

Федеральное государственное бюджетное образовательное учреждение высшего образования «Тамбовский государственный университет имени Г.Р. Державина»
Институт математики, физики и информационных технологий
Кафедра математического моделирования и информационных технологий

УТВЕРЖДАЮ:
Директор института



Н. Л. Королева
«05» июля 2021 г.

РАБОЧАЯ ПРОГРАММА

по дисциплине Б1.О.10 Алгоритмизация и программирование

Направление подготовки/специальность: 09.03.03 - Прикладная информатика

Профиль/направленность/специализация: Прикладная информатика в
информационной сфере

Уровень высшего образования: бакалавриат

Квалификация: Бакалавр

год набора: 2021

Автор программы:

Кандидат педагогических наук, доцент Клыгина Елена Владимировна

Рабочая программа составлена в соответствии с ФГОС ВО по направлению подготовки 09.03.03 - Прикладная информатика (уровень бакалавриата) (приказ Министерства образования и науки РФ от «19» сентября 2017 г. № 922).

Рабочая программа принята на заседании Кафедры математического моделирования и информационных технологий «18» мая 2021 г. Протокол № 9

Рассмотрена и одобрена на заседании Ученого совета Института математики, физики и информационных технологий, Протокол от «05» июля 2021 г. № 5.

СОДЕРЖАНИЕ

1. Цели и задачи дисциплины.....	4
2. Место дисциплины в структуре ОП бакалавра.....	4
3. Объем и содержание дисциплины.....	4
4. Контроль знаний обучающихся и типовые оценочные средства.....	12
5. Методические указания для обучающихся по освоению дисциплины (модуля).....	54
6. Учебно-методическое и информационное обеспечение дисциплины.....	56
7. Материально-техническое обеспечение дисциплины, программное обеспечение, профессиональные базы данных и информационные справочные системы.....	56

1. Цели и задачи дисциплины

1.1 Цель дисциплины – формирование компетенций:

ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения

1.2 Типы задач профессиональной деятельности, к которым готовятся обучающиеся в рамках освоения дисциплины:

- научно-исследовательский
- проектный

1.3 Дисциплина ориентирована на подготовку обучающихся к профессиональной деятельности в сфере: 06 Связь, информационные и коммуникационные технологии (в сфере проектирования, разработки, внедрения и эксплуатации информационных систем, управления их жизненным циклом)

1.4 В результате освоения дисциплины у обучающихся должны быть сформированы:

Обобщенные трудовые функции / трудовые функции / трудовые или профессиональные действия (при наличии профстандарта)	Код и наименование компетенции ФГОС ВО, необходимой для формирования трудового или профессионального действия	Индикаторы достижения компетенций
	ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	Имеет адекватное представление об основных алгоритмах и структурах данных, используемых для разработки программ

1.5 Согласование междисциплинарных связей дисциплин, обеспечивающих освоение компетенций:

ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения

№ п/п	Наименование дисциплин, определяющих междисциплинарные связи	Форма обучения							
		Очная (семестр)				Заочная (семестр)			
		2	3	6	8	2	3	6	9
1	Научно-исследовательская работа				+				+
2	Объектно-ориентированное программирование		+				+		
3	Ознакомительная практика			+				+	
4	Программирование на Python	+				+			

2. Место дисциплины в структуре ОП бакалавриата:

Дисциплина «Алгоритмизация и программирование» относится к обязательной части учебного плана ОП по направлению подготовки 09.03.03 - Прикладная информатика.

Дисциплина «Алгоритмизация и программирование» изучается в 1, 2 семестрах.

3. Объем и содержание дисциплины

3.1. Объем дисциплины: 10 з.е.

Очная: 10 з.е.

Заочная: 10 з.е.

Вид учебной работы	Очная (всего часов)	Заочная (всего часов)
Общая трудоёмкость дисциплины	360	360
Контактная работа	224	36
Лекции (Лекции)	64	12
Лабораторные (Лаб. раб.)	160	24
Самостоятельная работа (СР)	100	315
Экзамен	36	9

3.2.Содержание курса:

№ темы	Название раздела/темы	Вид учебной работы, час.						Формы текущего контроля
		Лекции		Лаб. раб.		СР		
		О	З	О	З	О	З	
1 семестр								
1	Технология разработки алгоритмов. Решение задач на ЭВМ	6	1	14	2	8	49	Выполнение практических заданий; Опрос
2	Основы программирования на языке Pascal	6	1	16	2	18	49	Выполнение практических заданий; Выполнение практических заданий - контрольный срез
3	Структуры данных в языке Pascal	12	2	28	4	22	36	Выполнение практических заданий; Тестирование; Выполнение практических заданий - контрольный срез
4	Алгоритмы поиска и простой сортировки	8	2	22	4	20	28	Выполнение практических заданий; Тестирование
2 семестр								
5	Рекурсивные вычисления	4	1	10	2	6	19	Опрос; Выполнение практических заданий
6	Основы визуального программирования в среде Delphi	6	1	12	2	4	20	Опрос; Выполнение практических заданий

7	Работа с графикой в среде Delphi	4	1	16	2	4	24	Выполнение практических заданий; Выполнение практических заданий - контрольный срез
8	Указатели. Динамическая память	6	1	16	2	6	20	Опрос; Выполнение практических заданий
9	Динамические структуры данных	10	1	18	2	6	28	Выполнение практических заданий; Тестирование; Выполнение практических заданий - контрольный срез
10	Модули	2	1	8	2	6	42	Выполнение практических заданий

Тема 1. Технология разработки алгоритмов. Решение задач на ЭВМ (ОПК-7)

Лекция.

Алгоритм, исполнитель. Система команд исполнителя. Основные свойства алгоритма. Основные алгоритмические структуры: линейные, разветвляющиеся, циклические, вспомогательные алгоритмы. Данные, типы данных. Примеры. Этапы решения задач на ЭВМ. Принципы разработки алгоритмов и программ. Технология структурного программирования: модульный подход, программирование сверху-вниз, пошаговая детализация. Техника редактирования и тестирования алгоритмов и программ. Устранение синтаксических ошибок. Тестирование программ и устранение алгоритмических ошибок.

Лабораторные работы.

Практическое занятие 1. Технология разработки алгоритмов. Моделирование задач и разработка линейных и разветвляющихся алгоритмов. Решение задач.

Практическое занятие 2. Технология разработки циклических алгоритмов. Решение задач.

Практическое занятие 3. Использование вспомогательных алгоритмов при решении задач.

Задания для самостоятельной работы.

1. Разработать алгоритм вычисления площади треугольника по формуле Герона.
2. С начала суток прошло N секунд (N задается пользователем). Определить сколько полных секунд прошло с начала очередной минуты.
3. Дано четырехзначное число. Найти число, образуемое при перестановке первой и второй, третьей и четвертой цифр заданного числа. Например, из числа 5434 получить 4543, из числа 7048 – число 784.
4. Дано четырехзначное число. Определить, кратно ли четырем произведение его цифр.
5. Разработать алгоритм подсчета суммы цифр n-значного числа.
6. Дано натуральное число A ($A < 9999$). Разработать алгоритм, проверяющий, различны ли все четыре цифры этого числа (с учетом четырех цифр). Например, в числе 3678 все цифры различны, в числе 0023 – нет.
7. Разработать алгоритм вычисления $S_n = 1*2 + 2*3 + 4 + \dots + n*(n+1) + \dots + 2n$, где n натуральное число.

Тема 2. Основы программирования на языке Pascal (ОПК-7)

Лекция.

Структура программы. Область действия описаний. Имена. Величины. Простейшие типы данных: целый, вещественный, символьный, логический и их представление в ЭВМ, организация ввода и вывода, форматный и бесформатный ввод/вывод. Выражения. Простейшие алгоритмы обработки данных: вычисления по формулам, рекуррентные вычисления. Основные операторы языка программирования Pascal: присвоения, ввода, вывода, комментария, условный оператор, оператор выбора, операторы цикла. Синтаксис, свойства и область применения функций и процедур. Примеры.

Лабораторные работы.

1. Работа в консольном приложении TurboDelphi. Набор и редактирование программ. Разработка и реализация простейших программ использованием стандартных функции и операции над числовыми данными при разработке линейных программ.
2. Работа в консольном приложении TurboDelphi. Операторы IF - THEN – ELSE; CASE. Решение задач на использование команд ветвления.
3. Работа в консольном приложении TurboDelphi. Циклы FOR, WHILE, REPEAT. Решение задач с использованием команд ветвления и циклов.
4. Работа в консольном приложении TurboDelphi. Процедуры и функции. Разработка программ с использованием процедур и функций. Решение задач.

Задания для самостоятельной работы.

1. Составить и выполнить программу, которая вычисляет периметр треугольника, заданного координатами своих вершин (X1, Y1), (X2, Y2), (X3, Y3).
2. Составить и выполнить программу, которая определяет, равен ли квадрат заданного трехзначного числа кубу суммы цифр этого числа.
3. Составить и выполнить программу, которая выводит True или False в зависимости от того, имеют три заданных целых числа одинаковую четность или нет.
4. Составить и выполнить программу, которая находит среди данных трех чисел среднее (под средним числом будем понимать число, которое больше наименьшего, но меньше наибольшего).
5. Составить и выполнить программу вывода на экран номера четверти, которой принадлежит точка с координатами (X,Y) при условии, что X и Y отличны от 0.
6. Банк предлагает 3 вида срочных вкладов: на 3 месяца под p1 процентов годовых, на 6 месяцев – под p2 и на год – под p3 процентов. Напишите программу, которая определяет, какой из вкладов наиболее выгоден для вкладчика.
7. Составить и выполнить программу, которая выводит на экран K-ую строку треугольника Паскаля.
8. Старинная задача. Сколько можно купить быков, коров и телят, если плата за быка 10000 рублей, за корову 5000 рублей, за теленка – 500 рублей, если на 100000 рублей надо купить 100 голов скота (в решении задачи использовать не более двух циклов).
9. Составить и выполнить программу, которая печатает все совершенные числа, меньшие данного X.
10. Составить и выполнить программу, которая вычисляет площадь выпуклого четырехугольника.
11. Составить и выполнить программу, которая вычисляет площадь выпуклого n-угольника с помощью формулы Герона. Известны все стороны и диагонали, выходящие из одной вершины.
12. Составить и выполнить программу, которая упорядочивает значение трех переменных, используя подпрограмму упорядочивания двух чисел.
13. Составить и выполнить программу, которая содержит функцию возведения в степень.
14. Составить и выполнить программу, которая определяет, в каком из двух заданных чисел больше цифр (использовать подпрограмму, подсчитывающую количество цифр числа).

Тема 3. Структуры данных в языке Pascal (ОПК-7)

Лекция.

Структуры данных: понятие, классификация, общая характеристика. Примеры. Способы описания стандартных массивов. Расположение массивов в памяти. Преимущества и недостатки использования стандартных массивов. Строковый тип в языке Pascal. Стандартные подпрограммы обработки строк. Концепция программируемых типов данных: перечислимый тип, ограниченный тип. Тип данных запись. Организация данных в форме записи. Основные принципы работы с записями. Тип данных множества. Основные процедуры и функции работы с множествами. Типизированные константы. Файловый тип. Числовые файлы. Текстовые файлы. Файлы записей. Создание файлов. Чтение файлов. Обработка файлов. Примеры.

Лабораторные работы.

1. Работа в консольном приложении TurboDelphi. Ограниченный и перечислимый типы данных.
2. Работа в консольном приложении TurboDelphi. Массивы в языке Pascal. Решение задач с использованием массивов.
3. Работа в консольном приложении TurboDelphi. Работа со строковыми величинами.
4. Работа в консольном приложении TurboDelphi. Работа с множествами.
5. Работа в консольном приложении TurboDelphi. Работа с комбинированным типом (записи).
6. Работа в консольном приложении TurboDelphi. Работа с файлами.

Задания для самостоятельной работы.

1. Составить и выполнить программу, которая удаляет все повторяющиеся элементы, оставляя только их первые вхождения (получить массив различных элементов).
2. Составить и выполнить программу, которая упорядочивает элементы массива по возрастанию.
3. На столе в двух столбиках лежат 64 золотые и 64 серебряные монеты соответственно. Как серебряные, так и золотые монеты упорядочены в порядке убывания масс (самая тяжелая – сверху, самая легкая – внизу). Массы всех монет разные. Составить и выполнить программу, которая определит наименьшее количество взвешиваний, необходимых для определения 64-й монеты в порядке убывания масс среди всех 128 монет (за один раз можно взвешивать две монеты и определять, которая из них тяжелее).
4. Подсчитать сумму цифр, встречающихся в строке.
5. Составить и выполнить программу, которая для двух данных слов А и В определяет, можно или нет из букв слова А составить слово В.
6. Составить и выполнить программу, которая шифрует текстовое сообщение. Можно использовать такой способ шифрования. Шифровальщик задает ключ шифровки – целое число, который определяет величину смещения букв русского алфавита. Например, если ключ задан равным 3, то в тексте буква «а» заменяется на букву «г» и т.д. Использовать все буквы русского алфавита.
7. Составить и выполнить программу, которая для заданного натурального числа К печатает все цифры, не входящие в его десятичную запись, в порядке возрастания.
8. Составить и выполнить программу, которая подсчитывает сколько есть чисел Фибоначчи среди заданных ста целых чисел, значения которых от 1 до 50.
9. Составить и выполнить программу, которая по заданному списку N студентов (фамилия и оценки, полученные по двум экзаменам) формирует новый список студентов, каждая строка которого содержит фамилию и средний балл каждого, а также средний балл группы.
10. Составить и выполнить программу, которая определяет площадь различных геометрических фигур: окружности (по ее радиусу), квадрата (по его стороне), прямоугольника (по двум сторонам), треугольника (по трем сторонам).
11. Разбить данный файл на два, записав в первый из них положительные, а в другой - все остальные числа.
12. Проверить, является ли файл целых чисел упорядоченным по возрастанию.
13. Дан текстовый файл. Найти самую "длинную" строку в текстовом файле.
14. Дан текстовый файл. Найти среднее число букв в одном слове текста.
15. Дан текстовый файл. Сколько раз каждый знак встречается в данном тексте.

Лекция.

Сортировка как метод переработки информации. Сортировка посредством выбора, вставками (простые вставки, метод Шелла), обменом (методы пузырька, модифицированного пузырька). Методы поиска. Поиск минимального (максимального) элемента. Поиск элемента с заданным значением. Последовательный и бинарный поиск. Примеры.

Лабораторные работы.

1. Работа в консольном приложении TurboDelphi. Организация поиска данных.
2. Работа в консольном приложении TurboDelphi. Организация сортировки данных.

Задания для самостоятельной работы.

1. Дан двумерный массив $[N:N]$. Упорядочить элементы каждого столбца по возрастанию.
2. Дан двумерный массив $[N:N]$. Определить и вывести на печать элементы массива в каждой строке, которые больше суммы элементов главной диагонали.

Тема 5. Рекурсивные вычисления (ОПК-7)

Лекция.

Рекурсивные определения. Свойства рекурсивных алгоритмов. Формы рекурсивных записей. Рекурсивные и итерационные алгоритмы обработки данных. Условия, обеспечивающие завершение последовательности рекурсивных вызовов. Идеи реализации рекурсивных вызовов в подпрограммах. Взаимосвязь итерации и рекурсии. Быстрая сортировка. Стековый калькулятор. Задача о Ханойских башнях.

Лабораторные работы.

Практическое занятие 1. Разработка и реализация рекурсивных программ. Решение задач с использованием рекурсии.

Практическое занятие 2. Разработка и реализация рекурсивных программ. Стековый калькулятор. Задача о Ханойских башнях.

Практическое занятие 3. Использование рекурсии.

Задания для самостоятельной работы.

1. Написать подпрограмму записи десятичного числа в двоичном представлении.
2. Написать подпрограмму записи числа в обратном порядке.
3. Написать подпрограмму вычисления значения функции $f(n)$, определенной следующим образом: $f(1)=3$, $f(2)=1$, $f(n)=2*f(n-1)-f(n-2)$, если $n>2$.

Тема 6. Основы визуального программирования в среде Delphi (ОПК-7)

Лекция.

Принципы визуального программирования. Основы работы в среде Delphi. Проекты Delphi. Организация проекта в Delphi, основные файлы проектов. Структура модулей. Заголовок модуля и связь модулей друг с другом. Интерфейсная часть. Исполняемая часть. Инициализирующая и завершающая части. Структура файла проекта. Управление проектом. Создание и сохранение нового проекта. Структура приложения. Общая характеристика визуальных компонентов. Основные свойства формы, типы форм и их установка. Простейшие визуальные компоненты (текстовые, кнопки, списки, индикаторы, группирующие компоненты, компоненты прокрутки и т.д.): свойства, методы, события. Применение диалоговых компонентов.

Лабораторные работы.

Лабораторная работа 1. Разработка приложений с использованием простейших компонентов ввода-вывода, управления (Edit, Label, StringGrid, Memo, Button, RadioGroup, RadioButton).

Разработка простейших приложений.

Лабораторная работа 2. Основы работы в визуальной среде программирования Delphi. Создание интерфейса программы. Элементы управления для выбора из списков. (ListBox, ComboBox).

Элементы UpDown, DateTimePicker и MonthCalendar.

Лабораторная работа 3. Основы работы в визуальной среде программирования Delphi. Разработка приложений. Диалоговые компоненты (OpenDialog, SaveDialog, FontDialog, PrintDialog, MainMenu, PopupMenu).

Лабораторная работа 4. Основы работы в визуальной среде программирования Delphi. Разработка приложений с усложнённым интерфейсом. Многооконные приложения. Модальные и немодальные окна.

Лабораторная работа 5. Основы работы в визуальной среде программирования Delphi. Работа с массивами.

Лабораторная работа 6. Основы работы в визуальной среде программирования Delphi. Работа с числовыми, текстовыми и файлами записей.

Задания для самостоятельной работы.

1. Разработать приложение «Калькулятор для научных вычислений».
2. Организация форм для регистрации пользователя программы с сохранением данных во внешнем файле.
3. Разработать приложение, с помощью которого можно отредактировать текст, задать параметры шрифта для отдельно выделяемых фрагментов, сохранить в файле на диске.

Тема 7. Работа с графикой в среде Delphi (ОПК-7)

Лекция.

Обработка графики. Холст. Карандаш и кисть. Карандаш. Кисть. Вывод текста. Методы вычерчивания графических примитивов. Линия. Ломаная линия. Окружность и эллипс. Дуга. Прямоугольник. Многоугольник. Сектор. Точка. Вывод иллюстраций. Создание анимации. Метод базовой точки. Построение графиков функций. Использование рекурсии в графических построениях.

Лабораторные работы.

Лабораторная работа 1. Разработка приложений с использованием графических возможностей Delphi.

Лабораторная работа 2. Построение графиков функций.

Лабораторная работа 3. Построение анимационных изображений.

Лабораторная работа 4. Использование рекурсии в графических построениях.

Задания для самостоятельной работы.

1. Разработать приложение демонстрации летящего шара (круг), разбрасывающего искры (разноцветные точки).
2. Разработать приложение, при запуске которого на форме рисуется круг и при нажатии на кнопку круг передвигается вправо на 5-7 миллиметров. При решении задачи использовать метод базовой точки.
3. Разработать рекурсивную программу построения кривой Леви.
4. Создать простейший графический редактор, позволяющий рисовать фигуры, а также сохранять и открывать файлы типа *.bmp.

Тема 8. Указатели. Динамическая память (ОПК-7)

Лекция.

Указатель. Типизированные и нетипизированные указатели. Работа с динамической памятью. Достоинства и недостатки динамической организации памяти. Основные процедуры и функции работы с динамическими переменными. Способы описания динамических массивов. Расположение массивов в памяти. Основные принципы работы с динамическими массивами. Примеры.

Лабораторные работы.

Лабораторная работа 1. Работа с указателями. Распределение памяти. Динамическое выделение памяти.

Лабораторная работа 2. Работа с динамическими массивами.

Задания для самостоятельной работы.

1. Дана последовательность целых чисел, количество которых заранее не определено. Считать концом последовательности ввод нулевого значения. Ввести и напечатать на разных строках четные и нечетные числа данной последовательности.
2. Создать динамический одномерный массив, заполнить его случайными числами и вывести на печать. Добавить после каждого четного элемента элемент со значением 0.
3. Создать динамический двумерный массив, заполнить его случайными числами и вывести на печать. Добавить K строк в начало матрицы.
4. Дана последовательность символов латинского алфавита, количество которых заранее не определено. Считать концом последовательности символ пробела. Вывести на печать гласные буквы из данной последовательности.

Тема 9. Динамические структуры данных (ОПК-7)

Лекция.

Динамические структуры данных. Организация линейных списков. Линейные списки. Построение списка. Задача поиска элемента в списке. Задача обхода списка. Задача сортировки списка. Рекурсивные алгоритмы работы со списками. Освобождение памяти, занятой элементами списка, возвращение памяти системе. Стеки, деки, очереди. Моделирование структур данных. Реализация стеков на базе массива и списка. Работа со структурами данных. Примеры.

Лабораторные работы.

Лабораторная работа 1. Работа с динамическими структурами данных: списки.

Цель: научиться разрабатывать программы создания и обработки динамической структуры данных списки.

Лабораторная работа 2. Работа с динамическими структурами данных: стеки.

Цель: научиться разрабатывать программы создания и обработки динамической структуры данных стеки.

Лабораторная работа 3. Работа с динамическими структурами данных: очередь.

Цель: научиться разрабатывать программы создания и обработки динамической структуры данных очередь.

Лабораторная работа 4. Работа с динамическими структурами данных: дек.

Цель: научиться разрабатывать программы создания и обработки динамической структуры данных дек.

Задания для самостоятельной работы.

1. Разработать функцию нахождения среднего арифметического элементов непустого списка.
2. Разработать рекурсивную функцию проверки наличия в списке заданного значения.
3. Разработать процедуру перестановки первого и последнего элементов непустого списка.
4. Разработать процедуру удаления из списка второго по величине элемента, если такой есть.
5. Примеры использования структуры данных «стек».
6. Примеры использования структуры данных «очередь».

Тема 10. Модули (ОПК-7)

Лекция.

Структура модулей. Заголовок модуля и связь модулей друг с другом. Интерфейсная часть. Исполняемая часть. Инициализирующая и завершающие части. Доступ к объединенным в модуле объектам. Типы модулей в Delphi: модули данных, модули динамических объектов, пакеты, модули потока команд.

Лабораторные работы.

Лабораторная работа 1. Модули. Структура модуля.

Лабораторная работа 2. Стандартные модули Delphi.

Задания для самостоятельной работы.

1. Разработать модуль для работы с комплексными числами.

2. Разработать модуль с типами и функциями для вычисления площади и периметра трапеции.
3. Разработать модуль с типами и функциями для вычисления площади и длины окружности.
4. Разработать модуль с типами и функциями для вычисления площади поверхности и объема шара.
5. Разработать модуль для вычисления площади и периметра многоугольника, заданного координатами вершин.

4. Контроль знаний обучающихся и типовые оценочные средства

4.1. Распределение баллов:

1 семестр

- текущий контроль – 80 баллов
- контрольные срезы – 2 среза по 10 баллов каждый
- премиальные баллы – 10 баллов

Распределение баллов по заданиям:

№ те мы	Название темы / вид учебной работы	Формы текущего контроля / срезы	Мах. кол-во баллов	Методика проведения занятия и оценки
1.	Технология разработки алгоритмов. Решение задач на ЭВМ	Выполнение практических заданий	15	<p>5 баллов – практическое задание самостоятельно выполнено в полном объеме, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, уверенно, логично, последовательно и аргументировано отвечает на вопросы, используя профессиональную терминологию;</p> <p>4 балла – практическое задание самостоятельно и в основном правильно выполнено, студент владеет представленным материалом, логично, последовательно и аргументировано отвечает на заданные вопросы, используя профессиональную терминологию;</p> <p>3 балла - практическое задание в целом выполнено, но в процессе выполнения задания допущены существенные ошибки, студент не в полной мере владеет информацией по теме;</p> <p>2 балла - практическое задание выполнено не в полной мере, студент слабо владеет информацией по теме, затрудняется с ответами на задаваемые вопросы.</p>
		Опрос	5	<p>5 баллов – студент умеет сопоставить полученную при подготовке к практическому занятию информацию, сравнивать разные точки зрения на анализируемую проблему, умеет четко формулировать свои вопросы и отвечать на задаваемые ему вопросы;</p> <p>4 балла - студент умеет применять полученную при подготовке к практическому занятию информацию, отвечать на большинство вопросов, вести дискуссию;</p> <p>3 балла – студент владеет теоретическим материалом по теме практического занятия, иногда затрудняется при ответе на вопросы;</p> <p>2 балла - студент в неполной мере владеет теоретическим материалом по теме практического занятия, затрудняется сформулировать свою точку зрения на обсуждаемую проблему.</p> <p>Если студент не владеет проблематикой практического занятия, не может отвечать на вопросы, зачитывает ответ по напечатанному тексту – ответ баллами не оценивается.</p>

2.	Основы программирования на языке Pascal	Выполнение практических заданий	20	<p>5 баллов – практическое задание самостоятельно выполнено в полном объеме, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, уверенно, логично, последовательно и аргументировано отвечает на вопросы, используя профессиональную терминологию;</p> <p>4 балла – практическое задание самостоятельно и в основном правильно выполнено, студент владеет представленным материалом, логично, последовательно и аргументировано отвечает на заданные вопросы, используя профессиональную терминологию;</p> <p>3 балла - практическое задание в целом выполнено, но в процессе выполнения задания допущены существенные ошибки, студент не в полной мере владеет информацией по теме;</p> <p>2 балла - практическое задание выполнено не в полной мере, студент слабо владеет информацией по теме, затрудняется с ответами на задаваемые вопросы.</p>
		Выполнение практических заданий - контрольный срез (контрольный срез)	10	<p>Контрольный срез выполняются по тематике практических занятий. Максимальное количество 10 баллов ставится за абсолютно правильно выполненное практическое задание. Студент показал полный объем знаний, умений в освоении пройденных тем и применение их на практике. Нет ошибок в логических рассуждениях, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, свободно отвечает на вопросы, используя профессиональную терминологию;</p> <p>9 баллов – задание выполнено полностью, но допущены 1-3 легко устранимых недочета. Студент владеет представленным материалом, отвечает на заданные вопросы;</p> <p>8 баллов – за полностью выполненное задание с 1-2 незначительными ошибками;</p> <p>7 баллов - за полностью выполненное задание с 1-2 ошибками;</p> <p>6 баллов – за 70% выполнения задания без ошибок;</p> <p>5 баллов – за 70% выполнения задания с небольшими недочетами;</p> <p>4 балла – за 60% выполнения задания с небольшими недочетами.</p> <p>1-3 балла - задание выполнено не полностью, допущены грубые ошибки.</p> <p>0 баллов – работа не сдана.</p>
3.	Структуры данных в языке Pascal	Выполнение практических заданий	20	<p>5 баллов – практическое задание самостоятельно выполнено в полном объеме, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, уверенно, логично, последовательно и аргументировано отвечает на вопросы, используя профессиональную терминологию;</p> <p>4 балла – практическое задание самостоятельно и в основном правильно выполнено, студент владеет представленным материалом, логично, последовательно и аргументировано отвечает на заданные вопросы, используя профессиональную терминологию;</p> <p>3 балла - практическое задание в целом выполнено, но в процессе выполнения задания допущены существенные ошибки, студент не в полной мере владеет информацией по теме;</p> <p>2 балла - практическое задание выполнено не в полной мере, студент слабо владеет информацией по теме, затрудняется с ответами на задаваемые вопросы.</p>
		Тестирование	5	<p>За тестирование максимальное количество баллов 5 ставится за правильное выполнение 90-100% тестовых заданий; 4 балла – за 75-89% выполнения тестовых заданий; 3 балла – за 60-74% выполнения тестовых заданий; 2 балла – за 50-59% выполнения тестовых заданий.</p> <p>Менее 50% правильных ответов баллов не дает.</p>

		Выполнение практических заданий - контрольный срез(контрольный срез)	10	<p>Контрольный срез выполняются по тематике практических занятий. Максимальное количество 10 баллов ставится за абсолютно правильно выполненное практическое задание. Студент показал полный объем знаний, умений в освоении пройденных тем и применение их на практике. Нет ошибок в логических рассуждениях, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, свободно отвечает на вопросы, используя профессиональную терминологию;</p> <p>9 баллов – задание выполнено полностью, но допущены 1-3 легко устранимых недочета. Студент владеет представленным материалом, отвечает на заданные вопросы;</p> <p>8 баллов – за полностью выполненное задание с 1-2 незначительными ошибками;</p> <p>7 баллов - за полностью выполненное задание с 1-2 ошибками;</p> <p>6 баллов – за 70% выполнения задания без ошибок;</p> <p>5 баллов – за 70% выполнения задания с небольшими недочетами;</p> <p>4 балла – за 60% выполнения задания с небольшими недочетами.</p> <p>1-3 балла - задание выполнено не полностью, допущены грубые ошибки.</p> <p>0 баллов – работа не сдана.</p>
4.	Алгоритмы поиска и простой сортировки	Выполнение практических заданий	10	<p>5 баллов – практическое задание самостоятельно выполнено в полном объеме, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, уверенно, логично, последовательно и аргументировано отвечает на вопросы, используя профессиональную терминологию;</p> <p>4 балла – практическое задание самостоятельно и в основном правильно выполнено, студент владеет представленным материалом, логично, последовательно и аргументировано отвечает на заданные вопросы, используя профессиональную терминологию;</p> <p>3 балла - практическое задание в целом выполнено, но в процессе выполнения задания допущены существенные ошибки, студент не в полной мере владеет информацией по теме;</p> <p>2 балла - практическое задание выполнено не в полной мере, студент слабо владеет информацией по теме, затрудняется с ответами на задаваемые вопросы.</p>
		Тестирование	5	<p>За тестирование максимальное количество баллов 5 ставится за правильное выполнение 90-100% тестовых заданий; 4 балла – за 75-89% выполнения тестовых заданий; 3 балла – за 60-74% выполнения тестовых заданий; 2 балла – за 50-59% выполнения тестовых заданий.</p> <p>Менее 50% правильных ответов баллов не дает.</p>
5.	Премияльные баллы		10	Дополнительные премияльные баллы могут быть начислены за постоянную активность во время практических занятий – 10 баллов.
6.	Итого за семестр		100	

2 семестр

- текущий контроль – 50 баллов
- контрольные срезы – 2 среза по 10 баллов каждый
- премияльные баллы – 20 баллов
- ответ на экзамене: не более 30 баллов

Распределение баллов по заданиям:

№ те мы	Название темы / вид учебной работы	Формы текущего контроля / срезы	Мах. кол-во баллов	Методика проведения занятия и оценки
1.	Рекурсивные вычисления	Опрос	5	<p>5 баллов – студент умеет сопоставить полученную при подготовке к практическому занятию информацию, сравнивать разные точки зрения на анализируемую проблему, умеет четко формулировать свои вопросы и отвечать на задаваемые ему вопросы;</p> <p>4 балла - студент умеет применять полученную при подготовке к практическому занятию информацию, отвечать на большинство вопросов, вести дискуссию;</p> <p>3 балла – студент владеет теоретическим материалом по теме практического занятия, иногда затрудняется при ответе на вопросы;</p> <p>2 балла - студент в неполной мере владеет теоретическим материалом по теме практического занятия, затрудняется сформулировать свою точку зрения на обсуждаемую проблему. Если студент не владеет проблематикой практического занятия, не может отвечать на вопросы, зачитывает ответ по напечатанному тексту – ответ баллами не оценивается.</p>
		Выполнение практических заданий	5	<p>5 баллов – практическое задание самостоятельно выполнено в полном объеме, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, уверенно, логично, последовательно и аргументировано отвечает на вопросы, используя профессиональную терминологию;</p> <p>4 балла – практическое задание самостоятельно и в основном правильно выполнено, студент владеет представленным материалом, логично, последовательно и аргументировано отвечает на заданные вопросы, используя профессиональную терминологию;</p> <p>3 балла - практическое задание в целом выполнено, но в процессе выполнения задания допущены существенные ошибки, студент не в полной мере владеет информацией по теме;</p> <p>2 балла - практическое задание выполнено не в полной мере, студент слабо владеет информацией по теме, затрудняется с ответами на задаваемые вопросы.</p>
2.	Основы визуального программирования в среде Delphi	Опрос	5	<p>5 баллов – студент умеет сопоставить полученную при подготовке к практическому занятию информацию, сравнивать разные точки зрения на анализируемую проблему, умеет четко формулировать свои вопросы и отвечать на задаваемые ему вопросы;</p> <p>4 балла - студент умеет применять полученную при подготовке к практическому занятию информацию, отвечать на большинство вопросов, вести дискуссию;</p> <p>3 балла – студент владеет теоретическим материалом по теме практического занятия, иногда затрудняется при ответе на вопросы;</p> <p>2 балла - студент в неполной мере владеет теоретическим материалом по теме практического занятия, затрудняется сформулировать свою точку зрения на обсуждаемую проблему. Если студент не владеет проблематикой практического занятия, не может отвечать на вопросы, зачитывает ответ по напечатанному тексту – ответ баллами не оценивается.</p>

		Выполнение практических заданий	5	<p>5 баллов – практическое задание самостоятельно выполнено в полном объеме, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, уверенно, логично, последовательно и аргументировано отвечает на вопросы, используя профессиональную терминологию;</p> <p>4 балла – практическое задание самостоятельно и в основном правильно выполнено, студент владеет представленным материалом, логично, последовательно и аргументировано отвечает на заданные вопросы, используя профессиональную терминологию;</p> <p>3 балла - практическое задание в целом выполнено, но в процессе выполнения задания допущены существенные ошибки, студент не в полной мере владеет информацией по теме;</p> <p>2 балла - практическое задание выполнено не в полной мере, студент слабо владеет информацией по теме, затрудняется с ответами на задаваемые вопросы.</p>
3.	Работа с графикой в среде Delphi	Выполнение практических заданий	5	<p>5 баллов – практическое задание самостоятельно выполнено в полном объеме, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, уверенно, логично, последовательно и аргументировано отвечает на вопросы, используя профессиональную терминологию;</p> <p>4 балла – практическое задание самостоятельно и в основном правильно выполнено, студент владеет представленным материалом, логично, последовательно и аргументировано отвечает на заданные вопросы, используя профессиональную терминологию;</p> <p>3 балла - практическое задание в целом выполнено, но в процессе выполнения задания допущены существенные ошибки, студент не в полной мере владеет информацией по теме;</p> <p>2 балла - практическое задание выполнено не в полной мере, студент слабо владеет информацией по теме, затрудняется с ответами на задаваемые вопросы.</p>
		Выполнение практических заданий - контрольный срез (контрольный срез)	10	<p>Контрольный срез выполняются по тематике практических занятий. Максимальное количество 10 баллов ставится за абсолютно правильно выполненное практическое задание. Студент показал полный объем знаний, умений в освоении пройденных тем и применение их на практике. Нет ошибок в логических рассуждениях, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, свободно отвечает на вопросы, используя профессиональную терминологию;</p> <p>9 баллов – задание выполнено полностью, но допущены 1-3 легко устранимых недочета. Студент владеет представленным материалом, отвечает на заданные вопросы;</p> <p>8 баллов – за полностью выполненное задание с 1-2 незначительными ошибками;</p> <p>7 баллов - за полностью выполненное задание с 1-2 ошибками;</p> <p>6 баллов – за 70% выполнения задания без ошибок;</p> <p>5 баллов – за 70% выполнения задания с небольшими недочетами;</p> <p>4 балла – за 60% выполнения задания с небольшими недочетами.</p> <p>1-3 балла - задание выполнено не полностью, допущены грубые ошибки.</p> <p>0 баллов – работа не сдана.</p>

4.	Указатели. Динамическая память	Опрос	5	<p>5 баллов – студент умеет сопоставить полученную при подготовке к практическому занятию информацию, сравнивать разные точки зрения на анализируемую проблему, умеет четко формулировать свои вопросы и отвечать на задаваемые ему вопросы;</p> <p>4 балла - студент умеет применять полученную при подготовке к практическому занятию информацию, отвечать на большинство вопросов, вести дискуссию;</p> <p>3 балла – студент владеет теоретическим материалом по теме практического занятия, иногда затрудняется при ответе на вопросы;</p> <p>2 балла - студент в неполной мере владеет теоретическим материалом по теме практического занятия, затрудняется сформулировать свою точку зрения на обсуждаемую проблему.</p> <p>Если студент не владеет проблематикой практического занятия, не может отвечать на вопросы, зачитывает ответ по напечатанному тексту – ответ баллами не оценивается.</p>
		Выполнение практических заданий	5	<p>5 баллов – практическое задание самостоятельно выполнено в полном объеме, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, уверенно, логично, последовательно и аргументировано отвечает на вопросы, используя профессиональную терминологию;</p> <p>4 балла – практическое задание самостоятельно и в основном правильно выполнено, студент владеет представленным материалом, логично, последовательно и аргументировано отвечает на заданные вопросы, используя профессиональную терминологию;</p> <p>3 балла - практическое задание в целом выполнено, но в процессе выполнения задания допущены существенные ошибки, студент не в полной мере владеет информацией по теме;</p> <p>2 балла - практическое задание выполнено не в полной мере, студент слабо владеет информацией по теме, затрудняется с ответами на задаваемые вопросы.</p>
5.	Динамические структуры данных	Выполнение практических заданий	5	<p>5 баллов – практическое задание самостоятельно выполнено в полном объеме, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, уверенно, логично, последовательно и аргументировано отвечает на вопросы, используя профессиональную терминологию;</p> <p>4 балла – практическое задание самостоятельно и в основном правильно выполнено, студент владеет представленным материалом, логично, последовательно и аргументировано отвечает на заданные вопросы, используя профессиональную терминологию;</p> <p>3 балла - практическое задание в целом выполнено, но в процессе выполнения задания допущены существенные ошибки, студент не в полной мере владеет информацией по теме;</p> <p>2 балла - практическое задание выполнено не в полной мере, студент слабо владеет информацией по теме, затрудняется с ответами на задаваемые вопросы.</p>
		Тестирование	5	<p>За тестирование максимальное количество баллов 5 ставится за правильное выполнение 90-100% тестовых заданий; 4 балла – за 75-89% выполнения тестовых заданий; 3 балла – за 60-74% выполнения тестовых заданий; 2 балла – за 50-59% выполнения тестовых заданий.</p> <p>Менее 50% правильных ответов баллов не дает.</p>

		Выполнение практических заданий - контрольный срез (контрольный срез)	10	<p>Контрольный срез выполняются по тематике практических занятий. Максимальное количество 10 баллов ставится за абсолютно правильно выполненное практическое задание. Студент показал полный объем знаний, умений в освоении пройденных тем и применение их на практике. Нет ошибок в логических рассуждениях, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, свободно отвечает на вопросы, используя профессиональную терминологию;</p> <p>9 баллов – задание выполнено полностью, но допущены 1-3 легко устранимых недочета. Студент владеет представленным материалом, отвечает на заданные вопросы;</p> <p>8 баллов – за полностью выполненное задание с 1-2 незначительными ошибками;</p> <p>7 баллов - за полностью выполненное задание с 1-2 ошибками;</p> <p>6 баллов – за 70% выполнения задания без ошибок;</p> <p>5 баллов – за 70% выполнения задания с небольшими недочетами;</p> <p>4 балла – за 60% выполнения задания с небольшими недочетами.</p> <p>1-3 балла - задание выполнено не полностью, допущены грубые ошибки.</p> <p>0 баллов – работа не сдана.</p>
6.	Модули	Выполнение практических заданий	5	<p>5 баллов – практическое задание самостоятельно выполнено в полном объеме, студент свободно владеет материалом, демонстрирует глубокие, систематизированные знания, уверенно, логично, последовательно и аргументировано отвечает на вопросы, используя профессиональную терминологию;</p> <p>4 балла – практическое задание самостоятельно и в основном правильно выполнено, студент владеет представленным материалом, логично, последовательно и аргументировано отвечает на заданные вопросы, используя профессиональную терминологию;</p> <p>3 балла - практическое задание в целом выполнено, но в процессе выполнения задания допущены существенные ошибки, студент не в полной мере владеет информацией по теме;</p> <p>2 балла - практическое задание выполнено не в полной мере, студент слабо владеет информацией по теме, затрудняется с ответами на задаваемые вопросы.</p>
7.	Премияльные баллы		20	<p>Дополнительные премияльные баллы могут быть начислены:</p> <ul style="list-style-type: none"> - за проект, выполненный по заказу работодателя и реализованный на практике – 20 баллов; - постоянная активность во время практических занятий – 10 баллов; - полностью подготовленная к публикации статья по тематике в рамках дисциплины – 10 баллов; - участие с докладом во всероссийской олимпиаде по тематике изучаемой дисциплины – 20 баллов; - участие в выставке по тематике изучаемой дисциплины – 20 баллов; - публикация статьи по тематике изучаемой дисциплины в сборнике студенческих работ / материалах всероссийской конференции / журнале из перечня ВАК – 10 / 15 / 20
8.	Ответ на экзамене		30	<p>10-17 баллов – студент раскрыл основные вопросы и задания билета на оценку «удовлетворительно»</p> <p>18-24 баллов – студент раскрыл основные вопросы и задания билета на оценку «хорошо»,</p> <p>25-30 баллов – студент раскрыл основные вопросы и задания билета на оценку «отлично».</p>

9.	Индивидуальные задания, с помощью которых можно набрать дополнительные баллы	20	Решение кейса (10 баллов) Прохождение тестирования (30 вопросов) по всему курсу дисциплины (10 баллов)
10.	Итого за семестр	100	

Итоговая оценка по экзамену выставляется в 100-балльной шкале и в традиционной четырехбалльной шкале. Перевод 100-балльной рейтинговой оценки по дисциплине в традиционную четырехбалльную осуществляется следующим образом:

100-балльная система	Традиционная система
85 - 100 баллов	Отлично
70 - 84 баллов	Хорошо
50 - 69 баллов	Удовлетворительно
Менее 50	Неудовлетворительно

4.2 Типовые оценочные средства текущего контроля

Выполнение практических заданий

Тема 1. Технология разработки алгоритмов. Решение задач на ЭВМ

Практическое занятие 1. Технология разработки алгоритмов. Моделирование задач и разработка линейных и разветвляющихся алгоритмов. Решение задач.

Практические задания.

1. Составить и выполнить алгоритм, которая находит среди данных трех чисел среднее (под средним числом будем понимать число, которое больше наименьшего, но меньше наибольшего).
2. Составить и выполнить алгоритм вывода на экран номера четверти, которой принадлежит точка с координатами (X,Y) при условии, что X и Y отличны от 0.
3. Составить и выполнить алгоритм, который располагает три числа в порядке возрастания.
4. Составить и выполнить алгоритм, который осуществляет перевод арабской цифры в римскую.
5. Составить и выполнить алгоритм, который подсчитывает, сколько вводится 1, 2 и 3 среди последовательности 5 чисел.
6. Составить и выполнить алгоритм, который по возрасту человека в годах ($B < 100$) приписывает к этому числу текст ГОД, ГОДА или ЛЕТ. Например, 1 ГОД; 23 ГОДА, 45 ЛЕТ.
7. Составить и выполнить алгоритм, который по номеру месяца определяет количество дней в месяце (год - не високосный).

Практическое занятие 2. Технология разработки циклических алгоритмов. Решение задач.

Практические задания.

1. В трехзначном числе зачеркнули первую цифру слева, когда полученное двузначное число умножили на 7, то получили данное число. Найти это число. Ответ: 350.
2. Даны натуральные числа n, k ($n, k \leq 9999$). Из чисел от n до k выберете те, запись которых содержит три одинаковых цифры. Например, числа 6766, 5444, 0006, 0060 содержат ровно три одинаковых цифры.
3. Сто шаров, массы которых равны соответственно 1, 2, 3, 4, ..., 100 г, укреплены на стержне так, что их центры находятся на расстоянии 10 см друг от друга. Составить и выполнить программу, которая находит центр тяжести системы (массой стержня пренебречь).
4. Составить и выполнить программу, которая приписывает по 1 в начало и в конец записи числа X. Например, было $X=3456$, стало $X=134561$.
5. Составить и выполнить программу, которая находит самую большую цифру целого числа.

Практическое занятие 3. Использование вспомогательных алгоритмов при решении задач.

Практические задания.

- 1 6. Составить программу нахождения и вывода на экран всех простых чисел из заданного промежутка $[n; m]$. (Массивы не использовать.)
- 2 7. Найти площадь фигуры, которая получена вырезанием из квадрата со стороной A квадрата со стороной B ($A > B$).
- 3 8. Найти сумму цифр всех целых чисел от n до m (n, m – целые числа)
- 4 9. Найти сумму цифр всех целых чисел от n до m
- 5 10. Найти сумму всех делителей чисел от n до m (n, m – натуральные числа).
- 6 11. Найти все совершенные числа на промежутке от n до m (совершенное число равно сумме всех своих делителей, кроме него самого).
- 7 12. Определить НОД четырех натуральных чисел

Тема 2. Основы программирования на языке Pascal

Практические занятия.

Практическое занятие 1. Работа в консольном приложении Turbo Delphi. Набор и редактирование программ. Разработка и реализация простейших программ использованием стандартных функций и операции над числовыми данными при разработке линейных программ.

Практические задания.

Составить и выполнить программу, меняющей местами значения переменных X и Y без использования промежуточной переменной.

Составить и выполнить программу нахождения суммы цифр заданного трехзначного числа.

Составить и выполнить программу определения времени приземления самолета в часах, минутах и секундах, если известно время взлета (часы, минуты, секунды) и продолжительность полета в секундах. Учесть возможный переход на другие сутки.

Составить и выполнить программу, которая определяет произведение цифр заданного четырехзначного числа.

Заданы три корня кубического уравнения: x_1, x_2, x_3 . Найти коэффициенты этого уравнения.

Заданы координаты точки подвески математического маятника $A(x_0, y_0, z_0)$ и координаты одной из точек его наивысшего подъема $B(x_1, y_1, z_1)$. Найти координаты самой низкой точки траектории и другой наивысшей точки подъема.

У квадрата ABCD на плоскости известны координаты двух противоположных вершин – точек A и C. Найти координаты точек B и D.

Русские неметрические единицы длины: 1 верста = 500 сажень; 1 сажень = 3 аршина; 1 аршин = 16 вершков; 1 вершок = 44,45 мм. Длина некоторого отрезка составляет p метров. Перевести ее в русскую неметрическую систему.

Найти координаты вершины параболы

Треугольник ABC задан длинами своих сторон. Найти длину высоты, опущенной из вершины A.

Практическое занятие 2. Работа в консольном приложении Turbo Delphi. Операторы IF - THEN – ELSE; CASE. Решение задач на использование команд ветвления.

Практические задания.

1. Составить и выполнить программу, которая находит среди данных трех чисел среднее (под средним числом будем понимать число, которое больше наименьшего, но меньше наибольшего).
2. Составить и выполнить программу, которая определяла, принадлежит ли точка с координатами (X,Y) прямоугольнику с координатами (X1,Y1) (X2,Y2).
3. Составить и выполнить программу, которая находит наибольшее значение трех чисел без дополнительной четвертой переменной.
4. Составить и выполнить программу, которая располагает три числа в порядке возрастания.

5. Даны размеры кирпича $A \times B \times C$ и размер квадратного отверстия в стене X . Составить и выполнить программу, которая определяет, можно ли просунуть этот кирпич через это отверстие (просовывать только так, чтобы ребра были параллельны сторонам).
 6. Составить и выполнить программу, которая осуществляет полное исследование совокупности корней биквадратного уравнения $A \cdot X^4 + B \cdot X^2 + C = 0$. Если корней нет, то выводится сообщение об этом, иначе – два или четыре корня.
 7. Составить и выполнить программу, которая определяет, является ли год високосным.
Примечание. По Григорианскому календарю, введенному в 1582 году, год G считается високосным, если G кратно четырем, не кратно ста, но кратно четыремстам.
 8. Составить и выполнить программу, которая определяет число дней в месяце.
 9. Составить и выполнить программу, которая определяет дату завтрашнего дня.
 10. Составить и выполнить программу, которая определяет дату вчерашнего дня.
 11. Составить и выполнить программу, которая определяет дату, наступающую через N дней после указанной даты.
 12. Составить и выполнить программу, которая определяет число дней между двумя датами.
 13. Напишите программу, которая проверяет, кратно ли введенное пользователем с клавиатуры значение числам 3 и 5.
 14. Напишите программу начисления торговой скидки. Вводится сумма покупок, программа рассчитывает сумму к оплате с учетом того, что при покупке товара на сумму от 500 до 1000 руб. предоставляется скидка 5 %, на сумму от 1001 до 10000 руб. – 7 %, на сумму свыше 10000 руб. – 9 %.
 15. Напишите программу элементарного калькулятора для выполнения четырех арифметических действий. Программа должна проверять корректность вводимых данных, например, исключать деление на ноль.
 16. Напишите программу, которая считывает три вещественных числа и заменяет каждое отрицательное нулем, а нулевое – числом.
 17. Напишите программу, которая по введенному возрасту определяет, к какой возрастной группе относится человек: дошкольник, учащийся, работающий или пенсионер.
 18. Составить программу перевода чисел из римской формы записи в десятичную арабскую.
 19. Составить и выполнить программу, которая подсчитывает, сколько вводится 1, 2 и 3 среди введенных 5 чисел.
 20. Составить и выполнить программу, которая по номеру дня недели выводит соответствующее название на русском и английском языках.
 21. Составить и выполнить программу, которая вычисляет площади прямоугольника, прямоугольного треугольника, трапеции, круга.
 22. Составить и выполнить программу, которая определяет, введено ли четное или нечетное число (меньшее 10), большее 10, но меньшее 100, или какое-то другое число.
 23. Составить и выполнить программу, которая определяет, введена ли цифра, строчная латинская буква, прописная латинская буква, нажата ли клавиша-стрелка или введен другой символ.
- Практическое занятие 3. Работа в консольном приложении Turbo Delphi. Циклы FOR, WHILE, REPEAT. Решение задач с использованием команд ветвления и циклов.
- Практические задания.
1. Даны натуральные числа A и B . Найти все пары дружественных чисел, лежащих в диапазоне от A до B . Два числа называются дружественными, если каждое из них равно сумме всех делителей другого (само число в качестве делителя не рассматривается).
 2. Составить и выполнить программу, которая находит все двузначные числа, которые в сумме с числом, записанным теми же цифрами, но в обратном порядке, дают квадрат натурального числа.
 3. Составить и выполнить программу, которая подсчитывает количество счастливых билетов.
 4. Составить и выполнить программу, которая выводит на экран K -ую строку треугольника Паскаля.
 5. Кирпичи укладываются так, что часть верхнего кирпича выступает над нижним, образуя карниз. Составить и выполнить программу, которая определяет максимальную длину карниза, которую можно получить из 20 кирпичей (длина кирпича 24 см, кладка сухая, без раствора).

6. Сто шаров, массы которых равны соответственно 1, 2, 3, 4, ..., 100 г, укреплены на стержне так, что их центры находятся на расстоянии 10 см друг от друга. Составить и выполнить программу, которая находит центр тяжести системы (массой стержня пренебречь).
7. Составить и выполнить программу, которая приписывает по 1 в начало и в конец записи числа X. Например, было $X=3456$, стало $X=134561$.
8. Составить и выполнить программу, которая находит самую большую цифру целого числа.
9. Составить и выполнить программу, которая вычисляет НОД двух чисел с помощью алгоритма Евклида.
10. Составить и выполнить программу, которая угадывает «задуманное» компьютером случайное натуральное число из диапазона (0,1000).
11. Составить и выполнить программу, которая подсчитывает, сколько чисел Фибоначчи находится в заданном диапазоне.
12. Составить и выполнить программу, которая выводит на экран пары взаимно-простых чисел в интервале [1, 1000].
13. Составить и выполнить программу, которая определяет, является ли заданное натуральное число палиндромом.
14. Камень свободно падает и в последнюю секунду падения проходит половину всего пути. Составить и выполнить программу, которая определяет, с какой высоты он падает.
15. Кошка догоняет мышь, бегущую по прямой со скоростью U. Скорость кошки $V>U$ и всегда направлена на мышь. В начальный момент скорости U и V перпендикулярны. Составить и выполнить программу, которая определяет, через сколько прыжков кошка настигнет мышь, если первоначальное расстояние между ними L, а длина прыжка кошки L1.
16. Составить и выполнить программу, которая находит наименьший, отличный от 1, делитель натурального числа.
17. Составить и выполнить программу, которая планирует закупку товара на сумму, не превышающую заданную величину.

Практическое занятие 4. Работа в консольном приложении Turbo Delphi. Процедуры и функции. Разработка программ с использованием процедур и функций. Решение задач.

Практические задания.

1. Составить и выполнить программу, которая содержит подпрограммы для вычисления периметра и площади треугольника.
2. Составить и выполнить программу, которая с помощью подпрограммы меняет местами значения двух переменных.
3. Составить и выполнить программу, которая находит все числа из промежутка от A до B, у которых больше всего делителей.
4. Составить и выполнить программу, которая содержит подпрограммы для действий над комплексными числами: сложение, умножение, деление, умножение на комплексное число.
5. Разработать функцию типа Boolean, определяющую, является ли год високосным. (Примечание. По Григорианскому календарю, введенному в 1582 году, год G считается високосным, если G кратно четырем, не кратно стам, но кратно четырестам).
6. Разработать функцию типа Integer, определяющую число дней в месяце.
7. Разработать процедуру, определяющую дату завтрашнего дня.
8. Разработать процедуру, определяющую дату вчерашнего дня.
9. Разработать процедуру, определяющую дату, наступающую через N дней после указанной даты.
10. Разработать процедуру, определяющую число дней между двумя датами.
11. Даны три натуральных числа. Найти их наибольший общий делитель.
12. Два натуральных числа называются дружественными, если каждое из них равно сумме всех делителей другого, за исключением его самого. Напечатать все пары дружественных чисел, не превосходящих заданного натурального числа.
13. Определить, являются ли перевертышами числа a и b. Указание: два числа называются перевертышами, если они записаны одними и теми же числами, но в обратном порядке.
14. Составьте программу для определения большего из четырех чисел.

15. Определить, сократима ли дробь m/n .

16. Даны координаты вершин треугольника $A(x_1; y_1)$, $B(x_2; y_2)$, $C(x_3, y_3)$. Найти его периметр.
Указание: воспользоваться формулой вычисления расстояния между двумя точками A и B .

17. Описать процедуру $PS(a, P, S)$, вычисляющую по стороне a равностороннего треугольника его периметр $P = 3 \cdot a$ и площадь $S = a^2 \cdot \sqrt{3}/4$ (a — входной, P и S — выходные параметры; все параметры являются вещественными). С помощью этой процедуры найти периметры и площади трех равносторонних треугольников с данными сторонами.

18. Описать процедуру $Por(K)$, меняющую порядок следования цифр целого положительного числа K на обратный (K — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой процедуры поменять порядок следования цифр на обратный для каждого из пяти данных целых чисел.

Тема 3. Структуры данных в языке Pascal

Практические занятия.

Практическое занятие 1. Работа в консольном приложении TurboDelphi. Ограниченный и перечислимый типы данных.

Практические задания.

Практическое занятие 2. Работа в консольном приложении TurboDelphi. Массивы в языке Pascal. Решение задач с использованием массивов.

Практические задания.

1. Дан массив целых чисел. Найти сумму всех простых чисел.
2. Даны три массива A , B , C . Найдите значение наибольшего элемента среди элементов трех массивов, определите, который из массивов его содержит.
3. Дан массив целых чисел. Вывести на экран все простые числа. Указание: использовать функцию, которая проверяет, является ли число простым.
4. Заданы два массива A и B . Создайте массив C , в котором каждый элемент равен сумме соответствующих элементов массивов A и B .
5. Заданы два массива, определите, равны ли их максимальные элементы.
6. Заданы два массива, содержится ли минимальный элемент первого массива среди элементов второго массива?
7. Даны два одномерных массива из целых чисел. Вывести элементы, которые есть в одном массиве, но нет в другом.
8. Составить и выполнить программу, которая определяет, есть ли отрицательный элемент в массиве.
9. Составить и выполнить программу, которая находит номер последнего отрицательного элемента массива.
10. Составить и выполнить программу, которая находит максимальный элемент массива и его номер, при условии, что все элементы различные.
11. Составить и выполнить программу, которая находит номера всех элементов с минимальным значением.
12. Составить и выполнить программу, которая определяет, есть ли в данном массиве два соседних положительных элемента. Найти номера первой пары.
13. Имеется перечень зверей. Определить в этом списке порядковый номер тигра, а также название и порядковый номер зверя после косули.
14. Составить и выполнить программу, которая находит скалярное произведение двух одномерных массивов.
15. Составить и выполнить программу, которая удаляет из массива максимальные элементы.
16. Составить и выполнить программу, которая вставляет число A после каждого элемента массива, кратного трем.
17. Составить и выполнить программу, которая в одномерном массиве номер и значение последнего отрицательного элемента или сообщает о его отсутствии.

18. Составить и выполнить программу, которая из двух заданных упорядоченных по возрастанию массивов $X[1..M]$ и $Y[1..T]$ образует упорядоченный по возрастанию третий массив $C[1..M+T]$.
19. Составить и выполнить программу, которая осуществляет сдвиг на одну позицию всех элементов вправо, а последний элемент ставит первым.
20. Составить и выполнить программу, которая для заданного одномерного массива $X[1..N]$ находит номера элементов k, m , такие, что $X[k] + X[k+1] + \dots + X[m]$ будет максимальной из всех сумм подряд идущих элементов.
21. Составить и выполнить программу, которая проверяет, является ли одномерный массив упорядоченным по возрастанию.
22. Составить и выполнить программу, которая осуществляет циклический сдвиг на K позиций всех элементов вправо, последние K элементов располагаются в начале массива.
23. Составить и выполнить программу, которая проверяет, имеется в массиве хотя бы одна пара чисел, совпадающих по величине.
24. Составить и выполнить программу, которая определяет в массиве число соседств из двух положительных чисел, число соседств из двух чисел одного знака.
25. Составить и выполнить программу, которая в одномерном массиве переставляет в обратном порядке элементы массива, расположенные между минимальным и максимальным элементами.
26. Составить и выполнить программу, которая по заданному одномерному массиву A строит новый массив B , в котором сначала идут все положительные, затем все нулевые, а затем все отрицательные элементы массива A , причем в каждой из этих трех групп порядок следования элементов должен быть сохранен (сортировку исходного массива не производить).
27. Составить и выполнить программу, которая удаляет все повторяющиеся элементы, оставляя только их первые вхождения (получить массив различных элементов).
28. Составить и выполнить программу, которая упорядочивает элементы массива по возрастанию.
29. Составить и выполнить программу, которая с помощью подпрограммы-функции, осуществляющей циклический сдвиг массива, сделает это несколько раз. Обобщить подпрограмму: циклический сдвиг должен производиться с частью массива.
30. Дано 200 вещественных чисел. Определить, сколько из них больше своих «соседей», то есть предыдущего и последующего чисел.
31. Составить и выполнить программу, которая находит наибольший и наименьший элементы массива и их номера.
32. Составить и выполнить программу, которая переставляет элементы, расположенные между максимальным и минимальным элементами массива, в обратном порядке.

Практическое занятие 3. Работа в консольном приложении TurboDelphi. Работа со строковыми величинами.

Практические задания.

1. Составить и выполнить программу, которая определяет количество вхождений данного символа в строку.
2. Строка состоит из нескольких предложений, каждое из которых кончается точкой, восклицательным или вопросительным знаком. Составить и выполнить программу, которая определяет количество предложений в строке.
3. Составить и выполнить программу, которая в данной строке S заменяет подстроку 'ый' (если она там есть) на подстроку 'ая' (например, слово 'красный' превращается в 'красная').
4. Составить и выполнить программу, которая выясняет, встречается ли данный символ в строке.
5. Составить и выполнить программу, которая выясняет, является ли слово S перевертышем.
6. Составить и выполнить программу, которая заменяет в строке один символ на другой, отметив, была ли проведена хотя бы одна замена (например, если символ '+' заменяется на '-', то строка 'a+b+c-d' преобразуется в строку 'a-b-c-d').
7. Составить и выполнить программу, которая по заданной фамилии, имени и отчеству печатает фамилию и инициалы.
8. Составить и выполнить программу, которая находит количество символов в строке, отличных от пробела («значащих символов»).

9. Составить и выполнить программу, которая по числу, не превышающему 999 и написанному арабскими цифрами, формирует его название.
10. Составить и выполнить программу, которая по названию числа, не превышающему 999 и написанному на русском (английском языке), формирует его цифровую запись.
11. Составить и выполнить программу, которая после ввода строки строчных латинских букв заменяет их на прописные.
12. Подсчитать сумму цифр, встречающихся в строке.
13. Определить, какое из двух слов длиннее и на сколько.
14. Определить, является ли какое-нибудь из двух слов частью другого.
15. Поменять в слове первую и последнюю буквы.
16. Если в слове нечетное число букв, то удвоить среднюю букву.
17. По последнему символу определить тип предложения (повествовательное, вопросительное, восклицательное).
18. Определить, является ли данный символ латинской буквой.
19. Удалить из слова среднюю букву (или две средних).
20. Составить и выполнить программу, которая дешифрует текстовое сообщение, зашифрованное в соответствии с условием предыдущей задачи.

Практическое занятие 4. Работа в консольном приложении TurboDelphi. Работа с множествами.

Практические задания.

1. Дана последовательность символов. Требуется построить и напечатать множество, элементами которого являются встречающиеся в последовательности:
 - а) цифры от '0' до '9';
 - б) буквы от 'A' до 'F' и от 'X' до 'Z';
 - в) буквы от 'G' до 'N' и цифры от '0' до '9';
 - г) знаки препинания;
 - д) буквы от 'A' до 'Z' и цифра от '0' до '5';
 - е) буквы от 'T' до 'X' и знаки препинания;
 - ж) цифры от '5' до '9' и знаки арифметических операций;
 - з) знаки арифметических операций и знаки препинания;
 - и) цифры и знаки арифметических операций;
 - к) знаки препинания и буквы от 'E' до 'N'.
2. Составить и выполнить программу, которая выделяет из множества целых чисел от 1 до 30 следующие множества:
 - а) множество чисел, кратных 2;
 - б) множество чисел, кратных 3;
 - в) множество чисел, кратных 6;
 - г) множество чисел, кратных 2 или 3;
 - д) множество чисел, кратных 3, но не кратных 2.
3. Составить и выполнить программу, которая по заданному тексту, состоящему из строчных латинских букв и заканчивающемся точкой, печатает:
 - а) все буквы, входящие в текст по одному разу;
 - б) все буквы, входящие в текст не менее двух раз.
4. Составить и выполнить программу, которая для заданного целого K (от 2 до 1000), используя метод «решета Эратосфена», печатает в убывающем порядке все простые числа из диапазона $K..2*K$.
5. Даны по 5-балльной шкале оценки по информатике трех студентов. Сформировать множество оценок, которые есть и у первого и второго, но нет у третьего студента. Вывести элементы этого множества в порядке убывания.
6. Даны два конечных множества A и B , элементами которых могут быть любые целые числа в диапазоне от 1 до 30. Найти прямое произведение этих множеств и вывести его на экран.

7. Даны два прямоугольника. Множества А и В – это множества точек, принадлежащих соответствующим прямоугольникам. Координаты точек – это натуральные числа от 1 до 10. Определить пересекаются ли данные прямоугольники, если пересекаются, то вывести на экран их общие точки.

8. В кругу стоит n детей, а в считалке m слов. Составить программу, которая напечатает номера детей в том порядке, в каком они выходят из круга.

9. Коту снится, что его окружили тринадцать мышей. Двенадцать из них серые, а одна белая. Слышит кот, что кто-то говорит ему: «Мурлыка, ты можешь съедать каждую тринадцатую мышку. Считай их по кругу в одном направлении. Белую мышку ты должен съесть последней». Задумался кот: с какой мышки начинать счет?

Практическое занятие 5. Работа в консольном приложении TurboDelphi. Работа с комбинированным типом (записи).

Практические задания.

1. Составить и выполнить программу, которая по заданному списку N студентов (фамилия и оценки, полученные по двум экзаменам) формирует новый список студентов, каждая строка которого содержит фамилию и средний балл каждого, а также средний балл группы.
2. Составить и выполнить программу, которая находит сумму и разность двух комплексных чисел.
3. Составить и выполнить программу, которая определяет площадь различных геометрических фигур: окружности (по ее радиусу), квадрата (по его стороне), прямоугольника (по двум сторонам), треугольника (по трем сторонам).
4. Создан массив автовладельцев. Для каждого автовладельца известны номер, марка автомобиля, фамилия и адрес. Написать программу, выясняющую, есть ли автовладельцы, имеющие более одного автомобиля.

5. Создан массив автовладельцев. Для каждого автовладельца известны номер, марка автомобиля, фамилия и адрес. Написать программу, выясняющую, сколько машин разных марок имеется.

6. Пусть точка описывается типом:

```
type point = record x,y : real end.
```

Выпуклый многоугольник задан массивом вершин, перечисленных в порядке обхода по часовой стрелке. Написать программу для определения периметра и площади многоугольника.

7. Пусть поле шахматной доски описывается типом

```
type doska = record vert: (a, b, c,d,e,f,g,h); horiz:1..8; end;
```

Описать логические функции, проверяющие, может ли любая из шахматных фигур за один ход перейти с поля $n1$ на поле $n2$.

8. Описать тип дата. Написать программу, которая:

- а) из двух дат $T1$ и $T2$ будет определять более раннюю;
- б) находить количество суток между заданными датами $T1$ и $T2$;
- в) количество лет, месяцев и дней между заданными датами $T1$ и $T2$;
- г) дату, которая наступит через M дней;
- д) дату, которая была за M дней до сегодняшнего дня.

Практическое занятие 6. Работа в консольном приложении TurboDelphi. Работа с файлами.

Практические задания.

1. Найти максимальный элемент файла и его порядковый номер.
2. В файле содержатся числа от 1 до 10. Сосчитать, сколько раз встречается каждое из них.
3. Определить, встречается ли число x в данном файле целых чисел.
4. Дан текстовый файл. Найти количество вхождений данного символа в текстовом файле.
5. Дан текстовый файл. Найти строку в текстовом файле (вывести на экран), в которой данный символ встречается максимальное количество раз.
6. Создать файл с типом записи, в котором содержатся сведения о сотрудниках учреждения. Записи имеют поля: Табельный №; Ф.И.О.; Должность; Оклад; Стаж работы. Упорядочить записи по стажу работы, вывести их. Вывести список фамилий сотрудников с указанием должности и оклада, месячная заработная плата которых превышает заданную величину.

7. Создать файл с типом записи, упорядоченный по № телефона. Поля записи: № телефона; Код города; Дата разговора; Количество минут; Сумма к оплате. Вывести записи файла в таблицу, предварительно упорядочив их по сумме. Вывести квитанцию по оплате телефонного разговора для заданного номера телефона.

8. Создать файл с типом записи. Запись имеет следующие поля: № счета; Ф.И.О.; Адрес (город, улица, дом); Сумма вклада; Срочность; Текущий год; Год вклада. Вывести на экран в виде таблицы сведения о вкладчиках, имеющих срочный вклад. Дополнить файл новыми записями при условии, что файл упорядочен по номеру счёта.

9. Создать файл с типом записи поля, в котором записи о товарах имеют поля: Номер покупателя; Номер торгового агента; Номер счета; Номер товара; Размер закупок; Цена; Описание. Отсортировать записи о товарах по номерам покупателей и вывести в виде таблицы. Вывести счета для покупателей, номера которых вводятся с экрана.

10. Создать файл с типом записи, содержащий сведения о рабочих предприятия. Каждая запись имеет поля: Наименование предприятия; Ф.И.О. рабочего; Должность; Зарплата, начисленная за месяц. Вывести в виде таблицы содержимое записей, в которых значение начисленной зарплаты ниже средней. Изменить содержание поля "зарплата" для заданных фамилий рабочих.

Тема 4. Алгоритмы поиска и простой сортировки

Практическое занятие 1. Работа в консольном приложении Turbo Delphi. Организация поиска данных.

Практическое задание.

Разработать программу по реализации методов поиска информации.

1. Исходный текст программы должен содержать:

- ☐ процедуру заполнения массива размерности N случайными элементами типа Element;
- ☐ процедуру распечатки на экране элементов массива;
- ☐ подпрограмму поиска минимального, максимального элемента массива;
- ☐ подпрограмму поиска номера минимального, максимального элемента массива;
- ☐ подпрограмму поиска минимального, максимального элемента массива и его номера;
- ☐ процедуру, реализующую метод перебора;
- ☐ процедуру, реализующую метод бинарного (дихотомического) поиска;
- ☐ процедуру, реализующую добавления элемента в массив;
- ☐ процедуру, реализующую удаление элемента из массива.

2. В основной части программы необходимо организовать:

- ☐ возможность выбора метода поиска в режиме диалога;
- ☐ возможность проведения, при необходимости, предварительной сортировки исходного массива;
- ☐ определение времени, затраченного на поиск;
- ☐ запись результатов в файл.

3. Проанализировать полученные результаты.

Практическое занятие 2. Работа в консольном приложении Turbo Delphi. Организация сортировки данных.

Практическое задание.

Разработать программу по реализации методов поиска информации.

1. Исходный текст программы должен содержать:

- ☐ процедуру заполнения массива размерности N случайными элементами типа Element;
- ☐ процедуру распечатки на экране элементов массива;
- ☐ процедуру, реализующую метод пузырька;
- ☐ процедуру, реализующую метод модифицированного пузырька;
- ☐ процедуру, реализующую метод быстрой сортировки;
- ☐ процедуру, реализующую метод выбора;
- ☐ процедуру, реализующую метод простых вставок;

- ☐ процедуру, реализующую метод Шелла;
 - ☐ процедуру, реализующую метод слияния;
 - ☐ процедуру, реализующую метод подсчета.
2. В основной части программы необходимо организовать:
- ☐ возможность выбора метода сортировки в режиме диалога;
 - ☐ определение времени, затраченного на сортировку;
 - ☐ запись результатов в файл.
3. Проанализировать полученные результаты.

Тема 5. Рекурсивные вычисления

Практическое занятие 1. Разработка и реализация рекурсивных программ. Решение задач с использованием рекурсии.

Практические задания.

1. Определить функцию $S(n)$, вычисляющую сумму цифр заданного натурального числа, и с ее помощью написать соответствующую подпрограмму-функцию.
2. Для целого неотрицательного n вычислить значение функции $f(n)$, определенной следующим образом: $f(0)=0$, $f(1)=1$, при $n>2$ $f(n) = f(n \text{ div } 2) + f(n \text{ div } 3)$.
3. Дать рекурсивное определение суммы и вычислить сумму элементов одномерного массива.
4. Определить, является ли заданная строка палиндромом.
5. Дать рекурсивное определение максимума элементов последовательности и определить максимум одномерного массива.
6. Дать рекурсивное определение возведения числа в степень и описать в программе.
7. Рекурсивно записать число в обратном порядке.
8. Написать подпрограмму вычисления N -го числа Фибоначчи: $F(n)=F(n-1)+F(n-2)$.
9. Рекурсивно определить число ближайшее в последовательности Фибоначчи.
10. Рекурсивно определить наибольший общий делитель двух натуральных чисел: $\text{Нод}(x,y)=x$, если $x=y$, $\text{Нод}(x,y)=\text{Нод}(x-y,y)$, если $x>y$, $\text{Нод}(x,y)=\text{Нод}(x,y-x)$, если $y>x$. Написать подпрограмму вычисления значения $\text{Нод}(x,y)$.
- 11.

Практическое занятие 2. Разработка и реализация рекурсивных программ. Стековый калькулятор. Задача о Ханойских башнях.

Практические задания.

1. Реализовать программу «Стековый калькулятор».
2. Реализовать рекурсивную программу решения задачи о Ханойских башнях.

Практическое занятие 3. Использование рекурсии.

Практические задания.

1. Реализуйте программу для решения задачи о восьми ферзях.
2. Напишите программу для реализации рисования снежинок Коха.
3. Напишите программу реализации задачи «Ход конем». Цель задачи – заставить коня ступить на каждую клетку шахматной доски всего один раз.
4. Реализация методов сортировки с использованием рекурсии. Быстрая сортировка.

Тема 6. Основы визуального программирования в среде Delphi

Лабораторная работа 1. Разработка приложений с использованием простейших компонентов ввода-вывода, управления (Edit, Label, StringGrid, Memo, Button, RadioGroup, RadioButton).

Разработка простейших приложений.

Практические задания.

1. Создать простейшее приложение и разработать его интерфейс: 1) изменить заголовок, размер, цвет и стиль формы; 2) поместить на форму надпись и изменить ее свойства (цвет, шрифт, начертание и размер).

2. При помощи визуальных компонентов создайте приложение «Вычисление арифметических операций».

3. На форме отобразить два компонента StringGrid1 и StringGrid2. В StringGrid1 с клавиатуры ввести значения элементов массива, а в StringGrid2 эти значения отобразить.

Лабораторная работа 2. Основы работы в визуальной среде программирования Delphi. Создание интерфейса программы. Элементы управления для выбора из списков. (ListBox, ComboBox). Элементы UpDown, DateTimePicker и MonthCalendar.

Практические задания.

1. Разработать программу, вводящую вещественные значения a , b , c и выполняющую один из вариантов вычисления: 1) определение максимального из данных чисел; 2) определение минимального из данных чисел в зависимости от выбора пользователя. Выбор варианта вычисления оформить с помощью компонента ListBox. Вывод результата выполнить в поле Label.
2. Разработать программу, вводящую вещественные значения сторон треугольника a , b , c и выполняющую один из вариантов вычисления: 1) определение периметра треугольника $p=a+b+c$; 2) определение площади треугольника $S=\sqrt{k(k-a)(k-b)(k-c)}$, $k=(a+b+c)/2$ в зависимости от выбора пользователя. Выбор варианта вычисления оформить с помощью компонента ComboBox. Вывод результата выполнить в поле Label.
3. Заполнить одномерный массив случайными числами и вывести полученный результат в ListBox.
4. Разработать программу, в которой организовать подсчет количества дней между двумя датами, выбранных из календаря.
5. Разработать программу, в которой организовать ввод мероприятий, намеченных на определенную дату в будущем, и своевременного напоминания о них.

Лабораторная работа 3. Основы работы в визуальной среде программирования Delphi. Разработка приложений. Диалоговые компоненты (OpenDialog, SaveDialog, FontDialog, PrintDialog, MainMenu, PopupMenu).

Практические задания.

1. Разработать приложение с возможностью выбора загружаемого файла, выбора шрифта, настройки принтера, работа с главным и контекстным и другими диалоговыми компонентами.
2. Создать простейший текстовый редактор, позволяющий вводить и редактировать текст, управлять шрифтом и цветом окна, а также открывать и сохранять файлы (аналог стандартного приложения Блокнот).

Лабораторная работа 4. Основы работы в визуальной среде программирования Delphi. Разработка приложений с усложнённым интерфейсом. Многооконные приложения. Модальные и немодальные окна.

Практические задания.

1. Разработать приложение интерфейс которого должен содержать несколько окон, которые должны последовательно менять друг друга при нажатии кнопки Далее. Первая форма должна содержать окно приветствия. Данная форма должна быть главной формой, остальные окна – модальными.
2. Разработать приложение интерфейс которого должен быть организован с использованием компонентов TabControl и PageControl.

Лабораторная работа 5. Основы работы в визуальной среде программирования Delphi. Работа с массивами.

Практические задания.

1. Заполнить одномерный массив случайными числами и вывести полученный результат в строку, в столбец. Заполнить двумерный массив случайными числами и вывести полученный результат.
2. Разработать приложение, в котором можно выбирать различные методы поиска элементов массива (минимальный/максимальный элемент, поиск элемента методом перебора/ дихотомии и т.д.).
3. Заполнить одномерный массив случайными числами. Отсортировать элементы массива методом «Пузырька». Организовать в компоненте Memo вывод исходного массива и вывод содержимого массива в каждом прохождении.

4. Заполнить одномерный массив случайными числами. Отсортировать элементы массива методом Шелла. Организовать в компоненте Метод вывод исходного массива и вывод содержимого массива в каждом прохождении.

5. Разработать приложение, в котором можно выбирать различные методы сортировки для упорядочивания элементов массива.

Лабораторная работа 6. Основы работы в визуальной среде программирования Delphi. Работа с числовыми, текстовыми и файлами записей.

Практические задания.

1. Разработать программу, вычисляющую сумму и среднее арифметическое чисел, хранящихся в указанном пользователе файле данных. Первая строка файла данных содержит единственное число, равное количеству обрабатываемых чисел. При вычислении суммы и среднего арифметического это число не должно учитываться.

2. Дан текстовый файл. Разработать программу, которая подсчитывает количество слов в файле. Программа должна предложить пользователю ввести имя текстового файла в поле ввода и после щелчка по кнопке подсчитать количество слов.

3. Создать файл с типом записи, в котором каждая запись содержит следующие поля о каждом абоненте фамилию, адрес (улица, номер дома), номер телефона. Вывести на экран всю базу данных и информацию об абонентах (фамилия номер телефона), проживающих на определенной улице.

4. Создать файл с типом записи, в котором каждая запись содержит следующие поля об автолюбителях: фамилия, марка автомобиля, цвет, год выпуска. Вывести на экран всю базу данных и информацию о владельцах (фамилия), имеющих автомобиль конкретной марки и определенного цвета.

5. Создать файл с типом записи, в котором каждая запись содержит следующие поля, содержащие фамилии и даты (день, месяц, год) рождения ваших друзей. Вывести на экран всю базу данных и информацию о тех, чье день рождения будет в следующем месяце.

Тема 7. Работа с графикой в среде Delphi

Лабораторная работа 1. Разработка приложений с использованием графических возможностей Delphi.

Практические задания.

1. Разработать программу построения 20 концентрических окружностей разного радиуса, но имеющих общий центр.

2. Разработать программу построения на форме 5 кругов и 5 четырехугольников. Каждый круг и четырехугольник имеет различный цвет, толщину контура, а также различный цвет и стиль заливки.

3. Разработать программу вычерчивания полукруга и квадрата, имеющих общую линию внешнего касания, по заданным радиусу полукруга и длине стороны квадрата. Залить полукруг и квадрат разными цветами. Ввод исходных данных выполнить на основной стартовой форме программы, а вывод графики – на дополнительной форме.

4. Разработать программу построения различных простых фигур (круг, квадрат, прямоугольник и т.д.). Реализовать возможность установки цвета линий и заливки фигур.

Лабораторная работа 2. Построение графиков функций.

Практические задания.

1. Разработать программу построения графика функции $f(x)=\sin(x)+\cos(2x)$ в диапазоне изменения аргумента x от a до b .

2. Разработать программу построения на одной координатной сетке графиков функций: $f_1(x)=2-\cos(x)$; $f_2(x)=2\sin(2x)+1$;

$f_3(x)=2f_1+f_2$ в диапазоне изменения аргумента x от $-?$ до $?$.

Лабораторная работа 3. Построение анимационных изображений.

Практические задания.

1. Разработать приложения имитации капли дождя на поверхности воды.

2. Разработать приложения имитации движения шара по бильярдному столу.

3. Разработать приложения построения траектории движения тела под углом к горизонту. Организовать возможность задания начальной скорости и угла.

4. Разработать приложение демонстрации движения планет солнечной системы.

Лабораторная работа 4. Использование рекурсии в графических построениях.

Практические задания.

1. Разработать рекурсивную программу построения кривой Гильберта.
2. Разработать рекурсивную программу построения Канторова множества.
3. Разработать рекурсивную программу построения кривая Госпера (снежинка Госпера).
4. Разработать рекурсивную программу построения треугольника Серпинского.
5. Разработать рекурсивную программу построения квадрата Серпинского.
6. Разработать рекурсивную программу построения дерева Пифагора.
7. Разработать рекурсивную программу построения кривой Дракона.

Тема 8. Указатели. Динамическая память

Практические занятия.

Лабораторная работа 1. Работа с указателями. Распределение памяти. Динамическое выделение памяти.

Практические задания.

1. Разработать программу демонстрирующую создание, присваивание, сравнение, обнуление (пустой указатель, нулевая ссылка, используется для любого типа), уничтожение динамических переменных.
2. Используя функции и процедуры для работы с памятью и указателями среды программирования Delphi, разработать программу, в которой реализованы следующие функций:
 - a) получить значения указателей полей текущего диспетчера памяти; получить указатель на заданный адрес памяти; получить указатель на участок памяти, которую занимает объект;
 - b) выделить на куче блок памяти заданного размера; выделить блок памяти заданного размера;
 - c) определить размер памяти, занимаемый переменной; определить, используется ли в настоящий момент диспетчер памяти, установленный по умолчанию, или был установлен другой диспетчер;
 - d) изменить размер динамически распределенного блока памяти; выполнить бинарное сравнение двух участков памяти;
 установить значения полей диспетчера памяти.
3. Используя процедуры для работы с динамическими переменными среды программирования Delphi, разработать программу, в которой реализованы следующие функций:
 - создать динамическую переменную;
 - создать динамическую переменную, выделяя под нее заданный объем памяти;
 - инициализировать динамическую переменную;
 - перераспределяет память для динамической переменной;
 - деинициализировать динамическую переменную;
 - высвободить память из-под динамической переменной используя процедуры: Dispose, FreeMem, SysFreeMem.

Лабораторная работа 2. Работа с динамическими массивами.

Практические задания.

1. Сформировать динамический одномерный массив, заполнить его случайными числами и вывести на печать.
2. Сформировать динамический одномерный массив. Удалить элемент равный среднему арифметическому элементов массива.

3. Сформировать динамический одномерный массив. Добавить по K элементов в начало и в конец массива.
4. Сформировать динамический двумерный массив, заполнить его случайными числами и вывести на печать.
5. Сформировать динамический двумерный массив. Добавить K столбцов, начиная со столбца с номером N.
6. Сформировать динамический двумерный массив. Удалить строки, начиная со строки K1 и до строки K2.
7. Сформировать динамический двумерный массив. Добавить столбец после столбца, содержащего наибольший элемент.
8. Разработать процедуру, в которой реализованы следующие действия: создание динамического массива, нахождения суммы элементов массива, уничтожение массива. Используя данную процедуру организовать работу с тремя двумерными массивами.
9. Разработать программу, в которой динамический массив используется для вычисления средней оценки произвольного количества студентов.
10. Разработать программу нахождения произведения элементов, расположенных под главной диагональю массива.

Для решения задачи использовать треугольный динамический массив (в данном массиве один из индексов всегда больше другого или равен ему).

11. Разработать программу, предназначенную для записи и хранения на диске названий и авторов книг личной библиотеки.

В программе использовать два динамических массива: один - для названия книг, другой – для имен и фамилий авторов.

Программа должна позволять пользователю вводить информацию о книге, а также находить или удалять данные о ней. Ограничение на количество книг нет.

12. Компания выпускает тостеры, миксеры и электрические хлеборезки. Каждый проданный тостер приносит компании доход в 100 рублей, каждый миксер – 150 рублей, каждая хлеборезка – 50 рублей. Бухгалтер компании накапливает информацию о продажах за последние три месяца.

Ему нужно вычислить доход за каждый месяц, общий доход за три месяца и средний ежемесячный доход. Бухгалтер выполняет этот расчет ежеквартально.

Разработать программу для выполнения всех расчетов. Для ввода данных использовать текстовый файл, который содержит три названия товаров, доход от продажи каждого товара и объем продаж товара данного вида за каждый из трех месяцев.

Тема 9. Динамические структуры данных

Практическое занятие.

Лабораторная работа 1. Работа с динамическими структурами данных: списки.

Цель: научиться разрабатывать программы создания и обработки динамической структуры данных списки.

Практические задания.

1. Разработать программу создания списка целых чисел и вывести их на экран.
2. Разработать программу реализации типовых операций над списками:
 - создание списка;
 - добавление звена в начало списка;
 - удаление звена из начала списка;
 - добавление звена в произвольное место списка, отличное от начала (например, после звена, указатель на которое задан);
 - удаление звена из произвольного места списка, отличного от начала (например, после звена, указатель на которое задан);
 - проверка, пуст ли список;

- очистка списка;
- печать списка.

3. Дан файл целых чисел, состоящий из нескольких отдельных последовательностей целых чисел, каждая из которых заканчивается нулем.

Ноль не является частью последовательности. Получить новый файл, состоящий из тех же последовательностей,

но внутри каждой числа должны быть записаны в обратном порядке.

Пример:

Дано: 8 10 3 12 0 7 9 2 0 4 12 0

Получить: 12 3 10 8 0 2 9 7 0 12 4 0

Лабораторная работа 2. Работа с динамическими структурами данных: стеки.

Цель: научиться разрабатывать программы создания и обработки динамической структуры данных стеки.

Практические задания.

1. Разработать программу создания стека целых чисел на базе массива и вывести их на экран.
2. Разработать программу создания стека целых чисел на базе списка и вывести их на экран.
3. Разработать программу реализации типовых операций над стеком:

- включение нового элемента;
- исключение элемента из стека;
- чтение элемента из стека;
- очистка стека;
- проверка пустоты стека;
- поиск элемента в стеке;
- определение текущего числа элементов в стеке.

4. Создайте программу, проверяющую скобочную последовательность на наличие ошибок их расстановки.

Лабораторная работа 3. Работа с динамическими структурами данных: очередь.

Цель: научиться разрабатывать программы создания и обработки динамической структуры данных очередь.

Практические задания.

1. Разработать программу создания очереди целых чисел на базе массива и вывести их на экран.
2. Разработать программу создания очереди целых чисел на базе списка и вывести их на экран.
3. Разработать программу реализации типовых операций над очередью:

- включение нового элемента в очередь;
- исключение элемента;
- определение размера очереди;
- очистка очереди;
- поиск элемента в очереди;
- чтение элементов из очереди.

Лабораторная работа 4. Работа с динамическими структурами данных: дек.

Цель: научиться разрабатывать программы создания и обработки динамической структуры данных дек.

Практические задания.

1. Разработать программу создания дека целых чисел на базе одномерного массива и вывести их на экран.
 2. Разработать программу создания дека целых чисел на базе списка и вывести их на экран.
 3. Разработать программу реализации типовых операций над деком:
- добавление элемента в начало дека;
 - добавление элемента в конец дека;
 - извлечение элемента из начала дека;

- извлечение элемента из конца дека;
- определение размера дека;
- проверка пустоты дека;
- поиск элемента в деке;
- очистка дека.

Тема 10. Модули

Практические занятия.

Лабораторная работа 1. Модули. Структура модуля.

Практические задания.

1. Разработать модуль для вычисления расстояний между точками на плоскости и площади треугольника, заданного тремя точками.
2. Разработать модуль для работы со стеком. В данном модуле реализовать основные операции над стеком (включение нового элемента, исключение элемента из стека, чтение элемента из стека, очистка стека, проверка пустоты стека, поиск элемента в стеке, определение текущего числа элементов в стеке).
3. Разработать модуль, реализующий работу с очередью.
4. Разработать модуль для вычисления расстояний в 3-мерном пространстве.
5. Разработать модуль с типами и функциями для вычисления площади поверхности и объема куба.

Лабораторная работа 2. Стандартные модули Delphi.

Практические задания.

1. Разработайте программу демонстрирующую работу с математическими функциями модуля Math.
2. Разработать модуль с дополнительными тригонометрическими функциями.
3. Разработать модуль с типами и функциями для вычисления площади и периметра прямоугольника.
4. Разработайте программу демонстрирующую работу с процедурами и функциями для работы с файлами, дисками, ошибками, строками и т.д. модуля SysUtils.
5. Разработать модуль для работы с матрицами: ввод и вывод значений, проверка на симметричность, заполнение случайными числами и т.д.

Выполнение практических заданий - контрольный срез

Тема 3. Структуры данных в языке Pascal

Практическое задание (контрольный срез). В соответствии с вариантом задания обработать матрицу, элементами которой являются случайные значения функции $F(X)$. Вывести на экран исходную матрицу и полученные результаты.

В программе использовать подпрограммы.

1. Для матрицы $A[1..M, 1..M]$ вычислить сумму элементов верхней треугольной матрицы. Функция: $F(X) = e^X - e^{-X} - 2$ при $-5 < X < 5$.
2. Для матрицы $A[1..M, 1..M]$ вычислить сумму положительных элементов нижней треугольной матрицы. Функция: $F(X) = X + \cos(X^3 + 2)$ при $-5 < X < 5$
3. Для матрицы $A[1..M, 1..N]$ определить и запомнить наибольшие элементы каждой строки. Функция: $F(X) = \sin X - \cos X$ при $-\pi < X < 2\pi$
4. Для матрицы $A[1..M, 1..N]$ вычислить суммы элементов каждой строки матрицы и определить наибольшую из вычисленных сумм. Функция: $F(X) = X^2 - 8$ при $-8 < X < 7$
5. Для матрицы $A[1..M, 1..M]$ определить наименьший элемент, а также номера строки и столбца, в которых он расположен. Функция: $F(X) = X^3 + 1$ при $-8 < X < 9$
6. Дана матрица $A[1..M, 1..N]$. Создать новую матрицу $B[1..N, 1..M]$, поменяв местами строки и столбцы. Функция: $F(X) = e^X \sin X$ при $-\pi < X < \pi$

7. Дана матрица $A[1..M, 1..N]$. Определить наименьший элемент в матрице и записать нули в строку и столбец, на пересечении которых он расположен. Вывести матрицу. Функция: $F(X) = X/13$ при $-50 < X < 75$
8. Дана целочисленная матрица $A[1..M, 1..M]$. Вычислить число отрицательных, положительных и нулевых элементов в ней. Функция: $F(X) = X$ при $-100 < X < 100$
9. Для матрицы $A[1..M, 1..M]$ найти наибольший элемент главной диагонали и вывести на печать строку, в которой он находится. Функция: $F(X) = \sin X + 2\cos X$ при $-\pi < X < \pi$
10. Дана матрица $A[1..M, 1..M]$. Найти наибольший элемент матрицы и отнормировать ее, поделив все элементы матрицы на наибольшее значение. Вывести матрицу. Функция: $F(X) = 5\cos(X+1)$ при $-\pi < X < 3\pi/2$
11. Для матрицы $A[1..M, 1..N]$ найти в каждой строке наибольший элемент и заменить его значением суммы элементов, предшествующих ему в строке. Вывести матрицу. Функция: $F(X) = e^X - e^{-X} - 2$ при $-5 < X < 5$
12. Для матрицы $A[1..M, 1..N]$ найти наименьший из положительных элементов. Функция: $F(X) = X + \cos(X+2)$ при $-5 < X < 5$
13. Для матрицы $A[1..M, 1..N]$ вычислить и запомнить произведения элементов четных строк и суммы элементов нечетных столбцов. Функция: $F(X) = \sin X - \cos X$ при $-\pi < X < 2\pi$
14. В матрице $A[1..M, 1..M]$ найти в каждой строке наибольший элемент и поменять его местами с соответствующим элементом главной диагонали. Вывести матрицу. Функция: $F(X) = X^2 - 8$ при $-8 < X < 7$
15. Для матрицы $A[1..M, 1..M]$ найти наименьший элемент каждого столбца и поменять его местами с соответствующим элементом побочной диагонали; вывести матрицу и сумму элементов побочной диагонали. Функция: $F(X) = X^3 + 1$ при $-8 < X < 9$
16. Вывести сумму и число положительных элементов каждого столбца матрицы $A[1..M, 1..N]$. Функция: $F(X) = e^X \sin X$ при $-\pi < X < \pi$
17. Вычислить сумму и число отрицательных элементов каждой строки матрицы $A[1..M, 1..N]$. Функция: $F(X) = X/13$ при $50 < X < 75$
18. Для целочисленной матрицы $A[1..M, 1..M]$ найти для каждой строки число элементов кратных 5 и наибольший из полученных результатов. Функция: $F(X) = X$ при $-100 < X < 100$
19. Вычислить сумму и число элементов матрицы $A[1..M, 1..M]$, находящихся под побочной диагональю и на ней. Функция: $F(X) = \sin X + 2\cos X$ при $-\pi < X < \pi$
20. Вычислить сумму и число положительных элементов матрицы $A[1..M, 1..M]$, находящихся над побочной диагональю. Функция: $F(X) = 5\cos(X+1)$ при $-\pi < X < 3\pi/2$
21. Записать на место отрицательных элементов матрицы $A[1..M, 1..M]$ нули и вывести ее. Функция: $F(X) = e^X - e^{-X} - 2$ при $-5 < X < 5$
22. Записать на место отрицательных элементов матрицы $A[1..M, 1..M]$ нули, а на место положительных – единицы. Вывести нижнюю треугольную матрицу. Функция: $F(X) = X + \cos(X+2)$ при $-5 < X < 5$
23. Найти наибольший и наименьший элементы матрицы $A[1..M, 1..N]$ и поменять их местами. Вывести матрицу. Функция: $F(X) = \sin X - \cos X$ при $-\pi < X < 2\pi$
24. Найти строки с наибольшей и наименьшей суммой элементов в матрице $A[1..M, 1..N]$. Вывести найденные строки и суммы их элементов. Функция: $F(X) = X^2 - 8$ при $8 < X < 7$
25. Упорядочить по возрастанию элементы каждой строки матрицы $A[1..M, 1..N]$. Вывести полученную матрицу. Функция: $F(X) = X^3 + 1$ при $-8 < X < 9$
26. Упорядочить по убыванию элементы каждого столбца матрицы $A[1..M, 1..N]$. Вывести полученную матрицу. Функция: $F(X) = e^X \sin X$ при $-\pi < X < \pi$
27. Отсортировать по возрастанию элементы главной диагонали матрицы $A[1..M, 1..M]$. Если элемент главной диагонали в строке положительный, то все оставшиеся элементы в строке заменить на 1, в противном случае – на 0. Вывести полученную матрицу. Функция: $F(X) = X/13$ при $-50 < X < 75$

28. Отсортировать по убыванию элементы побочной диагонали матрицы $A[1..M, 1..M]$. Если элемент побочной диагонали в столбце положителен, то все оставшиеся элементы строки заменить на 1, в противном случае – на 0. Вывести полученную матрицу. Функция: $F(X) = X$ при $-100 < X < 100$
29. Для каждой строки матрицы $A[1..M, 1..N]$ определить сумму и число положительных элементов до первого отрицательного. Найти наибольшую и наименьшую из полученных сумм. Функция: $F(X) = \sin X + 2\cos X$ при $-\pi < X < \pi$
30. Найти в каждом столбце матрицы $A[1..M, 1..N]$ наибольший элемент и отсортировать все нижестоящие элементы по убыванию. Вывести полученную матрицу. Функция: $F(X) = 5\cos(X+1)$ при $-\pi < X < 3\pi/2$

Выполнение практических заданий - контрольный срез

Тема 2. Основы программирования на языке Pascal

Практическое задание (контрольный срез). В соответствии с вариантом задания, вычислить значение суммы и ли произведения бесконечного ряда с заданной точностью E для всех значений X, изменяющихся от X _n до X _k с шагом H. Правильность вычисленного выражения в точках X _i проверить с помощью аналитического выражения ряда.									
№	X _n	X _k	H	E	Выражение				
1.	0.9		1	0.1	10.апр Y = ln(1+x)=x-x ² /2+x ³ /3-...+(-1) ⁿ⁻¹ x ⁿ /n+...				
2.		5	10		1	10.мар Y = sinX=x-x ³ /3!+x ⁵ /5!-...+(-1) ⁿ x ²ⁿ⁺¹ /(2n+1)!+...			
3.		3	5	0.2		10.мар Y = cosX=1-x ² /2!+x ⁴ /4!-...+(-1) ⁿ x ²ⁿ /(2n)!+...			
4.		2	10		1	10.апр Y = e-x=1 - x + x ² /2! - ...+(-1) ⁿ x ⁿ /n!+...			
5.		2	8	0.5		10.мар Y =(e ^x -e ^{-x})/2=x+x ³ /3!+x ⁵ /5!+...+x ²ⁿ⁺¹ /(2n+1)!+...			
6.	0.1	0.8		0.1		10.мар Y =arcsinX=x+x ³ /(2*3)+(1*3*x)/(2*4*5)+... +(1*3*5*...*(2n-1))/(2*4*6*...*(2n)*(2n+1))+...			
7.	-0.5	0.5		0.1		10.апр Y =(e ^x +e ^{-x})/2=1+x ² /2!+x ⁴ /4!+...+x ²ⁿ /(2n)!+...			
8.	-0.6	0.3		0.1		10.май Y =ln(1-x)=-x-x ² /2-x ³ /3-...-x ⁿ /n+...			
9.		2	10		1	10.мар Y = arctgX=1/x-1/(3x ³)+...+(-1) ⁿ /((2n+1)x ²ⁿ⁺¹)+...			
10.	-0.6	0.5		0.1		10.мар Y = (1+x) ^m =1+mx+m(m-1)x ² /2!+...+m(m-1)...(m-n+1)x ⁿ /n!+...			
11.	-0.5	0.6		0.1		10.апр Y = (1-x) ^m =1-mx+m(m-1)x ² /2!+...+(-1) ⁿ m(m-1)...(m-n+1)x ⁿ /n!+...			
12.	0.1	0.9		0.1		10.мар Y = (1+x) ^{-m} =1-mx+m(m+1)x ² /2!+...+(-1) ⁿ m(m+1)...(m+n-1)x ⁿ /n!+...			
13.	-0.3	0.4		0.1		10.апр Y = (1-x) ^m =1-mx+m(m+1)x ² /2!+...+m(m+1)...(m+n-1)x ⁿ /n!+...			
14.		3	8		1	10.апр Y = lnX= 2((x-1)/(x+1)+(x-1) ³ /(3(x+1) ³)+...+(x-1) ²ⁿ⁺¹ /((2n+1)(x+1) ²ⁿ⁺¹))+...			
15.	0.6	01.май		0.1		10.апр Y = lnX= (x-1)-(x-1) ² /2+(x-1) ³ /3-... +(-1) ⁿ⁺¹ (x-1) ⁿ /n+...			
16.	01.сен	02.фев	0.05			10.апр Y = lnX= (x-1)/x+(x-1) ² /(2x ²)+(x-1) ³ /(3x ³)+... +(x-1) ⁿ /(nx ⁿ)+...			
17.		3	8		1	10.апр Y = ln((x+1)/(x-1))=2(1/x+1/(3x ³)+...+1/((2n+1)x ²ⁿ⁺¹))+...			
18.	0.1	0.9		0.1		10.мар Y = ln((1+x)/(1-x))=2(x+x ³ /3+x ⁵ /5+... +x ²ⁿ⁺¹ /(2n+1))+...			
19.	0.1	0.8		0.1		10.мар Y = arccosX=p/2-x-x ³ /(2*3)-(1*3*x ⁵)/(2*4*5)+... +1*3*...*(2n-1)x ²ⁿ⁺¹ /(2*4*...*(2n)*(2n+1))+...			
20.	-0.1	0.7		0.1		10.апр Y = arctgX=x-x ³ /3+x ⁵ /5-... +(-1) ⁿ x ²ⁿ⁺¹ /(2n+1)+...			
21.		2	10		1	10.мар Y = arctgX=p/2-1/x+1/(3x ³)-1/(5x ⁵)+...+(-1) ⁿ⁺¹ n!/(2n+1)x ²ⁿ⁺¹ +...			
22.	-0.1	0.7		0.1		10.апр Y = arctgX=p/2-x+x ³ /3-x ⁵ /5+... +(-1) ⁿ 1x ²ⁿ⁺¹ /(2n+1)+...			
23.	0.3	0.9		0.1		10.апр Y = 1/(1-X)=(1+x)(1+x ²)*...*(1+x ²ⁿ)*...			
24.	0	6		1		10.мар Y = sinX=x(1-x ² /p ²)*(1-x ² /(4p ²))*...*(1-x ² /(n ² p ²))*...			
25.	-5	5		1		10.мар Y = cosX=x(1-4x ² /p ²)*(1-4x ² /(9p ²))*...*(1-4x ² /((2n-1) ² p ²))*...			
26.	01.янь	01.авг	0.1			10.апр Y =(e ^x -e ^{-x})/2=x(1+x ² /p ²)*(1+x ² /(4p ²))*... *(1+x ² /(n ² p ²))*...			
27.	01.май	02.мар	0.1			10.мар Y =(e ^x -e ^{-x})/2=(1+4x ² /p ²)*(1+4x ² /(9p ²))*... *(1+4x ² /((2n-1) ² p ²))*...			
28.	0.15	0.85		0.1		10.апр Y = sinX / X = cos(x/2)*cos(x/4)*... *cos(x/2n)*...			
29.	0.9	01.июн	0.1			10.апр Y = sinX / X = (1-4/3sin ² (x/3))*(1-4/3sin ² (x/9))*...*(1-4/3sin ² (x/3 ⁿ))*...			
30.		2	5	0.5		10.мар Y = e ^x = 1 + x + x ² /2! + ...+ x ⁿ /n!+...			

Тема 7. Работа с графикой в среде Delphi

В соответствии с вариантом задания разработать программу, демонстрирующую движение объекта.

1. Разработать программу, демонстрирующую вращение отрезка относительно своей середины.
2. Разработать программу, демонстрирующую вращение равнобедренного треугольника вокруг своей высоты, расположенной параллельно вертикальной оси экрана.
3. Разработать программу, демонстрирующую движение точки по внутреннему ободу колеса.
4. Разработать программу, демонстрирующую вращение квадрата относительно центра и одновременным перемещением вдоль экрана по горизонтали.

5. Разработать программу, демонстрирующую отражение ромба относительно осей OX , OY .
6. Разработать программу, демонстрирующую отражение отрезка относительно осей OX и OY .
7. Разработать программу, демонстрирующую масштабирование отрезка.
8. Разработать программу, демонстрирующую поворот треугольника на заданный угол.
9. Разработать программу, демонстрирующую отражение треугольника относительно осей OX , OY .
10. Разработать программу, демонстрирующую поворот ромба на заданный угол.
11. Разработать программу, демонстрирующую поворот отрезка на заданный угол.
12. Разработать программу, демонстрирующую масштабирование треугольника.
13. Разработать программу, демонстрирующую масштабирование ромба.
14. Разработать программу, демонстрирующую «калейдоскоп».
15. Разработать программу, демонстрирующую вращение правильного треугольника вокруг своего центра.
16. Разработать программу, демонстрирующую вращение прямоугольника вокруг одной из своих вершин.
17. Разработать программу, демонстрирующую закон Архимеда.
18. Разработать программу, демонстрирующую броуновское движение.
19. Разработать программу, демонстрирующую колебание тела.
20. Разработать программу, демонстрирующую вращение рамки в однородном магнитном поле.
21. Разработать программу, демонстрирующую преломление света в различных средах.
22. Разработать программу, демонстрирующую траектории движение искусственного спутника.
23. Разработать программу, демонстрирующую движение тела по наклонной плоскости.
24. Разработать программу, демонстрирующую движение тела по окружности.
25. Разработать программу, демонстрирующую модель атома.
26. Разработать программу, демонстрирующую электролиз.

Тема 9. Динамические структуры данных

Создать линейный односвязный список. Разработать программу обработки линейного односвязного списка в соответствии с вариантом задания.

1. Из списка удалить последнюю пару соседних элементов, сумма которых меньше заданной величины, а затем вставить заданное число перед минимальным элементом списка и после него. Освободить память, занимаемую элементами списка.
2. Из списка удалить элемент, стоящий перед минимальным элементом, затем элементы списка расположить в порядке убывания и вставить заданное число, не нарушая порядка расположения. Освободить память, занимаемую элементами списка.
3. Из списка удалить нулевые элементы, затем вставить заданное число перед максимальным элементом списка и после него. Освободить память, занимаемую элементами списка.
4. Из списка удалить последнюю пару соседних элементов, модуль разности между которыми превышает заданную величину, а затем элементы списка расположить в порядке возрастания и вставить заданное число, не нарушая порядка расположения. Освободить память, занимаемую элементами списка.
5. Из списка удалить положительные элементы, превышающие заданную величину, а затем вставить заданное число перед минимальным положительным элементом списка и после него. Освободить память, занимаемую элементами списка.
6. Из списка удалить минимальный и минимальный положительный элементы, а затем положительные элементы расположить в порядке возрастания и вставить заданное положительное число, не нарушая порядка расположения. Освободить память, занимаемую элементами списка.
7. Из списка удалить элемент, стоящий после минимального элемента, а затем вставить заданное число перед максимальным отрицательным элементом и после него. Освободить память, занимаемую элементами списка.

8. Из списка удалить отрицательные элементы, а затем оставшиеся элементы расположить в порядке убывания и вставить заданное положительное число, не нарушая порядка расположения. Освободить память, занимаемую элементами списка.
9. Из списка удалить положительные элементы, а затем вставить число, равное минимальному элементу списка, перед максимальным элементом и после него. Освободить память, занимаемую элементами списка.
10. Из списка удалить максимальный и максимальный отрицательный элементы, затем отрицательные элементы расположить в порядке возрастания и вставить заданное отрицательное число, не нарушая порядка расположения. Освободить память, занимаемую элементами списка.
11. Из списка удалить элемент, стоящий перед максимальным элементом, а затем вставить заданное число после каждого отрицательного элемента. Освободить память, занимаемую элементами списка.
12. Из списка удалить положительные элементы, не превышающие заданную величину, а затем отрицательные элементы списка расположить в порядке убывания и вставить заданное отрицательное число, не нарушая порядка расположения. Освободить память, занимаемую элементами списка.
13. Из списка удалить максимальный элемент, а затем вставить число, равное минимальному положительному элементу, перед минимальным элементом и после него. Освободить память, занимаемую элементами списка.
14. Из списка удалить элементы между первым и вторым элементами, модули которых меньше заданной величины, а затем расположить элементы в порядке возрастания модулей и вставить заданное число, не нарушая порядка расположения. Освободить память, занимаемую элементами списка.
15. Из списка удалить элемент, стоящий после максимального элемента, а затем перед последним нулевым элементом и после него вставить число, равное последнему отрицательному элементу списка. Освободить память, занимаемую элементами списка.
16. Из списка удалить максимальный элемент, а затем расположить элементы списка в порядке убывания модулей и вставить заданное число, не нарушая порядка расположения. Освободить память, занимаемую элементами списка.
17. Из списка удалить элементы, модули которых меньше заданной величины, затем вставить число, равное последнему положительному элементу, перед первым положительным элементом и после него. Освободить память, занимаемую элементами списка.
18. Из списка удалить элемент, стоящий перед первым нулевым элементом, а затем вставить заданное число перед каждым положительным элементом списка. Освободить память, занимаемую элементами списка.
19. Из списка удалить первые два нулевых элемента, а затем вставить заданное число перед первым положительным элементом и после последнего положительного элемента. Освободить память, занимаемую элементами списка.
20. Из списка удалить максимальный по модулю элемент, а затем вставить число, равное первому положительному элементу, перед последним отрицательным элементом и после него. Освободить память, занимаемую элементами списка.
21. Из списка удалить элементы, стоящие между первым и вторым положительными элементами, а затем вставить число, равное максимальному элементу списка, перед минимальным элементом и после него. Освободить память, занимаемую элементами списка.
22. Из списка удалить максимальный и минимальный элементы, а затем вставить заданное число перед первым отрицательным элементом и после последнего отрицательного элемента. Освободить память, занимаемую элементами списка.
23. Из списка удалить первые два положительных элемента, а затем вставить число, равное минимальному элементу списка, перед максимальным отрицательным элементом и после него. Освободить память, занимаемую элементами списка.
24. Из списка удалить первый отрицательный элемент, а затем вставить заданное число перед первым положительным элементом и после последнего положительного элемента. Освободить память, занимаемую элементами списка.

25. Из списка удалить второй отрицательный элемент, а затем вставить число, равное максимальному элементу списка, перед максимальным по модулю отрицательным элементом и после него. Освободить память, занимаемую элементами списка.
26. Из списка удалить элементы, стоящие между первым и вторым отрицательными элементами, а затем вставить заданное число перед максимальным элементом и после него. Освободить память, занимаемую элементами списка.
27. Из списка удалить последний отрицательный элемент, а затем вставить заданное число перед минимальным элементом списка и после минимального положительного элемента. Освободить память, занимаемую элементами списка.
28. Из списка удалить элементы, стоящие между первым и вторым нулевыми элементами, а затем вставить число, равное последнему элементу списка, перед первым отрицательным элементом и после него. Освободить память, занимаемую элементами списка.
29. Из списка удалить первый положительный элемент, а затем вставить заданное число перед максимальным по модулю элементом списка и после него. Освободить память, занимаемую элементами списка.
30. Из списка удалить первый и последний отрицательные элементы, а затем вставить число, равное второму положительному элементу, перед первым положительным элементом и после него. Освободить память, занимаемую элементами списка.

Опрос

Тема 1. Технология разработки алгоритмов. Решение задач на ЭВМ

1. Дать определения понятиям «алгоритм», «исполнитель», «система команд исполнителя».
2. Каковы свойства алгоритма? Примеры.
3. Какие существуют способы записи алгоритмов?
4. Какой алгоритм называется линейным? Примеры.
5. Какой алгоритм называется разветвляющимся? Примеры.
6. Какой алгоритм называется циклическим? Примеры.
7. Что такое вспомогательный алгоритм? Примеры.
8. Объяснить работу оператора присвоения.
9. Что такое данные, типы данных?
10. Описать виды типов данных.
11. Какие выражения называются арифметическими? Правила их записи.
12. Какие выражения называются логическими? Правила их записи.
13. Описать этапы решения задач на компьютере.
14. В чем заключается технология структурного программирования?
15. Описать технологию программирования «сверху-вниз».
16. Что такое «пошаговая детализация»?
17. Опишите классификацию ошибок, выявляющиеся при тестировании программ?
18. Способы выявления и устранения ошибок при тестировании программ.
19. Какие ошибки обнаруживает транслятор?
20. Опишите процесс устранения алгоритмических ошибок после тестирования программ.

Тема 5. Рекурсивные вычисления

1. Дайте определение рекурсии.
2. Приведите примеры содержательных задач, где для решения может быть использован рекурсивный вспомогательный алгоритм.
3. Что такое граничное условие и каково его назначение в рекурсивной подпрограмме?
4. Охарактеризуйте формы рекурсивных процедур.
5. Что такое рекурсивный спуск? Пример.

6. Что такое рекурсивный подъём? Пример.

7. Что такое глубина рекурсии? Пример.

8. На каком этапе выполнения рекурсивной подпрограммы могут выполняться её операторы?

9. Почему приведённый ниже алгоритм посимвольного формирования строки завершится аварийно?

```
Function Stroka : String;
```

```
Var C : Char;
```

```
Begin
```

```
  Write('Введите очередной символ: '); ReadLn(C);
```

```
  Stroka:=Stroka+C
```

```
End;
```

На каком этапе выполняются действия в этом алгоритме?

Тема 6. Основы визуального программирования в среде Delphi

1. Какие функции выполняет интегрированная среда программирования Delphi? Из каких частей она состоит?

2. Для чего служат визуальные компоненты? Как осуществляется управление визуальными компонентами?

3. Какие функции выполняет свойство компонента? Перечислите типы свойств.

4. Чем отличается работа со свойствами в режиме проектирования и

выполнения программы?

5. Какие типы файлов используются в приложении Delphi?

6. Основное назначение каждого типа файлов в приложении Delphi?

7. Охарактеризуйте обобщенную структуру приложения Delphi.

8. Какие задачи решает Менеджер Проектов?

9. Для чего предназначены окна дерева объектов и инспектора объектов?

10. Перечислите панели инструментов, используемые для разработки программ Delphi.

11. Назначение главного окна Delphi.

12. Назначение конструктора формы.

13. Что такое редактор кода?

14. Объектный инспектор и дерево объектов, их назначение.

15. Насколько важен этап создания интерфейса приложения? Можно ли без него обойтись?

16. Какие средства являются основой визуального программирования в среде Delphi?

17. Как помещаются на форму и настраиваются компоненты?

18. Как создаются обработчики событий?

19. Как осуществляется компиляция и проверка функционирования проекта?

20. Как узнать состав и назначение файлов проекта?

21. Как установить параметры компиляции и сборки проекта?

22. Как создать модальную форму в Delphi - проекте?

23. Как установить основные свойства модальной формы в проекте?

24. Как установить основные события модальной формы в проекте?

25. Как создать немодальную форму в проекте?

26. Что такое управляющие элементы окна, какие они бывают и для чего служат?

27. Назовите общие свойства управляющих элементов.

28. Назовите общие события управляющих элементов.

29. Что такое Палитра Компонентов и как она используется?

30. Как осуществляется управление компонентами в приложении?

31. Как установить основные свойства компонента в приложении?

32. Как установить основные события главной формы в приложении?
33. Что такое компоненты меню, какие они бывают и для чего служат?
34. Что такое свойства пунктов меню, какие они бывают и для чего служат?
35. Что такое Дизайнер Меню, какие функции и как он выполняет?
36. Как создаются пункты меню?
37. Как обрабатываются команды пунктов меню?
38. Как осуществляется синхронизация событий одноименных пунктов главного и всплывающего меню в приложении?
39. Какие основные типы ошибок в программах Вы знаете?
40. Какие средства отладки программ существуют в среде Delphi?
41. Как включить режим отладки?
42. Как установить контрольную точку?
43. Как отладить программу без отладчика?
44. Как разместить компонент в форме?
45. Как изменить размер компонента?

Тема 8. Указатели. Динамическая память

1. Для чего предназначены переменные ссылочного (указательного) типа?
2. Как осуществляется выделение и освобождение динамической памяти?
3. Как осуществляется доступ к динамическим переменным?
4. Что называется указателем?
5. Для чего необходимо значение nil при работе с указателями?
6. Назовите операции, которые допускаются над значениями ссылочного типа?
7. Какие стандартные процедура реализуют основные действия над динамическими переменными?
8. К чему приводит «потеря» указателя на данные, хранимые в динамической памяти?
9. Ограничена ли динамическая память?
10. Как получить указатель на участок памяти, которую занимает объект?
11. Как получить значения указателей полей текущего диспетчера памяти?
12. Как получить указатель на заданный адрес памяти?
13. Как можно выделить на куче блок памяти заданного размера?
14. Как можно выделить блок памяти заданного размера?
15. Как можно изменить размер динамически распределенного блока памяти?
16. Как можно выполнить бинарное сравнение двух участков памяти?
17. Как можно узнать размер памяти, занимаемый переменной?
18. Как можно установить значения полей диспетчера памяти?
19. Как можно определить, используется в настоящий момент диспетчер памяти, установленный по умолчанию, или был установлен другой диспетчер?
20. Как можно создать динамическую переменную?
21. Как можно создать динамическую переменную, выделяя под нее заданный объем памяти?
22. Как можно инициализировать динамическую переменную?
23. Как можно перераспределяет память для динамической переменной?
24. Как можно деинициализировать динамическую переменную?
25. Как можно высвободить память из-под динамической переменной?

Тестирование

Тема 3. Структуры данных в языке Pascal

1. Массив – это:

- 1) упорядоченная последовательность элементов одного и того же типа;
- 2) набор последовательно проиндексированных элементов одного и того же типа;
- 3) конечное упорядоченное множество элементов, которые могут быть различных стандартных типов.

2. Указать ошибку в следующих описаниях:

Const n=50;

Type

vector = array[real] of integer;

number=array[1..n] of char;

var ch: char; a: vector;

- 1) Неправильно описан тип vector;
 - 2) Неправильно описана переменная ch;
 - 3) Неправильно описан тип number.
3. Указать описание массива, содержащего разные символы и индексами которого являются двузначные числа:
- 1) Var vector = array [10..99] of 'a' .. 'z';
 - 2) Var vector = array [10..99] of char;
 - 3) Var vector = array [1..10, 1..99] of char.

4. Дан массив vector : array ['k'..'z'] of real. Указать правильный фрагмент программы, в котором всем элементам массива присваивается нулевое значение:

- 1) For j:='k' to 'z' do vector[j]:=0;
- 2) For j:=1 to 16 do vector[j]:=0;
- 3) For j:=k to z do vector[j]:=0.

5. Дан массив A. Что означает следующая последовательность символов: A[1] ?

- 1) элемент массива A с индексом равным 1;
- 2) переменной A присвоить значение 1;
- 3) первое значение переменной с именем A;
- 4) первая переменная с именем A.

6. Дан массив A. Что означает следующая последовательность символов: A[n] ?

- 1) n-ое значение переменной с именем A;
- 2) переменной A присвоить значение n;
- 3) элемент массива с A с индексом равным n;
- 4) n-ая переменная с именем A.

7. Дан массив Z. Что означает следующая последовательность символов: Z[X]:=Y ?

- 1) Компоненте с номером X массива Z присвоить значение, которое хранится в переменной Y;
- 2) Компоненте с номером Z массива X присвоить значение, которое хранится в переменной Y;
- 3) номеру X массива Z присвоить значение, которое хранится в переменной Y;
- 4) номером Z массива X присвоить значение, которое хранится в переменной Y.

8. Даны массивы Z и Y. Что означает следующая последовательность символов: Z[A]:=Y [A] ?

- 1) Элементу массива A с номером Z присвоить значение, которое хранится в элементе массива A с номером Y;
- 2) Элементу массива Z с номером A присвоить значение, которое хранится в элементе массива Y с номером A;
- 3) Номеру A элементу массива Z присвоить значение, которое хранится в номере A элемента массива Y;
- 4) Номеру Z элемента массива A присвоить значение, которое хранится в номере Y элемента массива A.

9. Даны массивы A и C. Что означает следующая последовательность символов: A[B]:=C[D] ?

- 1) Элементу массива А с номером В присвоить значение, которое хранится в элементе массива С под номером D;
- 2) Элементу массива В с номером А присвоить значение, которое хранится в элементе массива D с номером С;
- 3) Номеру А элементу массива В присвоить значение, которое хранится в номере С элемента массива D;
- 4) Номеру В элемента массива А присвоить значение, которое хранится в номере D элемента массива С.

10. Укажите присвоение К-ому элементу массива Р значение 123.

- 1) K[P]:=123;
- 2) P[K]:=123 ;
- 3) P[K]:='123';
- 4) K[P]:='123'.

11. Могут ли быть элементами некоторого массива следующие значения: False, True ?

- 1) Нет, так как данные значения различных типов;
- 2) Да, так как данные значения одного типа данных;
- 3) Нет, так как вещественные числа не могут быть элементами массива;
- 4) Нет, так как целые числа не могут быть элементами массива.

12. Могут ли быть индексами элементов некоторого массива следующие значения: False, True ?

- 1) Да, так как данные значения одного порядкового типа данных;
- 2) Нет, так как строки не могут быть индексами элементов массива;
- 3) Нет, так как данные значения логического типа;
- 4) Нет, так как нецелые числа не могут быть индексами элементов массива.

13. Как можно описать массив вещественных чисел с нумерацией индексов элементов массива от 1 до 10?

- 1) type Vector=Array[1..10] of real; Var A:Vector;
- 2) Var A: Array[1..10] of integer;
- 3) Var A: Array[1..10] of real;
- 4) Var A: Array[real] of 1..10.

14. Дано следующее описание:

Type Day = Array [1..365] of Integer;

Var D : Day,

Указать сколько элементов в массиве, какой тип его компонент, индекс его первого и последнего элемента.

- 1) 365; Integer; 1; 365
- 2) 364; Day ; 365; 1
- 3) Day; Integer; 1; 365
- 4) 364; Integer; 1; 365

15. Дано следующее описание: Var B : Array [-2..2] of Integer.

Указать сколько элементов в массиве, какой тип его компонент, индекс его первого и последнего элемента.

- 1) 4; Array [-2..2] of Integer; 1; 4
- 2) 5; Integer; -2; 2
- 3) 4; Integer; -2; 2
- 4) 2; Array [-2..2] of Integer; 1; 2

16. Дано следующее описание: Var C : Array [0..9] of Vector.

Указать сколько элементов в массиве, какой тип его компонент, индекс его первого и последнего элемента

- 1) 10; Vector; 0; 9
- 2) 9; 0..9; 1; 9

3) 10; 0..9; 0; 9

4) 9; C : Array [0..9] of Vector; 1; 9

17. Указать ошибки в следующем описании массива:

Type Vector = Array [Real] of Integer.

1) Тип индексов элементов массива не может быть вещественным;

2) Тип элементов массива не может быть вещественным;

3) Тип индексов элементов массива не может быть целым;

4) Тип элементов массива не может быть целым.

18. Дано следующее описание: Type massiv=Array[0..8] of Integer; Var aa: massiv.

Сколько элементов в массиве?

1) 8;

2) 9;

3) 7;

4) 1.

19. Дано следующее описание: Type massiv=Array[-5..5] of Integer; Var aa: massiv.

Сколько элементов в массиве?

1) 11;

2) 10;

3) 5;

4) 12.

20. Какие способы существуют для заполнения массива?

1) ввод значений с клавиатуры, описание типизированной константы-массива;

2) ввод значений с клавиатуры, описание типизированной константы-массива, присвоение в программе элементам массива каких-либо значений, например, случайных;

3) случайными и неслучайными числами;

4) ввод значений с клавиатуры, на экран дисплея, присвоение в программе элементам массива каких-либо значений.

21. Дано следующее описание массива: var A: Array[1..20] of Real.

Укажите фрагмент ввода элементов массива с клавиатуры.

1) For i:=1 to 20 do A[i]:=random(15);

2) Read(A[i]); For i:=1 to 20 do;

3) For i:=1 to 20 do Read(A[i]).

22. Дан фрагмент программы: S:=0; For i:=1 to N do S:=S+A[i].

Данный фрагмент программы служит для нахождения...

1) количества элементов одномерного массива;

2) суммы элементов одномерного массива;

3) суммы нулевых элементов одномерного массива.

23. Дан фрагмент программы:

T:=A[1]; K:=1;

For i:=1 to N do If A[i,j]>T then Begin T:=A[i]; K:=i end.

Данный фрагмент программы служит для нахождения...

1) максимального элемента и его индексов в одномерном массиве;

2) минимального элемента и его индексов в одномерном массиве;

3) максимального элемента в одномерном массиве;

4) минимального элемента в одномерном массиве.

24. Дан фрагмент программы:

T:=A[1]; For i:=1 to N do If A[i]>T then T:=A[i].

Данный фрагмент программы служит для нахождения...

1) минимального элемента в одномерном массиве;

- 2) максимального элемента в одномерном массиве;
- 3) максимального элемента и его индексов в одномерном массиве.

25. Дан фрагмент программы:

$N:=0$; For $j:=1$ to M do If $A[j]>0$ then $N:=N+1$.

Данный фрагмент программы служит для нахождения...

- 1) количества положительных элементов в одномерном массиве;
- 2) количества отрицательных элементов в одномерном массиве;
- 3) количества элементов одномерного массива.

Тема 4. Алгоритмы поиска и простой сортировки

1. Что такое сортировка?

- a) операция, упорядочивающая множество элементов по ключам, по которым определено отношение порядка;
- b) процесс расположения последовательностей в определенном порядке;
- c) процесс группировки элементов, когда нужно выделить все элементы с одинаковыми значениями.

2. Идея алгоритма методом «Пузырька»:

- a) выделение наименьшего элемента и обмен местами его с первым элементом;
- b) многократное сравнение соседних элементов и их перестановка;
- c) просмотр и установка каждого элемента в подходящее место среди уже отсортированных элементов.

3. Идея алгоритма методом «Выбора»:

- a) выделение наименьшего элемента и обмен местами его с первым элементом;
- b) многократное сравнение соседних элементов и их перестановка;
- c) просмотр и установка каждого элемента в подходящее место среди уже отсортированных элементов.

4. Идея алгоритма методом «Простых вставок»:

- a) выделение наименьшего элемента и обмен местами его с первым элементом;
- b) многократное сравнение соседних элементов и их перестановка;
- c) просмотр и установка каждого элемента в подходящее место среди уже отсортированных элементов.

5. В чем заключается сортировка выбором?

- a) Выделяется наименьший или наибольший элемент и ставится на первое или последнее место, затем из оставшихся, выбирается наибольший или наименьший элемент и ставится соответственно на второе или предпоследнее место.
- b) Выделяется произвольный элемент и ставится на первое или последнее место, затем из оставшихся, выбирается произвольный элемент и ставится на второе или предпоследнее место.
- c) Выделяется наименьший элемент и ставится на первое или последнее место, затем из оставшихся, выбирается наибольший элемент и ставится на второе или предпоследнее место.
- d) Выделяется наибольший элемент и ставится на первое или последнее место, затем из оставшихся, выбирается произвольный элемент и ставится соответственно на второе или предпоследнее место.

6. В чем заключается сортировка вставками?

- a) Для некоторых элементов ищется место среди ранее упорядоченных элементов, затем этот элемент ставится в найденную позицию.
- b) Для каждого элемента ищется место среди ранее упорядоченных элементов, затем этот элемент ставится в найденную позицию.
- c) Для каждого элемента ищется произвольное место среди остальных элементов, затем этот элемент ставится в найденную позицию.
- d) Для некоторых элементов ищется место среди всех остальных элементов, затем этот элемент ставится в найденную позицию.

7. Назовите идею сортировки обменом:

- a) Однократное попарное сравнение стоящих рядом элементов массива и их перестановка в заданном порядке.
- b) Многократное попарное сравнение стоящих рядом элементов массива и их перестановка в заданном порядке.
- c) Многократное попарное сравнение стоящих рядом элементов массива и их перестановка в произвольном порядке.
- d) Многократное попарное сравнение произвольных элементов массива и их перестановка в заданном порядке.

8. Из каких операций состоит сортировка вставками?

- a) Выбор любого элемента последовательности, сравнение его с первым элементом, вставка его в соответствующее место.
- b) Сортировка первых (K-1) элементов, вставка K-го элемента в надлежащее место.
- c) Просмотр упорядоченной последовательности для нахождения наибольшего ключа, меньшего или равного данному ключу, вставка новой записи в определенное место упорядоченной последовательности.

9. В информатике «сортировка» - это ...

- a) Процесс расположения элементов только в возрастающем порядке.
- b) Процесс расположения элементов в возрастающем или убывающем порядке.
- c) Процесс расположения элементов только в убывающем порядке.
- d) Процесс расположения элементов в произвольном порядке.

10. Какой метод сортировки запрограммирован?

```
begin
  for i:=2 to n do
    x:=a[i];
    j:=i-1;
    k:=1;
    while j>0 do
      if a[i]>=a[j] then begin
        k:=j+1;
        j:=0;
      end
      else j:=j-1;
    for j:=i downto k+1 do
      begin
        a[j]:=a[j-1];
        a[k]:=x;
      end;
    end;
```

- a) метод обмена;
- b) метод выбора;
- c) метод простых вставок.

11. Какой метод сортировки запрограммирован?

```
for i:=1 to n-1 do
  begin
    min:=x[i];
    nomer:=i;
    for j:=i+1 to n do
      if x[j]<min then begin
        min:=x[j];
        nomer:=j;
```

end;

```
t:=x[i];
x[i]:=min;
x[nomer]:=t;
end;
```

- a) метод обмена;
- b) метод выбора;
- c) метод простых вставок.

12. В чем заключается задача поиска?

- a) в нахождении элемента в соответствующей структуре данных по заданному образцу;
- b) в нахождении элемента массива по заданному образцу;
- c) в нахождении главного элемента в соответствующей структуре данных по заданному образцу.

13. В чем состоит идея бинарного поиска в массиве?

- a) аргумент поиска или запрос сравнивается со средним ключом, результат сравнения приводит к решению задачи;
- b) запрос сравнивается со средним ключом; результат сравнения либо приводит к решению задачи, либо позволяет определить, в какой части массива продолжать поиск;
- c) просматриваются подряд все записи массива до получения решения задачи.

14. Какая задача поиска реализована в следующей процедуре?

```
procedure Poisk (a : massiv; var m : integer);
var i : integer;
Begin
  m:=A[1];
  for i:=1 to n do
    if m<A[i] then m:=A[i];
  end;
```

- a) первого минимального элемента в массиве;
- b) первого максимального элемента в массиве;
- c) последнего минимального элемента в массиве;
- d) последнего максимального элемента в массиве.

15. Какая задача поиска реализована в следующей процедуре?

```
procedure Poisk(a : massiv; var m : integer);
var i : integer;
Begin
  m:=A[1];
  for i:=1 to n do
    if m>A[i] then m:=A[i];
  end;
```

- a) первого минимального элемента в массиве;
- b) первого максимального элемента в массиве;
- c) последнего минимального элемента в массиве;
- d) последнего максимального элемента в массиве.

16. Какая задача поиска реализована в следующей процедуре?

```
procedure Poisk (a : massiv; var m : integer);
var i : integer;
Begin
  m:= A[1];
  for i:=1 to n do
    if m<=A[i] then m:= A[i];
```

end;

- a) первого минимального элемента в массиве;
- b) первого максимального элемента в массиве;
- c) последнего минимального элемента в массиве;
- d) последнего максимального элемента в массиве.

17. Какая задача поиска реализована в следующей процедуре?

```
procedure Poisk (a : massiv; var m : integer);
```

```
  var i : integer;
```

```
  Begin
```

```
    m:= A[1]
```

```
    for i:=1 to n do
```

```
      if m>=A[i] then m:= A[i];
```

```
    end;
```

- a) первого минимального элемента в массиве;
- b) первого максимального элемента в массиве;
- c) последнего минимального элемента в массиве;
- d) последнего максимального элемента в массиве.

18. Вставьте пропущенный фрагмент {****} в процедуре.

```
procedure Poisk (x: massiv; var k:integer);
```

```
  var k,i: integer; min: element;
```

```
  begin
```

```
    min:= x[1]; k:= 1;
```

```
    for i:= 2 to n do
```

```
      if x[i]< min then begin
```

```
        min:= x[i]; {****};
```

```
      end;
```

```
    end;
```

```
  k:=k+1;
```

```
  k:=i;
```

```
  k:=n.
```

19. В результате дихотомического поиска элемента в последовательности границы промежутка поиска...

- a) уменьшаются в два раза;
- b) увеличиваются в два раза;
- c) не изменяются.

20. Бинарный поиск элемента в массиве возможен, если...

массив не отсортирован;

массив отсортирован.

21. Какая задача поиска реализована в следующей процедуре?

```
procedure Poisk (x: massiv; P: element);
```

```
  var i, K: integer;
```

```
  begin
```

```
    K:= 0
```

```
    for i:=1 to n do
```

```
      if x[i]= P then K:= i;
```

```
      if K=0 then write ('нет элемента')
```

```
      else write ('номер =' ,K);
```

```
    end;
```

- a) поиск номера минимального элемента в массиве;

b) поиск элемента в неупорядоченной последовательности;

c) поиск элемента в упорядоченной последовательности.

22. Вставьте пропущенный фрагмент {*****} в процедуру.

procedure dichotom (x: massiv; P: element; var index: integer)

var i, No, Nk : integer;

begin

No:=1; Nk:=n; INDEX:=0;

repeat

i:=(No + Nk) div 2

if x[i]= P then INDEX:= i

else

if x[i]> P then {*****}

else No:=i+1

until (index > 0) or (No > Nk)

end;

a) Nk:= Nk div 2;

b) Nk:= i-1;

c) Nk:= i+1.

23. В чем состоит идея поиска элемента в массиве простым перебором?

a) аргумент поиска или запрос сравнивается со средним ключом, результат сравнения приводит к решению задачи;

b) последовательным сравнением элементов массива с образцом до тех пор, пока не будет найден элемент, равный образцу, или не будут проверены все элементы;

c) последовательным сравнением всех элементов массива друг с другом до тех пор, пока не будет найден искомый элемент.

24. В чем состоит идея поиска максимального элемента в массиве с различными элементами?

a) Сначала делается предположение, что первый элемент массива является максимальным, затем остальные элементы массива последовательно сравниваются с этим элементом. Если во время очередной проверки обнаруживается, что проверяемый элемент больше принятого за максимальный, то этот элемент становится максимальным и продолжается проверка оставшихся элементов.

b) Сначала делается предположение, что первый элемент массива является максимальным, затем остальные элементы массива последовательно сравниваются с первым элементом. Если во время очередной проверки обнаруживается, что проверяемый элемент больше первого, то этот элемент становится максимальным и продолжается проверка оставшихся элементов.

25. Укажите действия, выполняющиеся в процессе поиска элемента с заданным значением в упорядоченной последовательности?

a) В процессе поиска осуществляется последовательное сравнение элементов массива с образцом до тех пор, пока не будет найден элемент, равный образцу, или не будут проверены все элементы. Когда обнаруживается искомый элемент, запоминается его номер.

b) В процессе поиска границы промежутка сдвигаются друг к другу, причем после каждого сравнения изменяется только одна граница: либо верхняя, либо нижняя. Промежуток последовательно уменьшается в 2 раза.

Тема 9. Динамические структуры данных

1. Структуры данных делятся на:

1) Файлы и данные статической структуры

2) Данные статической структуры и данные динамической структуры

3) Данные статической структуры и файлы

2. К простым данным статической структуры не относятся:

1) записи

- 2) файлы
- 3) массивы
3. К данным стандартных типов не относятся:
 - 4) символ
 - 5) список
 - 6) логические
4. К арифметическим типам не относятся:
 - 1) Integer
 - 2) Boolean
 - 3) Real
5. Составные данные статической структуры делятся на:
 - 1) однородные и простые
 - 2) однородные и неоднородные
 - 3) скалярные и неоднородные
6. К однородным типам не относятся:
 - 1) строки
 - 2) записи
 - 3) множества
7. К неоднородным типам не относятся:
 - 1) записи
 - 2) массивы
 - 3) объекты
8. К однородным типам не относят:
 - 1) множества
 - 2) объекты
 - 3) строки
9. Данные динамической структуры делятся на:
 - 1) Файлы, несвязанные динамические данные, составные данные
 - 2) Файлы, несвязанные динамические данные, связанные динамические данные
 - 3) Массивы, несвязанные динамические данные, связанные динамические данные
10. Данные, имеющие дополнительную связь между последним и первым элементами списка, - это:
 - 1) очередь
 - 2) кольцевые списки
 - 3) стек
11. Частный случай линейного односвязного списка, для которого разрешены только два действия: добавление элемента в конец списка и удаление элемента из начала списка, - это:
 - 1) кольцевой список
 - 2) очередь
 - 3) дек
12. Частный случай линейного односвязного списка, для которого разрешено добавлять или удалять элементы только с одного конца списка, который называется вершиной, - это:
 - 1) очередь
 - 2) стек
 - 3) дек
13. Последовательность элементов, в которой новые данные могут быть присоединены с обеих сторон и также с обеих можно удалять ее элементы, - это:
 - 1) стек
 - 2) дек
 - 3) очередь

14. Какую динамическую структуру можно представить в виде трубы с запаянным концом?

- 1) стек
- 2) дерево
- 3) очередь
- 4) таблица

15. Укажите недопустимую операцию со стеком

- 1) проверить пуст стек или нет
- 2) поменять два элемента стека местами
- 3) сделать стек пустым
- 4) взять элемент из стека

16. Какому закону подчиняются элементы стека?

- 1) последним вошел - первым вышел
- 2) последним вошел - последним вышел
- 3) первым вошел - первым вышел
- 4) порядок входа и выхода произвольный

17. С применением какой структуры можно решать задачу о ханойских башнях?

- 1) очереди
- 2) стека
- 3) дерева
- 4) таблицы

18. Последовательность элементов, образующаяся в порядке их поступления - это

- 1) очередь
- 2) стек
- 3) дек
- 4) дерево

19. Какая структура данных подчиняется закону: первым вошел - первым вышел?

- 1) дерево
- 2) стек
- 3) очередь
- 4) таблица

20. Как двигаются элементы очереди?

- 1) в одном направлении
- 2) в противоположных направлениях
- 3) произвольно
- 4) элементы не двигаются

21. Укажите недопустимую операцию с очередями

- 1) добавить элемент в конец очереди
- 2) сделать очередь пустой
- 3) добавить элемент в начало очереди
- 4) взять элемент из начала очереди

22. Укажите данные, имеющие иерархическую структуру

- 1) стек
- 2) дерево
- 3) очередь
- 4) таблица

23. На какие структуры можно отображать стек?

- 1) вектор, список
- 2) вектор, список, очередь
- 3) вектор, очередь

4) список, очередь, дерево

24. Разновидностью какой структуры являются структуры стек и очередь

1) вектор

2) дерева

3) линейного списка

25. Какое название имеет линейный список, в котором добавления и исключения возможны только в конце?

1) очередь

2) стек

3) дерево

4) вектор

26. Какое название имеет линейный список, в котором элементы добавляются только в конец, а исключаются только из начала?

1) вектор

2) стек

3) очередь

4) дерево

27. Очередь, стек, дек, список относятся к...

1) кольцевым данным

2) линейным данным

3) разветвляющимся данным

4) несвязным данным

4.3 Промежуточная аттестация по дисциплине проводится в форме экзамена

Типовые вопросы экзамена (ОПК-7)

1. Языки программирования: понятие, классификация. Сравнение языков программирования.
2. Системы программирования: понятие, виды. Трансляторы (компиляторы и интерпретаторы).
3. Этапы решения задач на ЭВМ. Техника редактирования алгоритмов и программ. Устранение синтаксических ошибок. Тестирование программ и устранение алгоритмических ошибок.
4. Понятие алгоритма, исполнителя алгоритма. Основные свойства алгоритмов.
5. Способы записи алгоритмов. Типовые алгоритмические структуры. Примеры.
6. Основные понятия языка Pascal: алфавит, лексемы, имена, служебные слова.
7. Структура программ на языке Pascal. Примеры.
8. Описания в языке Pascal: типы, переменные, константы. Запись арифметических и логических выражений в языке Pascal.
9. Данные. Переменные и константы. Типы данных. Стандартные и программируемые типы.
10. Операторы присвоения, ввода, вывода, комментария в языке Pascal: синтаксис, свойства и область применения.
11. Условный оператор и оператор выбора в языке Pascal: синтаксис, свойства и область применения.
12. Операторы цикла в языке Pascal: синтаксис, свойства, принцип работы и область применения.
13. Функции в языке Pascal: синтаксис, свойства и область применения.
14. Процедуры в языке Pascal: синтаксис, свойства и область применения.
15. Структуры данных: понятие, классификация, общая характеристика. Примеры.
16. Базовые типы данных языка Pascal (целые, вещественные, символьные, логические).
17. Перечислимый тип данных в языке Pascal. Ограниченный тип данных в языке Pascal.
18. Массивы в языке Pascal.
19. Множественный тип данных в языке Pascal.

20. Строковый тип в языке Pascal. Стандартные подпрограммы обработки строк. Примеры.
21. Комбинированный тип данных (записи) в языке Pascal. Примеры.
22. Файловый тип данных: числовые, текстовые файлы, файлы записей. Стандартные процедуры и функции обработки. Примеры.
23. Обзор методов сортировки. Сравнительный анализ.
24. Сортировка выбором. Алгоритм. Примеры.
25. Сортировка вставками. Простые вставки. Алгоритм. Примеры.
26. Сортировка вставками. Метод Шелла. Алгоритм. Примеры.
27. Сортировка обменом. Метод Пузырька. Алгоритм. Примеры.
28. Сортировка обменом. Метод Модифицированного пузырька. Алгоритм. Примеры.
29. Сортировка обменом. Метод Быстрой сортировки. Алгоритм. Примеры.
30. Сортировка методом Подсчета. Алгоритм. Примеры.
31. Сортировка методом слияния.
32. Обзор методов поиска информации. Сравнительный анализ.
33. Поиск минимального (максимального) элемента. Алгоритм. Примеры.
34. Поиск элемента с заданным значением. Поиск в неупорядоченной последовательности. Алгоритм. Примеры.
35. Поиск элемента с заданным значением. Поиск в упорядоченной последовательности. Дихотомический поиск. Алгоритм. Примеры.
36. Примеры алгоритмов решения задач.
37. Рекурсия. Реализация. Формы рекурсивных записей. Примеры. Задача о Ханойских башнях.
38. Визуальная среда программирования Delphi. Возможности, аппаратные и программные требования.
39. Организация проекта в Delphi. Основные файлы проектов.
40. Модули. Структура модулей. Типы модулей в Delphi: модули данных, модули динамических объектов, пакеты, модули потока команд.
41. Форма, свойства и события формы.
42. Компоненты ввода и отображения текстовой информации (Edit, Label, ListBox, StringGrid, Memo и т.д.): назначение, свойства, методы.
43. Кнопки, индикаторы, управляющие элементы (Button, RadioGroup, RadioButton и т.д.): назначение, свойства, методы.
44. Создание и управление меню. Применение диалоговых компонентов. Примеры.
45. Стандартные события (события мыши, клавиатуры, системные события) визуальных компонент.
46. Графические возможности среды Delphi. Построение графических изображений.
47. Динамическая память. Виды памяти. Правила работы с динамической памятью. Операции NEW и DELETE. Динамические массивы.
48. Указатели, динамическое выделение памяти в языке Pascal.
49. Динамические структуры данных. Описание. Примеры.
50. Списки. Реализация списков. Примеры.
51. Абстрактный тип данных. Стеки. Реализация стеков (через массив, список). Примеры.
52. Абстрактный тип данных. Очереди. Реализация очереди (через массив, список). Примеры.
53. Абстрактный тип данных. Деки. Реализация дека. Примеры.

Типовые задания для экзамена (ОПК-7)

- 1 Разработать программу, содержащую процедуру сортировки последовательности по убыванию методом простых вставок.
- 2 Разработать программу, содержащую процедуру поиска элемента бинарным способом.
- 3 Дано вещественное X , целое N . Вычислить сумму: $\sin(X) * \sin(3X) * \dots * \sin((2N-1)*X)$.

- 4 Разработать программу, которая находит самую большую цифру целого числа.
- 5 Разработать программу, которая определяет количество вхождений данного символа в строку.
- 6 Разработать программу, которая выделяет из множества целых чисел от 1 до 30 множество чисел, кратных 2
- 7 Разработать программу, которая разбивает массив на два массива: в первом - отрицательные элементы из исходного массива, а во втором – положительные. Относительный порядок элементов сохранить.
- 8 Разработать программу построения графических изображений.
- 9 Разработать программу демонстрирующую работу с динамическими массивами.
- 10 Разработать программу демонстрирующую работу со списками.
- 11 Разработать программу демонстрирующую работу со стеками.

4.4. Шкала оценивания промежуточной аттестации

Оценка	Компетенции	Дескрипторы (уровни) – основные признаки освоения (показатели достижения результата)
«отлично» (85 - 100 баллов)	ОПК-7	Способен разрабатывать алгоритмы и программы, пригодные для практического применения
«хорошо» (70 - 84 баллов)	ОПК-7	Способен разрабатывать алгоритмы и программы, пригодные для практического применения, испытывает затруднения в разработке алгоритмов и программ
«удовлетворительно» (50 - 69 баллов)	ОПК-7	В целом способен решать задачи профессиональной деятельности, испытывает затруднения в разработке алгоритмов и программ
«неудовлетворительно» (менее 50 баллов)	ОПК-7	Не способен разрабатывать алгоритмы и программы, пригодные для практического применения

5. Методические указания для обучающихся по освоению дисциплины (модуля)

5.1 Методические указания по организации самостоятельной работы обучающихся:

Приступая к изучению дисциплины, в первую очередь обучающимся необходимо ознакомиться содержанием рабочей программы дисциплины (РПД), которая определяет содержание, объем, а также порядок изучения и преподавания учебной дисциплины, ее раздела, части.

Для самостоятельной работы важное значение имеют разделы «Объем и содержание дисциплины», «Учебно-методическое и информационное обеспечение дисциплины» и «Материально-техническое обеспечение дисциплины, программное обеспечение, профессиональные базы данных и информационные справочные системы».

В разделе «Объем и содержание дисциплины» указываются все разделы и темы изучаемой дисциплины, а также виды занятий и планируемый объем в академических часах.

В разделе «Учебно-методическое и информационное обеспечение дисциплины» указана рекомендуемая основная и дополнительная литература.

В разделе «Материально-техническое обеспечение дисциплины, программное обеспечение, профессиональные базы данных и информационные справочные системы» содержится перечень профессиональных баз данных и информационных справочных систем, необходимых для освоения дисциплины.

5.2 Рекомендации обучающимся по работе с теоретическими материалами по дисциплине

При изучении и проработке теоретического материала необходимо:

- просмотреть еще раз презентацию лекции в системе MOODLe, повторить законспектированный на лекционном занятии материал и дополнить его с учетом рекомендованной дополнительной литературы;
- при самостоятельном изучении теоретической темы сделать конспект, используя рекомендованные в РПД источники, профессиональные базы данных и информационные справочные системы;
- ответить на вопросы для самостоятельной работы, по теме представленные в пункте 3.2 РПД.

- при подготовке к текущему контролю использовать материалы фонда оценочных средств (ФОС).

5.3 Рекомендации по работе с научной и учебной литературой

Работа с основной и дополнительной литературой является главной формой самостоятельной работы и необходима при подготовке к устному опросу на семинарских занятиях, к дебатам, тестированию, экзамену. Она включает проработку лекционного материала и рекомендованных источников и литературы по тематике лекций.

Конспект лекции должен содержать реферативную запись основных вопросов лекции, в том числе с опорой на размещенные в системе MOODLe презентации, основных источников и литературы по темам, выводы по каждому вопросу. Конспект может быть выполнен в рамках распечатки выдачи презентаций лекций или в отдельной тетради по предмету. Он должен быть аккуратным, хорошо читаемым, не содержать не относящуюся к теме информацию или рисунки.

Конспекты научной литературы при самостоятельной подготовке к занятиям должны содержать ответы на каждый поставленный в теме вопрос, иметь ссылку на источник информации с обязательным указанием автора, названия и года издания используемой научной литературы. Конспект может быть опорным (содержать лишь основные ключевые позиции), но при этом позволяющим дать полный ответ по вопросу, может быть подробным. Объем конспекта определяется самим студентом.

В процессе работы с основной и дополнительной литературой студент может:

- делать записи по ходу чтения в виде простого или развернутого плана (создавать перечень основных вопросов, рассмотренных в источнике);
- составлять тезисы (цитирование наиболее важных мест статьи или монографии, короткое изложение основных мыслей автора);
- готовить аннотации (краткое обобщение основных вопросов работы);
- создавать конспекты (развернутые тезисы).

5.4. Рекомендации по подготовке к отдельным заданиям текущего контроля

Собеседование предполагает организацию беседы преподавателя со студентами по вопросам практического занятия с целью более обстоятельного выявления их знаний по определенному разделу, теме, проблеме и т.п. Все члены группы могут участвовать в обсуждении, добавлять информацию, дискутировать, задавать вопросы и т.д.

Устный опрос может применяться в различных формах: фронтальный, индивидуальный, комбинированный. Основные качества устного ответа подлежащего оценке:

- правильность ответа по содержанию;
- полнота и глубина ответа;
- сознательность ответа;
- логика изложения материала;
- рациональность использованных приемов и способов решения поставленной учебной задачи;
- своевременность и эффективность использования наглядных пособий и технических средств при ответе;
- использование дополнительного материала;
- рациональность использования времени, отведенного на задание.

Устный опрос может сопровождаться презентацией, которая подготавливается по одному из вопросов практического занятия. При выступлении с презентацией необходимо обращать внимание на такие моменты как:

- содержание презентации: актуальность темы, полнота ее раскрытия, смысловое содержание, соответствие заявленной темы содержанию, соответствие методическим требованиям (цели, ссылки на ресурсы, соответствие содержания и литературы), практическая направленность, соответствие содержания заявленной форме, адекватность использования технических средств учебным задачам, последовательность и логичность презентуемого материала;
- оформление презентации: объем (оптимальное количество), дизайн (читаемость, наличие и соответствие графики и анимации, звуковое оформление, структурирование информации, соответствие заявленным требованиям), оригинальность оформления, эстетика, использование возможности программной среды, соответствие стандартам оформления;

- личностные качества: ораторские способности, соблюдение регламента, эмоциональность, умение ответить на вопросы, систематизированные, глубокие и полные знания по всем разделам программы;
- содержание выступления: логичность изложения материала, раскрытие темы, доступность изложения, эффективность применения средств ИКТ, способы и условия достижения результативности и эффективности для выполнения задач своей профессиональной или учебной деятельности, доказательность принимаемых решений, умение аргументировать свои заключения, выводы.

6. Учебно-методическое и информационное обеспечение дисциплины

6.1 Основная литература:

1. Программирование на языке Delphi : лабораторный практикум: в 2 ч., Ч.1. - Тамбов: Изд-во ТГУ, 2010. - 116 с.
2. Программирование на языке Delphi : лабораторный практикум : в 2 ч., Ч.2. - Тамбов: [Издат. дом ТГУ], 2011. - 192 с.
3. Программирование на языке Delphi : лабораторный практикум : в 2 ч., Ч.2. - Тамбов: [Издат. дом ТГУ], 2011. - 192 с.
4. Клыгина Е.В. Основы алгоритмизации и программирования для студентов-заочников специальностей "Математика", "Физика" : учеб. пособие. - Тамбов: [Издат. дом ТГУ им. Г.Р. Державина], 2013. - 152 с.

6.2 Дополнительная литература:

1. Клыгина Е.В., Кузнецов В.В., Кузьмина Н.В. Программирование на языке Паскаль. Ч.1 : Учеб. пособие. - Тамбов: ТГУ, 2004. - 112с.
2. Бойцов Э.А. Алгоритмы и структуры данных : учеб. пособие для вузов. - Тамбов: Изд-во ТГУ, 2010. - 51 с.
3. Вирт, Никлаус Алгоритмы и структуры данных. - 2024-10-28; Алгоритмы и структуры данных. - Саратов: Профобразование, 2019. - 272 с. - Текст : электронный // IPR BOOKS [сайт]. - URL: <http://www.iprbookshop.ru/88753.html>
4. Сальников Ю.Н. Программирование. Базовый курс : учеб. пособ.. - М.: Маркет ДС, 2010. - 335 с.
5. Лопатин Д.В. Объектно-ориентированное программирование : Учебно-метод. пособие. - Тамбов: Изд-во ТГУ, 2001. - 77 с.

6.3 Иные источники:

1. Федеральный портал «Российское образование» - <http://www.edu.ru/>
2. Федеральное хранилище «Единая коллекция цифровых образовательных ресурсов» - <http://school-collection.edu.ru/>
3. Национальный Открытый Университет «ИНТУИТ» - <http://www.intuit.ru/>
4. Уроки Delphi начинающим с нуля - <http://www.delphi-manual.ru/>
5. Программирование - это просто - <http://easyprog.ru>
6. Языки программирования. Delphi. Уроки Delphi - <http://delphi-prg.ru/category/delphi-7-uroki>

7. Материально-техническое обеспечение дисциплины, программное обеспечение, профессиональные базы данных и информационные справочные системы

Для проведения занятий по дисциплине необходимо следующее материально-техническое обеспечение: учебные аудитории для проведения занятий лекционного и семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, помещения для самостоятельной работы.

Учебные аудитории и помещения для самостоятельной работы укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Помещения для самостоятельной работы укомплектованы компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду Университета.

Для проведения занятий лекционного типа используются наборы демонстрационного оборудования, обеспечивающие тематические иллюстрации (проектор, ноутбук, экран/ интерактивная доска).

Лицензионное и свободно распространяемое программное обеспечение:

Delphi 2007 for Win32 Professional

Microsoft Office Профессиональный плюс 2007

Профессиональные базы данных и информационные справочные системы:

1. Электронный каталог Фундаментальной библиотеки ТГУ. – URL: <http://biblio.tsutmb.ru/elektronnyij-katalog>
2. Университетская библиотека онлайн: электронно-библиотечная система. – URL: <https://biblioclub.ru>
3. Консультант студента. Гуманитарные науки: электронно-библиотечная система. – URL: <https://www.studentlibrary.ru>
4. Научная электронная библиотека eLIBRARY.ru. – URL: <https://elibrary.ru>
5. Российская государственная библиотека. – URL: <https://www.rsl.ru>
6. Российская национальная библиотека. – URL: <http://nlr.ru>
7. Президентская библиотека имени Б.Н. Ельцина. – URL: <https://www.prlib.ru>
8. Научная электронная библиотека Российской академии естествознания. – URL: <https://www.monographies.ru>
9. Электронная библиотека РФФИ. – URL: <https://www.rfbr.ru/rffi/ru/library>

Электронная информационно-образовательная среда

https://auth.tsutmb.ru/authorize?response_type=code&client_id=moodle&state=xyz

Взаимодействие преподавателя и студента в процессе обучения осуществляется посредством мультимедийных, гипертекстовых, сетевых, телекоммуникационных технологий, используемых в электронной информационно-образовательной среде университета.