала $[-10;20]$. Просмотреть содержимое стека. Найти сумму положительных чисел, хранящихся в стеке.
2. Сформировать стек из 5 чисел. Найти произведение 3-го и 4-го чисел из стека. Результат поместить в стек.
3. Заполнить стек 10 случайными числами из интервала $[-10;20]$. Найти максимальное число.

Форма представления отчета:

Отчет представить в письменном виде, который должен содержать: алгоритм в виде блок-схемы, программу и результат ее выполнения;

Задание Рекурсия.

Цель работы: изучить рекурсивные алгоритмы.

Рекомендации к самостоятельной работе:

Изучить материал лекции «Рекурсия».

Содержание работы:

1. Вычислить $(a! + b!)/a!$ используя рекурсивную функцию вычисления факториала
2. Вычислить $(1+2+3+4+5)/(1+2+3+4+5+6+7+8)$, используя рекурсивную функцию вычисления суммы первых n натуральных чисел.
3. Составить рекурсивную функцию вычисления n -го члена последовательности: $a_1 = 0, a_i = 2 * a_{i-1} + i$. Найти произведение 3-го и 7-го членов последовательности.
4. Составить рекурсивную функцию нахождения суммы n членов арифметической прогрессии 1, 3, ... Найти сумму с 5-го по 10-й членов прогрессии

Форма представления отчета:

Отчет представить в письменном виде, который должен содержать: алгоритм в виде блок-схемы, программу и результат ее выполнения.

IX. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Требования к аудиториям (помещениям, местам) для проведения занятий:

Лекционные аудитории и компьютерные классы для проведения лабораторных работ должны быть оснащенные мультимедийным оборудованием для проведения интерактивных занятий.

Подключение к сети Интернет в компьютерном классе – обязательно, в лекционной аудитории – желательно.

Требования к оборудованию рабочих мест преподавателя и обучающихся:

РМП: Мультимедийное оборудование.

РМО: компьютеры с аудио наушниками (в соответствии с наполняемостью подгрупп), подключенные к сети Интернет. Необходимо наличие общедоступного сетевого диска для обмена информацией.

В компьютерном классе должно быть установлено следующее программное обеспечение:

1. ОС Windows (не ниже 7);
2. среда программирования Pascal ABC;
3. MS Office 2010 и выше: Word, Excel, PowerPoint и др.;
4. проигрыватели мультимедийных файлов: FLV Player, KMPlayer, Windows Media Player и др.;

5. Web-браузеры: Mozilla Firefox, Opera и др. с поддержкой Flash и Java (TM).

Х. ФОНДЫ ОЦЕНОЧНЫХ СРЕДСТВ

Критерии выставления оценки студенту на экзамене

Оценка «отлично» выставляется студенту, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал монографической литературы, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.

Оценка «хорошо» выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения.

Оценка «удовлетворительно» выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ.

Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.

Оценка по дисциплине может быть выставлена по результатам коллоквиумов и контрольных работ. При этом критерии оценки те же, что и на экзамене.

Оценочные средства для промежуточной аттестации

Примеры зачетных промежуточных заданий.

Описать идею указанного алгоритма, дать оценку его алгоритмической сложности. Рассмотреть особенности реализации и область применимости:

- Описать функцию, которая вычисляет среднее арифметическое элементов

- непустого списка.
- Описать функцию, которая меняет местами первый и последний элементы
- непустого списка.
- Описать функцию, которая вставляет новый элемент перед каждым вхождением заданного элемента.
- Описать функцию, которая проверяет на равенство списки L1 и L2:
- Описать функцию, которая определяет, входит ли список L1 в список L2;
- Описать функцию, которая переносит в конец непустого списки L его первый
 - элемент;
- Описать функцию, которая переносит в начало непустого списка L его последний элемент;
- Описать функцию, которая копирует в список L за каждым вхождением заданного элемента все элементы списка L1.
- Описать процедуру, которая объединяет два упорядоченных по неубыванию
 - списка L1 и L2 в один упорядоченный по неубыванию список, сменив соответствующим образом ссылки в L1 и L2.
 - Описать функцию, которая проверяет, упорядочены ли элементы списка по алфавиту.
- сортировка бинарным деревом
- шейкерная сортировка
- циклическая перестановка списка сортировка
- сортировка стеком
- сортировка TimSort
- сортировка вставками
- сортировка выбором
- сортировка слияние
- сортировка с помощью очереди
- быстрая сортировка
- сортировка четн/нечетн
- сортировка JSORT
- цифровая (поразрядная) сортировка
- блочная сортировка
- сортировка Шелла

- поиск Армстронга,
- градиентный спуск,
- поразрядный поиск
- бинарный поиск,
- интерполяционный поиск,
- поиск подстроки в строке
- поиск Ахо-Карасик
- двоичный поиск,
- индексный поиск,
- случайный поиск
- линейный поиск,
- троичный поиск
- метод касательных,
- метод приближения,
- метод хорд
- алгоритм Эратосфен,
- последовательный поиск
- обезьяний поиск,
- алгоритм Дейкстры,
- поиск МиниМакс
- поиск Фиббоначчи,
- поиск по бору,
- поиск Мориса-Пратта
- Кеширование

Этапы выполнения работы.

1. Выполнить необходимую формализацию постановки задачи.
2. Провести исследование возможности применения известных алгоритмов к решению данной задачи.
3. Подготовить проверочные тесты.
4. Записать и отладить код программы. Провести тестирование.
5. Оформить отчет, содержащий основные результаты работы.

В письменном отчете должны содержаться:

1. Постановка задачи. Исходные данные.
2. Обоснование возможности применения данного алгоритма.
3. Программа, реализующая поставленную задачу.

Темы экзаменационных вопросов

1. Поиск подстроки в строке.
Префикс-функция
Алгоритм Кнута, Морриса, Пратта.
Алгоритм Бойера и Мура
2. Длинная арифметика
Ввод и вывод длинных чисел
Операции сложения, сравнения, вычитания, умножения
3. Конечные автоматы на примере разбора текстовых цепочек.
4. Алгоритмы сжатия данных
Равномерное кодирование
RLE
Код Шеннона-Фано
Код Хаффмана
Алгоритм Jpeg, Png
5. Передача данных
Помехоустойчивый код, расстояние Хэмминга
Бит четности, утроение бита
Код Хэмминга
6. Шифрование
Симметричный шифр
Шифр Цезаря, Виженера
Хэширование и пароли
Алгоритм RSA
Стеганография
7. Динамические структуры
Связные списки
Стек
Очередь
Кольцо
Дек
8. Деревья.
Двоичные деревья. Представление.
Прямой, обратный и симметричный обходы.
Деревья двоичного поиска.
Дерево арифметических выражений
Сбалансированное дерево
Добавление и удаление узлов
9. Хэширование.
10. Графы.

Алгоритмы на графах.
Остовное дерево наименьшей стоимости.
Метод поиска в глубину.
Алгоритм нахождения кратчайшего пути.
Задача Штейнера.
Паросочетания.

Пример тестов для промежуточной проверки

1. Задан массив $X[1..N]$. Какое условие надо поставить вместо многоточия, чтобы в результате в переменную i был записан номер элемента, равного R ? Вводите ответ без пробелов.

```
i:=1  
while (i<=N) and ( ... ) do  
i:=i+1;
```

Ответ:

2. Задан массив $X[1..N]$. Какое условие надо поставить вместо многоточия, чтобы найти минимальный элемент массива в переменной M ? Вводите ответ без пробелов.

```
M:=X[1];  
for k:=2 to N do  
if ... then M := X[k];
```

Ответ:

3. Задан массив $X[1..N]$. Какой оператор надо поставить вместо многоточия, чтобы найти номер максимального элемента массива в переменной M ? Вводите ответ без пробелов.

```
M:=1;  
for k:=2 to N do  
if X[k]>X[M] then
```

...

Ответ:

4. Требуется поменять местами значения элементов массива $X[a]$ и $X[b]$. Какой оператор нужно добавить вместо многоточия? В ответе не используйте пробелы.

```
p := X[a];  
...  
X[b] := p;
```

Ответ:

5. Задан массив $X[1..N]$. Какой оператор надо поставить вместо многоточия, чтобы в результате в переменную nR был записан номер элемента, равного R ? Вводите ответ без пробелов.

```

nR:= 0;
for k:=1 to N do
if X[k]=R then begin
...
break
end;

```

Ответ:

6. Задан массив $X[1..N]$. Что нужно поставить вместо многоточия, чтобы в результате элементы в массиве X были переставлены в обратном порядке?

```

for k:=1 to ... do begin
c:=X[k];
X[k]:=X[N+1-k];
X[N+1-k]:=c
end;

```

Ответ:

7. Задан массив $X[1..N]$. Какой оператор нужно поставить вместо многоточия, чтобы в результате элементы в массиве X были сдвинуты на один вправо (циклически)? В ответе не используйте пробелы.

```

c:=X[N];
for k:=N downto 2 do begin
...
end;
X[1]:=c;

```

Ответ:

8. Требуется выделить все отрицательные элементы массива $A[1..N]$ в начало массива $B[1..N]$.

Какой оператор надо вставить в программу вместо многоточия? Вводите ответ без пробелов.

```

c:=1;
for k:=1 to N do
if A[k]<0 then begin
...
c:=c+1
end;

```

Ответ:

9. Что будет выведено на экран после выполнения программы?

```

s := '123';
s := s + '0' + s;
s := s + s;
writeln ( s );

```

Ответ:

10. Что будет выведено на экран после выполнения

этой программы?

```
s := '123';  
s := s + s[2] + s + s[3];  
writeln ( s );
```

Ответ:

11. Что будет выведено на экран после выполнения этой программы?

```
s := '12345';  
n := Length(s) +  
Length('456');  
writeln ( n );
```

Ответ:

12. Что будет выведено на экран после выполнения программы?

```
s := '123456789';  
s1 := 'abcdef';  
s := Copy(s,5,2) + '0' +  
Copy(s1,3,3);  
writeln ( s );
```

Ответ:

13. Что будет выведено на экран после выполнения этой программы?

```
s := '123456';  
q := 'abc';  
Delete ( s, 2, 3 );  
Insert ( s, q, 2 );  
writeln ( q );
```

Ответ:

14. Что будет выведено на экран после окончания работы программы?

```
s := '123456789';  
Delete ( s, 1, 2 );  
n := Pos ( '6', s );  
writeln ( n );
```

Ответ:

15. Что будет выведено на экран после выполнения этой программы?

```
s := '123456789';  
n := Pos ( '456', s );  
Delete ( s, n+2, 3 );  
writeln ( s );
```

Ответ:

16. Сколько единиц будет выведено на экран?

```
s := '123456';  
while Length(s) > 0 do begin
```

```
Delete ( s, 1, 2 );  
writeln ( '#12#');  
end;
```

Ответ:

17. Программа должна считать, сколько раз в строке S встречается подстрока '#12#'. Какой оператор надо вставить вместо многоточия? Вводите ответ без пробелов.

```
c := 0;  
repeat  
...  
if p <> 0 then begin  
c := c + 1;  
Delete ( S, 1, p+1)  
end;  
until p = 0;
```

Ответ:

18. Программа должна считать, сколько раз в строке S встречается цифра '#0#'. Какой оператор надо вставить вместо многоточия? Вводите ответ без пробелов.

```
c := 0;  
for k:=1 to Length(S) do begin  
if S[k] = '#0#'; then  
...  
end;
```

Ответ:

19. Какой оператор надо вставить вместо многоточия, чтобы в строке S были записаны символы строки [tt]Q[/] в обратном порядке? Вводите ответ без пробелов.

```
Q := '#123456789#';  
S := '#1#';  
for k:=2 to 9 do begin  
...  
end;
```

Ответ:

20. Задан массив X[1..N]. Определите число операций сложения, которые выполняются при работе этой программы:

```
S:=X[1]+X[N];  
for k:=1 to N do  
X[k]:=X[k]+X[k]+S;  
Для обозначения операции умножения используйте символ *.
```

Ответ:

21. Задан массив $X[1..N]$. Определите число операций умножения, которые выполняются при работе этой программы:

```
S:=X[1]*X[N];  
for k:=1 to N do begin  
  X[k]:=2*X[k]+S;  
  for i:=1 to 3 do  
    S:=S*2;  
  end;
```

Для обозначения операции умножения используйте символ $*$.

Ответ:

22. Задан массив $X[1..N]$. Определите число операций сложения, которые выполняются при работе этой программы:

```
S:=X[1]+X[N]+3;  
for k:=1 to N do  
  for m:=1 to N do  
    X[k]:=X[k]+S;
```

Для обозначения операции умножения используйте символ $*$.

Ответ: