

**Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Национальный исследовательский университет «МЭИ»**

Направление подготовки/специальность: 09.03.01 Информатика и вычислительная техника

Наименование образовательной программы: Вычислительно-измерительные системы

Уровень образования: высшее образование - бакалавриат

Форма обучения: Очная

**Оценочные материалы
по дисциплине
Программирование**

**Москва
2024**

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ РАЗРАБОТАЛ:

Разработчик

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Гречкина П.В.
Идентификатор	R1bd1b949-ShutovaPV-762e3978	

П.В. Гречкина

СОГЛАСОВАНО:

Руководитель
образовательной
программы

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Хвостов А.А.
Идентификатор	Rd7c1e2e7-KhvostovAA-a55ec66d	

А.А. Хвостов

Заведующий
выпускающей
кафедрой

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Самокрутов А.А.
Идентификатор	R145b9cc2-SamokrutovAA-7b5e7df	

А.А.
Самокрутов

ОБЩАЯ ЧАСТЬ

Оценочные материалы по дисциплине предназначены для оценки достижения обучающимися запланированных результатов обучения по дисциплине, этапа формирования запланированных компетенций и уровня освоения дисциплины.

Оценочные материалы по дисциплине включают оценочные средства для проведения мероприятий текущего контроля успеваемости и промежуточной аттестации.

Формируемые у обучающегося компетенции:

1. ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения

ИД-1 Демонстрирует знание основных языков программирования и работы с базами данных, операционных систем и оболочек, современных программных сред разработки информационных систем и технологий

ИД-2 Применяет языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ

ИД-3 Применяет методы и технологии отладки и оптимизации программного обеспечения

2. ОПК-9 Способен осваивать методики использования программных средств для решения практических задач

ИД-1 Демонстрирует знание методов и способов использования программных средств для решения практических задач

ИД-2 Использует программные средства для решения практических задач

и включает:

для текущего контроля успеваемости:

Форма реализации: Защита задания

1. КМ1-8 Итоговое индивидуальное задание (1 семестр). Выполнение и защита типового расчёта. (Расчетно-графическая работа)
2. КМ2-2 Связный линейный список (Лаб..3 , Паскаль) (Домашнее задание)
3. КМ2-4 Рекурсия (Лаб.9-10, C|C++) (Домашнее задание)
4. КМ2-5 Итоговое индивидуальное задание (2 семестр) (Расчетно-графическая работа)

Форма реализации: Компьютерное задание

1. КМ1-1 Лабораторная работа 1 (1 семестр). ПО для языка Паскаль (Лабораторная работа)
2. КМ1-4 Обработка массивов (лаб.5,7-8, Паскаль) (Программирование (код))
3. КМ1-5 Лабораторная работа 9 (1 семестр). Исключения и итерационные циклы (Программирование (код))
4. КМ1-6 Процедуры и функции, их параметры (лаб.6,10,11, Паскаль) (Домашнее задание)
5. КМ2-1 Закрепление (Лаб.1-2, Паскаль) (Программирование (код))

Форма реализации: Письменная работа

1. КМ1-2 Спецификация (Лаб2-4, Паскаль) (Домашнее задание)
2. КМ1-3 Контрольная работа 1 (1 семестр) (Контрольная работа)
3. КМ1-7 Контрольная работа 2 (1 семестр). Процедуры и параметры. (Контрольная работа)

Форма реализации: Смешанная форма

1. КМ2-3 Знакомство с C/C++ (Лаб.4-8, язык Си) (Лабораторная работа)

БРС дисциплины

1 семестр

Перечень контрольных мероприятий текущего контроля успеваемости по дисциплине:

- КМ-1 КМ1-1 Лабораторная работа 1 (1 семестр). ПО для языка Паскаль (Лабораторная работа)
- КМ-2 КМ1-2 Спецификация (Лаб2-4, Паскаль) (Домашнее задание)
- КМ-3 КМ1-3 Контрольная работа 1 (1 семестр) (Контрольная работа)
- КМ-4 КМ1-4 Обработка массивов (лаб.5,7-8, Паскаль) (Программирование (код))
- КМ-5 КМ1-5 Лабораторная работа 9 (1 семестр). Исключения и итерационные циклы (Программирование (код))
- КМ-6 КМ1-6 Процедуры и функции, их параметры (лаб.6,10,11, Паскаль) (Домашнее задание)
- КМ-7 КМ1-7 Контрольная работа 2 (1 семестр). Процедуры и параметры. (Контрольная работа)
- КМ-8 КМ1-8 Итоговое индивидуальное задание (1 семестр). Выполнение и защита типового расчёта. (Расчетно-графическая работа)

Вид промежуточной аттестации – Экзамен.

Раздел дисциплины	Веса контрольных мероприятий, %								
	Индекс КМ:	КМ-1	КМ-2	КМ-3	КМ-4	КМ-5	КМ-6	КМ-7	КМ-8
	Срок КМ:	2	6	8	9	10	13	14	15
Алгоритмизация и программирование									
Алгоритм и данные. Основы технологии разработки программ. Простые типы данных	+	+	+						+
Структурированный тип данных – статический массив. Алгоритмы обработки массивов, композиция алгоритмов. (Паскаль/Delphi)			+	+	+	+		+	+
Структурированные типы данных в Паскаль/Delphi: запись и множество									+
Технологии разработки, тестирование и отладка, исключительные ситуации									
Текстовый файл и обработка исключительных ситуаций (Паскаль/Delphi)						+			+
Отладка и тестирование. Критерии качества программных продуктов.							+		+
Процедурное программирование на Паскаль/Delphi									
Процедуры и модули(unit) в языке Паскаль/Delphi							+	+	+
Вес КМ:	4	11	10	13	9	18	20	15	

2 семестр

Перечень контрольных мероприятий текущего контроля успеваемости по дисциплине:

- КМ-1 КМ2-1 Закрепление (Лаб.1-2, Паскаль) (Программирование (код))
- КМ-2 КМ2-2 Связный линейный список (Лаб..3 , Паскаль) (Домашнее задание)
- КМ-3 КМ2-3 Знакомство с С|С++ (Лаб.4-8, язык Си) (Лабораторная работа)
- КМ-4 КМ2-4 Рекурсия (Лаб.9-10, С|С++) (Домашнее задание)
- КМ-5 КМ2-5 Итоговое индивидуальное задание (2 семестр) (Расчетно-графическая работа)

Вид промежуточной аттестации – Экзамен.

Раздел дисциплины	Веса контрольных мероприятий, %					
	Индекс КМ:	КМ-1	КМ-2	КМ-3	КМ-4	КМ-5
	Срок КМ:	3	6	11	14	15
Динамические типы данных в Паскаль/Delphi и нетекстовые файлы						
Статические и динамические строки и массивы. Типы с управляемым временем жизни в Паскаль/Delphi.	+					+
Типизированный и бестиповой файлы в Паскаль/Delphi	+					+
Указатели в Delphi. Моделирование собственного типа динамического массива и связанного списка.		+				+
Процедурное программирование на С С++						
Структура простой программы на С				+		
Функции, параметры, прототипы и модули в С С++				+		+
Текстовые и двоичные файлы, структуры, объединения, строки в С				+		+
Методы уточнения корня на отрезке. Указатель на функцию в С				+		
Рекурсивные алгоритмы						
Рекурсия. Рекурсивный алгоритм, дерево вызовов, глубина, объем, стек вызовов.					+	+
Моделирование типов						
Информационно-логические структуры: стек, очередь, дек, бинарное дерево. Их моделирование					+	+
Моделирование целого типа большого диапазона						+
Вес КМ:		12	10	35	18	25

СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ТЕКУЩЕГО КОНТРОЛЯ

I. Оценочные средства для оценки запланированных результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций

Индекс компетенции	Индикатор	Запланированные результаты обучения по дисциплине	Контрольная точка
ОПК-8	ИД-1 _{опк-8} Демонстрирует знание основных языков программирования и работы с базами данных, операционных систем и оболочек, современных программных сред разработки информационных систем и технологий	Знать: типы данных, их структуры и подходящие алгоритмы их обработки процедурное программирование на двух языках высокого уровня - Паскаль/Delphi и C C++ Уметь: разрабатывать алгоритм и записывать программу на языке процедурного программирования (Паскаль/Delphi)	КМ-3 КМ1-3 Контрольная работа 1 (1 семестр) (Контрольная работа) КМ-7 КМ1-7 Контрольная работа 2 (1 семестр). Процедуры и параметры. (Контрольная работа) КМ-13 КМ2-5 Итоговое индивидуальное задание (2 семестр) (Расчетно-графическая работа)
ОПК-8	ИД-2 _{опк-8} Применяет языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач	Знать: алгоритмы простых внутренних сортировок Уметь: разрабатывать простые алгоритмы на поиск экстремальных значений и их позиции разрабатывать простые алгоритмы на поиск количества, суммы,	КМ-3 КМ1-2 Спецификация (Лаб2-4, Паскаль) (Домашнее задание) КМ-4 КМ1-4 Обработка массивов (лаб.5,7-8, Паскаль) (Программирование (код)) КМ-6 КМ1-6 Процедуры и функции, их параметры (лаб.6,10,11, Паскаль) (Домашнее задание) КМ-9 КМ2-1 Закрепление (Лаб.1-2, Паскаль) (Программирование (код)) КМ-10 КМ2-2 Связный линейный список (Лаб..3 , Паскаль) (Домашнее задание) КМ-11 КМ2-3 Знакомство с C C++ (Лаб.4-8, язык Си) (Лабораторная работа)

	<p>различных классов, ведения баз данных и информационных хранилищ</p>	<p>произведения разрабатывать алгоритмы с вложенными друг в друга простыми алгоритмами и циклами разрабатывать алгоритмы по их словесному описанию и использовать указатели на функции в языке C/C++ работать со строками на языке C/C++, разделять текст на символы, слова и предложения базовыми функциями работать с указателями, выделять, освобождать память и создавать динамические связанные списки на языке Паскаль/Delphi повторно использовать процедуры с параметрами, отличать и описывать процедуры общего вида и функции на языке Паскаль/Delphi обрабатывать строки и множества на языке Паскаль/Delphi моделировать информационно-логическую структуру</p>	<p>KM-12 KM2-4 Рекурсия (Лаб.9-10, C/C++) (Домашнее задание)</p>
--	--	---	--

		<p>Стек (LIFO) на основе динамического связного списка на языке C/C++ использовать и редактировать типизированные файлы из записей на языке Паскаль/Delphi выделять подзадачи и разрабатывать алгоритмы на проверку наличия или отсутствия подходящих по условию значений выделять подзадачи и описывать функции и их прототипы, вызывать функции и передавать параметры по значению, адресу и ссылке на языке C/C++ выделять подзадачи и описывать процедуры и их параметры, вызывать процедуры на языке Паскаль/Delphi ставить и уточнять задачу и выбирать подходящие типы и структуры данных и разрабатывать интерфейс консольного приложения разрабатывать алгоритмы для обработки двоичных файлов и структур данных</p>	
--	--	---	--

		на языке C C++	
ОПК-8	ИД-3 _{ОПК-8} Применяет методы и технологии отладки и оптимизации программного обеспечения	<p>Знать:</p> <p>управляющие структуры языка программирования, в том числе итерационные циклы с несколькими возможными условиями выхода, для написания более эффективного программного кода</p> <p>Уметь:</p> <p>разрабатывать рекурсивные алгоритмы и строить полное дерево вызовов, высчитывать глубину вызовов и объем рекурсии</p> <p>применять технологию обработки исключительных ситуаций для адекватного ответа на ошибки времени выполнения программы и использовать итерационные циклы со сложным условием выхода</p>	<p>КМ-5 КМ1-5 Лабораторная работа 9 (1 семестр). Исключения и итерационные циклы (Программирование (код))</p> <p>КМ-12 КМ2-4 Рекурсия (Лаб.9-10, C C++) (Домашнее задание)</p>
ОПК-9	ИД-1 _{ОПК-9} Демонстрирует знание методов и способов использования программных средств для решения практических задач	<p>Знать:</p> <p>функциональные возможности программных средств по созданию многомодульных проектов и отладке программ на процедурном языке</p>	<p>КМ-3 КМ1-2 Спецификация (Лаб2-4, Паскаль) (Домашнее задание)</p> <p>КМ-8 КМ1-8 Итоговое индивидуальное задание (1 семестр). Выполнение и защита типового расчёта. (Расчетно-графическая работа)</p>

		<p>программирования, о возможности пошагового выполнения программ, о возможности передаче параметров программе для функционального и структурного тестирования (на основе черного и белого ящиков)</p> <p>Уметь: использовать в программе стандартные и нестандартные текстовые файлы для ввода, вывода и хранения информации</p>	
ОПК-9	ИД-2 _{ОПК-9} Использует программные средства для решения практических задач	<p>Знать: как вычислять значения стандартных математических функций, используя современное средство разработки программ на языке Паскаль/Delphi</p> <p>Уметь: использовать современное программное средство разработки программ для решения поставленной задачи на языке C/C++</p>	<p>КМ-1 КМ1-1 Лабораторная работа 1 (1 семестр). ПО для языка Паскаль (Лабораторная работа)</p> <p>КМ-11 КМ2-3 Знакомство с C/C++ (Лаб.4-8, язык Си) (Лабораторная работа)</p>

II. Содержание оценочных средств. Шкала и критерии оценивания

1 семестр

КМ-1. КМ1-1 Лабораторная работа 1 (1 семестр). ПО для языка Паскаль

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 4

Процедура проведения контрольного мероприятия: Студент должен уметь работать со средой разработки программ: создать новый проект консольного приложения, набрать предложенную программу на языке Паскаль, сохранить ее, записать в программу свою формулу на языке программирования, откомпилировать, запустить и проверить результат при контрольном значении, отладить, закрыть и открыть для проверки преподавателем.

Краткое содержание задания:

- Следуя указания преподавателя, запустить среду разработки программ (PascalABC или Lazarus), ознакомиться с интерфейсом;
- Создать первое консольное приложение для изучения вывода числовых и строковых констант и форматов их вывода, следуя указаниям преподавателя; отладить и сохранить программу в отдельной папке; закрыть все файлы проекта.
- - Создать в стандартной программе для редактирования текста «Блокнот» файл с кодом программы для первой лабораторной, Сохранить его в новой папке, созданной на жестком диске, и открыть в среде разработки; отладить и сохранить программу.
- - Изменить арифметическое выражение в последнем проекте, закодировав функцию для своего варианта. Проверить правильность с помощью контрольного значения при $x = 0,5$.

Таблица для кодирования функций:

$5x \rightarrow 5^x$	$\pi \rightarrow pi$	$e^x \rightarrow \exp(x)$	$\text{ArcTg } x \rightarrow \arctan(x)$	$\sqrt[n]{x^m} \rightarrow \exp(n/m * \ln(x))$,
$\frac{2}{3} \rightarrow 2/3$	$ x \rightarrow \text{abs}(x)$	$\ln x \rightarrow \ln(x)$	$\sin x \rightarrow \sin(x)$	если $x > 0$
$1 \frac{2}{3} \rightarrow 1+2/3$	$x^2 \rightarrow \text{sq}(x)$	$\text{Log}_x \rightarrow \ln(x)/\ln(n)$	$\cos x \rightarrow \cos(x)$	если $\sqrt[n]{x}$, то $n=1$
	$\sqrt{x} \rightarrow \text{sqrt}(x)$	$\frac{a+b}{c} \rightarrow (a+b)/c$	$\text{tg } x \rightarrow \sin(x)/\cos(x)$	если $ x ^n$, то $m=1$

Примеры функций по вариантам (1-7 и 16-21):

№	Формула	Значение при $x = 0,5$	№	Формула	Значение при $x = 0,5$
1	$\sqrt{e^{2,2x}} - \left \sin \frac{\pi x}{x + \frac{2}{3}} \right + 1,7$	2.45833	16	$\sqrt{e^{2x} \sqrt{x} - \frac{x-1}{2}} \cdot \left \cos 2,5x \right $	0.39745
2	$\sqrt{\sqrt{x^2} + \sqrt{x^{4-x}} + \ln x-20,5 }$	4.08657	17	$\frac{[7,2-10x]}{\sqrt{\frac{x}{9} + e^{2x}}} \cdot \text{arctg} \frac{4\text{tg } 2x}{\sqrt{1,1x^3}}$	2.36642
3	$\left(\frac{1}{7} + \ln \sqrt{x} \right) e^{\sqrt{ x-3 }}$	-0.69331	18	$\frac{x^3}{\sqrt{3}} - e^x \ln 1,37^x + x^3 + \frac{4}{3}$	-0.22986
4	$\frac{\sqrt{x} \sin \frac{x^2}{2} - 1,3}{\sqrt[3]{x} + e^{3x} + \cos x }$	-0.19452	19	$\frac{\ln \sqrt{\pi + 2-x }}{3 - \frac{1}{x}} + \sqrt[3]{x^2} \sin 1,4x$	1.17336
5	$\sqrt{e^{\ln 4}} + 2 \ln 3x - \frac{1}{9}$	1.97070	20	$\left(\sqrt{\ln^2 x + \text{tg} \cos \pi} \right) \left \ln \frac{x}{10,5} \right $	2.38453
6	$\left(\sqrt{1+x^2} + \frac{ \ln^3 x }{1,6+x^4} \right) \sin 7x$	-0.46245	21	$\frac{\sin(0,5\pi x) + \sqrt[3]{x}}{\sqrt{ \cos(\pi x) } + e^{\sqrt{x}} }$	0.77789
7	$\frac{\sqrt{\frac{1}{5}} + \sqrt{e^x}}{ \ln x^2 - 1,3 }$	0.42529	22	$\frac{1}{\sqrt{x}} + \ln^2 0,2 + \sin x \cdot \sqrt{x^2}$	1.50832

Контрольные вопросы/задания:

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
Знать: как вычислять значения стандартных математических функций, используя современное средство	1. Можно ли создать файл проекта консольного приложения вне среды разработки приложений? Как? (1)

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
разработки программ на языке Паскаль/Delphi	<p>2.Как запустить программу? Каким пунктом меню можно воспользоваться? Есть ли соответствующая кнопка на панели инструментов? Есть ли быстрые клавиши для запуска? (1)</p> <p>3.Можно ли и как откомпилировать проект, но не запускать? Можно ли проверить синтаксис, но не запускать? (1)</p> <p>4.В чем отличия между процедурами вывода Write и WriteLn? (1)</p> <p>5.Как вывести два целых числа в одной строке? Как вывести их в двух последовательных строках? (1)</p> <p>6.Как описать простую переменную вещественного типа с плавающей точкой? Как запросить у пользователя ее значение? (1)</p> <p>7.Как указать при выводе числа с плавающей точкой, что надо ограничиться двумя первыми знаками из дробной части, округлив число? (1)</p> <p>8.Предложите способ возведения в целую степень отрицательного числа. (1)</p> <p>9.Как задержать до нажатия клавиши ENTER окно консольного вывода для просмотра выведенной в него информации? (1)</p> <p>10.Поясните указанную преподавателем строку своей программы. (1)</p>

Описание шкалы оценивания:

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: Оценка "отлично" выставляется, если задание выполнено в полном объеме или выполнено преимущественно верно и не содержит лишних действий, и студент легко отвечает на вопросы по программе. Недочеты исправлены в указанные сроки с первой попытки

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: Оценка "хорошо" выставляется, если задание выполнено в полном объеме и выполнено преимущественно верно, и может ответить на заданный контрольный вопрос, но не сделал приглашения к вводу, либо не вывел пояснения к выводу, либо не указал формат вывода результата, либо не смог или не захотел без преподавателя протестировать и сравнить результат функции с контрольным значением при 0,5. И для устранения недочета требуется не более двух попыток

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется, если задание преимущественно выполнено, но есть несколько замечаний (см. про "хорошо") и для устранения недочетов требуется более двух попыток, либо не на все вопросы студент может ответить, либо время контрольного мероприятия просрочено на четыре недели и более по неуважительной причине

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: Оценка "неудовлетворительно" выставляется, если задание выполнено неверно или преимущественно не выполнено, или студент не понимает собственный сдаваемый код и не может ответить на большинство контрольных вопросов. Если студент не показал ни разу программу, то "0".

КМ-2. КМ1-2 Спецификация (Лаб2-4, Паскаль)

Формы реализации: Письменная работа

Тип контрольного мероприятия: Домашнее задание

Вес контрольного мероприятия в БРС: 11

Процедура проведения контрольного мероприятия: Рассчитать оценку за выполнение и защиту Лабораторных работ 2-4, направленных поэтапное решение и доработку общей задачи и единой спецификации. Обсуждение в течении 5-10 минут написанной студентом спецификации к лабораторным 2-4 (1 семестр) с тремя версиями программы, допущенных в спецификации ошибок и путей ее устранения. Работа по требованию преподавателя может быть заранее сдана на проверку на бумажном носителе для сокращения времени обсуждения на занятии.

Краткое содержание задания:

Часть.1. Лабораторная 2 .Выделение входных и выходных данных, уточнение постановки задачи и таблицы данных, форм ввода и вывода.

Разработать нисходящим способом алгоритм, отделив ввод и вывод от её решения, и написать программу на Паскале, создав консольное приложение для MS Windows, для решения задачи из нижеприведенного перечня задач. При этом:

- 1) Ввод исходных данных выполнить с клавиатуры, не забывая о приглашениях к вводу.
- 2) Вывод исходных данных и результатов выполнить на экран вывода консольного приложения, не забывая о пояснениях.
- 3) Все задачи в первом семестре выполнять только с использованием статических (не динамических открытых) массивов.
- 4) При наличии альтернативных решений, особенно отрицательных, предусмотреть вывод соответствующих сообщений. Например, «Невозможно найти среднее значение среди положительных элементов, т.к. таких элементов в массиве нет».
- 5) Код решения задачи пока не писать, заменив временно простейшей **заглушкой** – прямым присваиванием значения результатам.
- 6) Создайте (рукописную или в электронном виде) спецификацию задачи для уровня абстракций А0. Для этого:
 - а) Приведите условие задачи из перечня задач без каких-либо изменений;
 - б) Уточните условие задачи, добавив типы, имена и структуры исходных и данных и результатов, не указанные в условии, и предусмотрите вывод сообщений при невозможности найти положительное решение;
 - в) Продумайте пример, демонстрирующий порядок решения задачи при положительном решении; или несколько примеров, демонстрирующих различные альтернативные решения задачи; г) Заполните таблицу данных, выделив исходные данные и результаты, и вспомогательные переменные; продумайте, какими должны быть диапазоны исходных данных, чтобы при решении задачи не возникла ситуация переполнения и не потерялась задуманная точность решения, но при этом они были достаточно разнообразными для всесторонней демонстрации работы алгоритма.
 - д) Аномальные ситуации можно не рассматривать, особенно, если ранее не изучали язык Pascal или Delphi, или рассмотреть лишь частично.
 - е) Продумать порядок ввода исходных данных и текст приглашений к вводу, а также порядок вывода исходных данных и результатов (при разных альтернативных решениях, если такие есть), и отразить их в виде форм ввода и вывода;

- ж) Составить функциональные тесты для проверки правильности работы программы для исходных данных, лежащих внутри выбранных диапазонов, и, особенно, на их границах, а также для разных альтернативных решений и экстремальных и средних значениях выходных данных и промежуточных результатов;
- з) Изобразите в виде блок-схемы порядок выполнения ввода и вывода, выделив подзадачу(и) решения в отдельный блок, для которого известны пока только имена входных и выходных данных;
- и) Напишите код программы;
- к) Структурные тесты здесь и далее (до выполнения типового расчета) можно не составлять.

Примеры задач по вариантам:

1. Найти сумму и число тех элементов заданного массива X_1, X_2, \dots, X_n , которые попадают на заданный отрезок.
2. Подсчитать по отдельности суммы $S1$ и $S2$ и количества $M1$ и $M2$ отрицательных и положительных элементов заданного одномерного массива.
3. Выделяя из заданных элементов X_1, X_2, \dots, X_n положительные элементы, для которых к тому же справедливо равенство $\sin X_i \geq 0$, найти число и произведение такого рода элементов.
4. Найти сумму и общее количество тех элементов заданного массива X_1, X_2, \dots, X_n , абсолютная величина которых отличается от P не более, чем на T (P и T – заданные величины).
5. Для заданного массива X_1, X_2, \dots, X_n найти среднее арифметическое элементов, имеющих четные номера и притом положительных, а для заданного массива Y_1, Y_2, \dots, Y_n найти среднее арифметическое элементов, имеющих нечетные номера и притом отрицательных.
6. При заданных элементах X_1, X_2, \dots, X_n найти по отдельности суммы $S1$ и $S2$ и количества $M1$ и $M2$ элементов, значения которых соответственно больше P и меньше $-P$.
7. При заданных X_1, X_2, \dots, X_n и Y_1, Y_2, \dots, Y_n , проверяя на равенство элементы пар $(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)$, подсчитать число случаев равенства элементов пар; одновременно найти среднее арифметическое элементов X_1, X_2, \dots, X_n .
8. Вычислить куб суммы и число тех элементов заданного массива X_1, X_2, \dots, X_n , значения которых меньше R или находятся в пределах от T до P .

Часть 2. Лабораторная работа 3. Алгоритм и блок-схема, тесты

Доработать алгоритм и программу из Лабораторной работы №2, подставив вместо заглушки реальный алгоритм поиска решения. При этом:

- 1) Ввод исходных данных выполнить с клавиатуры, не забывая о приглашениях к вводу.
- 2) Вывод исходных данных и результатов выполнить на экран вывода консольного приложения, не забывая о пояснениях.
- 3) Доработайте спецификацию задачи, дополнив ее спецификациями для подзадач(и) решения в сокращенном виде (только описание метода, алгоритм в виде блок-схемы подзадач(и) и программный код, который следует добавить в программу вместо заглушки.
- 4) Проверьте работу программы, составив функциональные тесты, и внесите при необходимости изменения в программу при необходимости.

Часть 3. Лабораторная работа 4. Текстовый файл

Изменить ранее разработанную в Лабораторной работе №3 программу таким образом, чтобы ввод всех исходных данных осуществлялся из нестандартного текстового файла, и вывод также перенаправить в нестандартный текстовый файл. При этом:

- 1) Ввод исходных данных выполнить из нестандартного текстового файла.
- 2) Вывод исходных данных и результатов также выполнить в нестандартный текстовый файл.
- 3) Для взаимодействия с файлами здесь и далее используйте собственные файловые переменные, а не стандартные input и output.
- 4) Проверьте работу измененной программы на ранее составленных тестах.

Контрольные вопросы/задания:

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
Уметь: разрабатывать простые алгоритмы на поиск количества, суммы, произведения	1. Почему за начальное значение при поиске суммы всех элементов одномерного массива можно принять ноль, а при поиске произведения – единицу? (1-3) 2. Почему при объединении кода алгоритмов поиска количества (K) и суммы (S) четных элементов

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
	<p>массива В из N элементов (см. код ниже) нужны операторные скобки Begin..End? Что будет, если их убрать? Почему в этом примере нет необходимости ставить операторные скобки вокруг тела цикла for (после второй строки Begin и после седьмой строки ещё один End)? Фрагменты кода примера:</p> <p>Поиск суммы: S:=0; For i:=1 to N do If B[i] mod 2 = 0 then S:= S + B[i];</p> <p>Поиск количества: K:=0; For i:=1 to N do If B[i] mod 2 = 0 then K:= K + 1;</p> <p>Поиск суммы и количества: S:=0; K:=0; For i:=1 to N do If B[i] mod 2 = 0 then Begin S:= S + B[i]; K:= K + 1 End;</p> <p>(1-3) 3.Поясните указанную преподавателем строку своей программы или часть блок-схемы. (1-3)</p>
<p>Уметь: ставить и уточнять задачу и выбирать подходящие типы и структуры данных и разрабатывать интерфейс консольного приложения</p>	<p>1.Становится ли результатом работы программы исходное данное, если его вывести на экран перед результатами работы программы? (1-2) 2.Как определить диапазон значений суммы элементов массива, если известно, что в массиве может храниться до 20 значений с диапазонами от -100 до +100? (1-2) 3.Как определить диапазон значений количества положительных элементов массива, если известно, что в массиве может храниться до 20 значений с диапазонами от -100 до +100? (1-2) 4.Как ввести значения двух целочисленных переменных в одной строке через пробел при вводе с клавиатуры? Как ввести из разных строк? (1-2) 5.Как ввести только первые значения из двух последовательно введенных с клавиатуры наборов целочисленных значений, разделенных пробелами? Например, ввести 10 и 20 из 10 11 12 13 -10 20 21</p>

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
	<p>(1-2)</p> <p>6. В чем отличия между процедурами ввода Read и ReadLn? 7. Что такое отрицательное решение? Положительное? Альтернативное? Есть ли такие решения в вашей задаче? (1-2)</p> <p>7. Что такое аномальная ситуация? Приведите пример условия возникновения аномальной ситуации в вашей задаче. (1-2)</p> <p>8. Ситуация, когда невозможно найти среднее значение отрицательных элементов в одномерном массиве из 10 элементов с диапазонами значений [-1000, +1000], это аномалия, ошибка или отрицательное решение? (1-2)</p> <p>9. Какого рода ошибок можно избежать, если не только вводить, но и выводить исходные данные? (1-2)</p> <p>10. Нужен ли алгоритм или программа для составления функциональных тестов? (1-2)</p> <p>11. Поясните указанную преподавателем строку своей спецификации. (1-2)</p>
<p>Уметь: использовать в программе стандартные и нестандартные текстовые файлы для ввода, вывода и хранения информации</p>	<p>1. Каковы недостатки использования стандартных переменных input и output для организации ввода и вывода из/в нестандартные текстовые файлы? (1-4)</p> <p>2. Как описать файловую переменную для последующей работы с ее помощью с собственным текстовым файлом? Как связать ее с этим файлом? (1-4)</p> <p>3. Можно ли при связывании файловой переменной с текстовым файлом указать только имя файла, не указывая путь к нему? Какой путь будет подразумеваться по умолчанию в первую очередь? (1-4)</p> <p>4. Какие есть способы открытия текстового файла в Паскале? Чем они отличаются? (1-4)</p> <p>5. Как закрыть текстовый файл в Паскале? Зачем это делать? (1-4)</p> <p>6. Как связана форма ввода из спецификации к задаче с тем, что надо разместить в текстовом файле, из которого будет осуществляться ввод исходных данных? (1-4)</p> <p>7. Как связана форма вывода из спецификации к задаче с тем, что будет выведено в файл в результате работы программы? (1-4)</p> <p>8. Как создать текстовый файл с помощью программы, написанной на Паскале? (1-4)</p>

Описание шкалы оценивания:

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: Если минимальная оценка по Лаб.2-4 это "5".
Оценка по Лаб.2-4: оценка "отлично" выставляется, если задание выполнено в полном

объеме или выполнено преимущественно верно и не содержит лишних действий, и студент легко отвечает на вопросы по программе и спецификации. Недочеты исправлены в указанные сроки с первой попытки

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: Если минимальная из трёх оценок по Лаб.2-4 это "4". Оценка "хорошо" по Лаб.2-4 выставляется, если задание выполнено в полном объеме и выполнено преимущественно верно, и может ответить преимущественно верно на заданные контрольные вопросы, но не все недочеты устранены на момент простановки оценки. И для устранения недочетов потребовалось не более двух попыток

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: Если минимальная из трёх оценок по Лаб.2-4 это "3". Оценка "удовлетворительно" по Лаб.2-4 выставляется, если задание преимущественно выполнено, но есть несколько замечаний и для устранения недочетов потребовалось более двух попыток и не на все вопросы студент может ответить, либо время сдачи работы просрочено на три недели и более по неуважительной причине

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: Если одна из Лаб. работ 2-4 не выполнена вовсе, либо выполнена на "2", при этом "2" - минимальная из трёх оценок. Оценка "неудовлетворительно" по Лаб.2-4 выставляется, если задание выполнено неверно или преимущественно не выполнено, или студент не понимает собственный сдаваемый код и не может ответить на большинство контрольных вопросов. Если студент не показал ни разу спецификацию или программу, то "0".

КМ-3. КМ1-3 Контрольная работа 1 (1 семестр)

Формы реализации: Письменная работа

Тип контрольного мероприятия: Контрольная работа

Вес контрольного мероприятия в БРС: 10

Процедура проведения контрольного мероприятия: Перед началом контрольной работы студентам напоминаются логические операции и их обозначение в формулах и на языке программирования Delphi, приоритет выполнения операций, таблица истинности. Разбирается пример создания логического выражения для описания области, ограниченной прямыми, окружностями и эллипсами. Напомнить формулы прямых, окружностей и эллипсов. Затем на 45 минут дается письменное задание, которое надо написать темной яркой авторучкой на светлом чистом листе бумаги. Работа сдается на проверку преподавателю не позднее 45 минут от получения заданий.

Краткое содержание задания:

Задание состоит из двух частей.

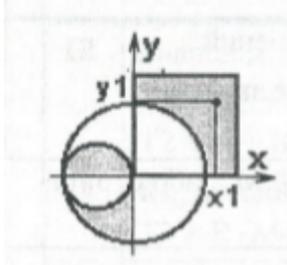
Задача №1. Логические выражения

а) Запишите их по правилам языка программирования Delphi.
 б) Установить истинность (ложность) приводимых ниже логических выражений при заданных значениях.

a	b	NOT a НЕ a, -a отрицание	a AND b a И b, a ∧ b одновременно	a OR b a ИЛИ b, a ∨ b хотя бы одно	a XOR b "Исключающее ИЛИ" только одно
True	True	False	True	True	False
True	False	True	False	True	True
False	True	True	False	True	True
False	False	True	False	False	False

Задача №2. Проверка попадания точки в заданную область

Заданы два одномерных массива X и Y из N элементов хранящих координаты N точек на плоскости.

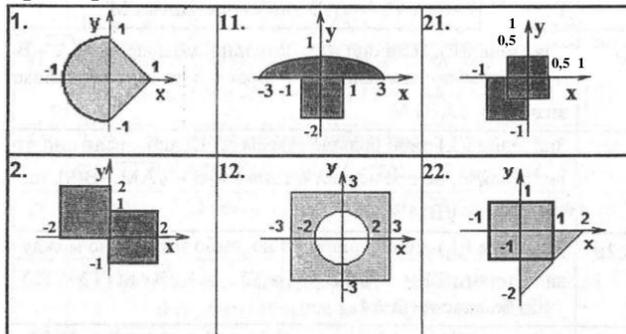


1. Записать логическое выражение, соответствующее области истинности.
2. Составить программу для подсчета количества попавших в заданную область.

Примеры вариантов функций для задания 1:

1. $(x \leq 2) \vee (2y = 10) \wedge (x/4 \neq 0,25) \vee (xy \geq 0)$ при $x = 1, y = 20$.	16. $a \vee \neg(b \vee d) \vee (p \neg q < 0) \wedge (q \neq 0) \vee (a \wedge b \vee d)$ при $a = \text{false}, b = \text{false}, d = \text{true}, p = 2, q = 0$.
2. $\neg(a > b) \vee (b > c) \vee (b/c > 3) \wedge \neg(b + c = 1)$ при $a = 1, b = 2, c = 3$.	17. $\neg(x + y > 2) \vee (z \neq 0) \wedge (x + 0,25 \leq 2) \vee \neg(x = y)$ при $x = 1,5, y = 1, z = 0$.
3. $(x + y \neq 3) \vee \neg(x/y \leq 2) \vee (z = 3x) \wedge (y > x + 1)$ при $x = 1, y = 2, z = 3$.	18. $c \vee (b \wedge c) \wedge (p + t > q) \vee (p + t - 2q < 0)$ при $a = \text{true}, b = \text{true}, c = \text{false}, p = 3, t = 5, q = 7$.
4. $(x/a = 1) \vee (b/(a+b) < 1) \wedge (b = a) \vee (x \neq 6)$ при $x = 6, a = 2, b = 3$.	19. $\neg a \vee b \vee (d + f \neq 0) \wedge \neg(d > 0) \vee (a \vee c \wedge b)$ при $a = \text{true}, b = \text{false}, c = \text{true}, d = -1, f = 1$.

Примеры областей для задания 2:



Контрольные вопросы/задания:

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
Знать: типы данных, их структуры и подходящие алгоритмы их обработки	<ol style="list-style-type: none"> 1. В задании 1а) Как записать это логическое выражение на Delphi? Как пишутся знаки сравнения, деления, умножения? Когда нужны круглые скобки? 2. В задании 1б) Какой порядок выполнения логических операций? И каков их результат? 3. В задании 2а) Сколько границ у закрашенной области? Какими неравенствами можно описать область и какой(какими) логической операцией их соединить? 4. В задании 2б) Как описать одномерные статические массивы в языке Delphi? Какой тип элементов подойдет, чтобы хотя бы десяток точек можно было найти внутри области, а не на ее границах? Как ввести значение N и значения координат N точек двух статических массивов X, Y? Отличается ли ввод значений массива от ввода значений простой переменной? Как найти количество подходящих точек, координаты которых хранятся в массивах? Как

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
	пишутся знаки неравенств? Можно ли проверить сразу весь массив X: ($X > 0$), или надо проверять каждый элемент массива отдельно?

Описание шкалы оценивания:

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: Оценка "отлично" выставляется, если задание выполнено в полном объеме или с одной-двумя незначительными описками

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: Оценка "хорошо" выставляется, если большинство вопросов раскрыто. выбрано верное направление для решения задачи, но есть две-три незначительные ошибки

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется, если задание преимущественно выполнено, но есть очень много мелких ошибок, или срок сдачи работы просрочен на месяц и более

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: Оценка "неудовлетворительно" выставляется, если задание выполнено неверно или преимущественно не выполнено, содержит существенные ошибки и недоработки, например, студент не различает алгоритмы ввода простых переменных и массивов, не знает, что требуется описание переменных и как их описать правильно

КМ-4. КМ1-4 Обработка массивов (лаб.5,7-8, Паскаль)

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Программирование (код)

Вес контрольного мероприятия в БРС: 13

Процедура проведения контрольного мероприятия: Рассчитать оценку за выполнение и защиту Лабораторных работ 5,7,8, направленных на изучение различных алгоритмов обработки массивов и их комбинаций. Для каждой лабораторной студент 5-7 минут демонстрирует на лабораторном занятии свой код на заранее разработанных своих тестах и отвечает на контрольные вопросы. Если не успевает, то может сдать спецификацию с программой на проверку в бумажном или электронном виде, но при наличии существенных ошибок они обсуждаются на лабораторном занятии за компьютером. Если сдана лабораторная работа 11 (1 семестр), то лабораторная работа 8 (1 семестр) может быть зачтена преподавателем с той же оценкой, поскольку Лабораторная работа 11 включает в себя выполнение лабораторной 8.

Краткое содержание задания:

Выполнить последовательно за 3-4 недели и защитить Лабораторные работы 5,7,8:

Лабораторная работа 5 (1 семестр). Экстремум

Разработать нисходящим способом алгоритм, выделив подзадачу поиска экстремума, и написать программу на Паскале, создав консольное приложение для решения задачи из нижеприведенного перечня задач. При этом:

1) Ввод исходных данных выполнить из нестандартного текстового файла.

- 2) Вывод исходных данных и результатов также выполнить в нестандартный текстовый файл.
- 3) Имена нестандартных текстовых файлов передавайте в программу через ее параметры.
- 4) Для взаимодействия с файлами здесь и далее используйте собственные файловые переменные, а не стандартные input и output.
- 5) Аномальные ситуации можно не рассматривать.
- 6) Создайте функциональные тесты для проверки работы программы. Для каждого из тестов здесь и далее создайте отдельные текстовые файлы с исходными данными.
- 7) Создайте отчет-спецификацию по примеру спецификации к лабораторным работам 2-4.

Примеры заданий по вариантам:

1. Даны одномерные массивы A и C из n элементов каждый. Определить наибольший элемент (значение и номер) в массиве A и наименьший элемент в массиве C .
2. Дан одномерный массив A из n элементов. Определить наибольшую по абсолютной величине разность между A_i и A_{i+1} .
3. Даны одномерные массивы A и C из n элементов каждый. Определить номер наибольшего из отношений A_i/C_i .
4. Дан одномерный массив A из n элементов. Определить номер наименьшей среди сумм $\sum_{i=1}^m A_i$, где $m = 1, 2, \dots, n$.
5. Даны одномерные массивы A и C из n элементов каждый. Определить произведение наибольшего элемента в массиве A и наименьшего элемента в массиве C .
6. Дан одномерный массив A из n элементов. Определить разность наибольшего и наименьшего элементов в массиве A .
7. Даны одномерные массивы A и C из n элементов каждый. Определить номер наименьшей по абсолютной величине разности $A_i - C_i$.
8. Дан одномерный массив A из n элементов. Определить наименьшую из разностей $|A_i - |A_{i+1}|$.

Лабораторная работа 7 (1 семестр). Сортировка двумя методами

Разработать алгоритм и написать две программы на Паскале, создав консольные приложения для решения задачи упорядочения массива двумя методами: методом простого выбора и обменным методом «пузырька». При этом:

- 1) Ввод исходных данных выполнить из нестандартного текстового файла.
- 2) Вывод исходных данных и результатов также выполнить в нестандартный текстовый файл.
- 3) Имена нестандартных текстовых файлов передавайте в программу через ее параметры.
- 4) Аномальные ситуации можно не рассматривать.
- 5) Создайте функциональные тесты для проверки работы программы, в том числе с исходным массивом, упорядоченным в порядке, обратном требуемому, и массивом с элементами уже в нужном порядке, и с одинаковыми элементами, с массивами минимальной и максимальной длины.
- 6) Для решения задачи можно написать как две программы, так и одну с возможностью выбора метода.
- 7) Поскольку по конечному результату виден только порядок сортировки (возрастание/убывание), для проверки работы метода сортировки выводите состояние массива после каждого прохода по массиву, чтобы можно было проследить направление перемещения экстремумов и «всплытие пузырьков». Один проход по массиву – одна строка, очередной экстремум на своем месте.

Варианты заданий в таблице:

Порядок сортировки и ее направления указан в таблице в заголовке столбца, в котором находится номер варианта:

Порядок Что ищем и переставляем	Возрастание (Неубывание)						Убывание (Невозрастание)					
	Мин в начало			Мах в конец			Мах в начало			Мин в конец		
Тип данных	цел	вещ	сим**	цел	вещ	сим**	цел	вещ	сим**	цел	вещ	сим**
№ варианта*	1	2*	3	4*	5	6	7*	8*	9	10*	11	12
	13*	14	15	16	17*	18	19*	20*	21	22	23*	24
	25	26*	27	28*	29	30	31	32	33	34*	35	36

*В вариантах, помеченных звездочкой, сортировать численные значения по значению *абсолютной* величины. Например, массив (0, -1, 2, -3, 10, -11) упорядочен по возрастанию абсолютной величины.

**Символьный тип имеет особенности ввода: разделители не нужны, они (разделители) – тоже символы! Желание усложнить себе задачу при сортировке символов могут сортировать не любые символы (кроме управляющих, конечно) по кодам символов (как это происходит по умолчанию при сравнении двух символов), а сделать, например, сортировку символов кириллицы по алфавиту, включая букву ё, или символов кириллицы/латиницы по алфавиту независимо от регистра: 'AaBCcDDeZz', 'AaBBГДддЯя'.

Лабораторная работа 8 (1 семестр). Кратные циклы и вложенные алгоритмы

Разработать алгоритм и написать программу на Паскале, создав консольное приложение для решения задачи своего варианта из нижеприведенного перечня. При этом:

- 1) Ввод исходных данных выполнить из нестандартного текстового файла.
- 2) Вывод исходных данных и результатов также выполнить в нестандартный текстовый файл.
- 3) Имена нестандартных текстовых файлов передавайте в программу через ее параметры.
- 4) Аномальные ситуации можно не рассматривать.
- 5) Все задачи в первом семестре выполнять только с использованием статических (не динамических открытых) массивов, и без goto, break, halt. Можно использовать exit или halt, но не из тела цикла.
- 6) Создайте функциональные тесты для проверки работы программы.

Примеры вариантов задач:

1. Дана матрица D , состоящая из n строк и n столбцов. Изменить все строки матрицы, в которых отрицателен элемент главной диагонали: к каждому элементу i -й строки прибавляется элемент T_i из заданного массива T_1, T_2, \dots, T_n . Подсчитать число измененных строк матрицы.
2. Дана матрица B , состоящая из n строк и n столбцов. Получить массив D_1, D_2, \dots, D_n по правилу: $D_i = 0$, если все элементы i -го столбца матрицы равны 0, иначе $D_i = 1$. Найти также сумму всех элементов матрицы.
3. Дана матрица D , состоящая из n строк и n столбцов. Найти среднее арифметическое отрицательных элементов матрицы, лежащих ниже главной диагонали, и среднее арифметическое всех элементов главной диагонали.
4. Дана матрица B , состоящая из n строк и n столбцов. Изменить матрицу, заменив каждый отрицательный элемент, лежащий выше главной диагонали, его абсолютной величиной. Найти также сумму элементов главной диагонали.
5. Дана матрица A , состоящая из n строк и n столбцов и задан массив X_1, X_2, \dots, X_n . Получить массив D_1, D_2, \dots, D_n по правилу: $D_i = \max(X_i, \sum_{j=1}^i A_{ij})$ и подсчитать в каком числе случаев сумма строки матрицы A оказывалась больше значения соответствующего элемента массива X .
6. Дана матрица B , состоящая из n строк и n столбцов. Найти среднее арифметическое элементов матрицы и сумму элементов тех строк матрицы, в которых отрицателен элемент главной диагонали.

Контрольные вопросы/задания:

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
Знать: алгоритмы простых внутренних сортировок	<ol style="list-style-type: none"> 1. Сколько раз достаточно повторять выполнение тела внешнего цикла при сортировке массива длиной N для гарантированного упорядочения этого массива? (1-7) 2. Есть ли для метода «пузырька» способ проверить, что массив уже упорядочен, хотя выполнено меньше необходимого числа проходов по массиву? (1-7) 3. При каждом следующем проходе по массиву количество элементов, которые надо сравнить остается таким же, уменьшается или увеличивается? (Для каждого из методов ответьте) (1-7) 4. Какие знаете способы для обмена значениями двух переменных? (1-7) 5. Какие есть способы обмена без участия третьей переменной? Какие у них ограничения? (1-7) 6. Как можно проверить при сортировке по возрастанию (неубыванию) в каком направлении вы перемещаете элементы: минимум в начало или максимум в конец? (1-7) 7. Как можно проверить при сортировке по убыванию (возрастанию) в каком направлении вы перемещаете элементы: максимум в начало или минимум в конец? (1-7) 8. В чем особенность ввода символьных значений в отличие от ввода целочисленных значений? (1-7)
Уметь: разрабатывать алгоритмы с вложенными	1. Как выполнить ввод из текстового файла значений элементов двумерного массива, состоящего из N строк и

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
друг в друга простыми алгоритмами и циклами	<p>М столбцов целых чисел, если в этом файле все числа расположены «в виде матрицы»: элементы строк разделены пробелами или табуляциями, а элементы разных строк разделены концом строки? (1-8)</p> <p>2.Как выполнить вывод в текстовый файл значений элементов двухмерного массива, состоящего из N строк и M столбцов вещественных чисел с плавающей точкой, если в этом файле все числа должны быть расположены «в виде матрицы»: элементы строк 4 разделены пробелами или табуляциями, а элементы разных строк разделены концом строки? Как указать ширину столбцов при выводе? (1-8)</p> <p>3.Есть ли в вашей задаче альтернативные отрицательные решения?(1-8)</p> <p>4.Поясните указанную преподавателем строку своей программы, спецификации. (1-7-8)</p>
Уметь: разрабатывать простые алгоритмы на поиск экстремальных значений и их позиции	<p>1.Если несколько элементов в массиве имеют экстремально большое значение, какой из них считать искомым максимальным элементом? Где в спецификации это надо уточнить? (1-5)</p> <p>2.Если нет ни одного элемента, подходящего под условие поиска среди них экстремума, является ли это ошибкой, аномалией, отрицательным решением? (1-5)</p> <p>3.Какие достоинства у возможности передавать имена файлов через параметры программы? Какие недостатки? (1-5)</p> <p>4.Как можно узнать сколько параметров получила программа помимо нулевого, который есть всегда? Что передается программе в нулевом параметре? (1-5)</p> <p>5.Как узнать значение каждого из полученных параметров программы? (1-5)</p> <p>6.Как передать программе параметры при отладке программы в среде разработке? (1-5)</p> <p>7.Как передать программе параметры при запуске exe-файла в среде MS Windows? (1-5)</p> <p>8.Поясните указанную преподавателем строку своей программы или спецификации к лаб.5. (1-5)</p>

Описание шкалы оценивания:

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: Если минимальная оценка по Лаб.5,7,8 это "5". Оценка по Лаб.5,7,8: оценка "отлично" выставляется, если задание выполнено в полном объеме или выполнено преимущественно верно и не содержит лишних действий, и студент легко отвечает на вопросы по программе и спецификации. Недочеты исправлены в указанные сроки с первой попытки

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: Если минимальная из трёх оценок по Лаб.5,7,8 это "4". Оценка "хорошо" по Лаб.5,7,8 выставляется, если задание выполнено в

полном объеме и выполнено преимущественно верно, и может ответить преимущественно верно на заданные контрольные вопросы, но не все недочеты устранены на момент постановки оценки. И для устранения недочетов потребовалось не более двух попыток

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: Если минимальная из трёх оценок по Лаб.5,7,8 это "3". Оценка "удовлетворительно" по Лаб.5,7,8 выставляется, если задание преимущественно выполнено, но есть несколько замечаний и для устранения недочетов потребовалось более двух попыток и не на все вопросы студент может ответить, либо время сдачи работы просрочено на три недели и более по неуважительной причине

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: Если минимальная из трёх оценок по Лаб.5,7,8 это "2".. Оценка "неудовлетворительно" по Лаб.5,7,8 выставляется, если задание выполнено неверно или преимущественно не выполнено, или студент не понимает собственный сдаваемый код и не может ответить на большинство контрольных вопросов. Если студент не показал ни разу спецификацию или программу, не приступил к их выполнению, то "0".

КМ-5. КМ1-5 Лабораторная работа 9 (1 семестр). Исключения и итерационные циклы

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Программирование (код)

Вес контрольного мероприятия в БРС: 9

Процедура проведения контрольного мероприятия: Студент 5-10 минут демонстрирует на лабораторном занятии блок-схему и свой код на заранее разработанных своих тестах и отвечает на контрольные вопросы. Если не успевает, то может сдать спецификацию с программой на проверку в бумажном и электронном виде, но при наличии существенных ошибок они обсуждаются на лабораторном занятии за компьютером. Работа по требованию преподавателя может быть заранее сдана на проверку для сокращения времени обсуждения на занятии.

Краткое содержание задания:

Разработать алгоритм и написать программу на Паскале, создав консольное приложение для решения задачи своего варианта из нижеприведенного перечня. При этом:

- 1) Ввод исходных данных выполнить из нестандартного текстового файла.
- 2) Вывод исходных данных и результатов также выполнить в нестандартный текстовый файл.
- 3) Имена нестандартных текстовых файлов передавайте в программу через ее параметры.
- 4) Предусмотрите проверку исключительных ситуаций, связанных с открытием и чтением из внешнего файла. Аномальные ситуации, связанные с некорректными диапазонами исходных данных можно не рассматривать.
- 5) Для решения задачи, не считая подзадачи ввода и вывода, не использовать параметрический цикл ДЛЯ, а только итерационные циклы ПОКА или ДО. Не использовать досрочный выход из цикла с помощью goto, break, exit, halt и т.д. и т.п. во всех лабораторных работах семестра.
- 6) Создайте функциональные тесты для проверки работы программы, начертите блок-схему(ы) для подзадачи поиска строки/столбца в матрице.

Примеры заданий по вариантам:

1. Дана прямоугольная матрица из символов. Найти номер последней строки, все элементы в которой одинаковы.
2. Найти в прямоугольной матрице первую строку, все элементы которой положительны.
3. Найти в прямоугольной матрице последнюю строку, все элементы которой упорядочены по убыванию.
4. Найти в целочисленной прямоугольной матрице первую строку, все элементы которой кратны 5.
5. Найти в квадратной матрице последнюю строку, все элементы которой упорядочены по возрастанию.
6. Найти в заданной квадратной матрице первую строку, все элементы которой отрицательны.
7. Дана прямоугольная матрица из символов. Проверить, что во всех строках есть символ «Я» или «я». Если не во всех, то найти номер последней строки без этих символов.
8. Проверить, все ли строки квадратной матрицы упорядочены по убыванию, если не все, сообщить номер последней неупорядоченной.
9. Проверить, все ли строки прямоугольной матрицы упорядочены по возрастанию абсолютных величин элементов, если не все, сообщить номер первой неупорядоченной.
10. Проверить, все ли строки квадратной матрицы содержат хотя бы один положительный элемент, если не все, сообщить номер последней, где нет положительных элементов.

Контрольные вопросы/задания:

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
<p>Знать: управляющие структуры языка программирования, в том числе итерационные циклы с несколькими возможными условиями выхода, для написания более эффективного программного кода</p>	<ol style="list-style-type: none"> 1. Как свести алгоритм с досрочным выходом из цикла for к циклам while и repeat (метод флажка)? (1-9) 2. Какие есть итерационные циклы в Паскале? (1-9) 3. В каком из итерационных циклов указывается условие выхода из цикла, а в каком – условие продолжения выполнения тела цикла? (1-9) 4. Чем отличается цикл с предусловием от цикла с постусловием? Начертите блок-схемы таких циклов? (1-9) 5. Чем отличаются итерационные циклы от параметрического цикла for? (1-9)
<p>Уметь: применять технологию обработки исключительных ситуаций для адекватного ответа на ошибки времени выполнения программы и использовать итерационные циклы со сложным условием выхода</p>	<ol style="list-style-type: none"> 1. При каких действиях с файлами в программе возможно появление исключительных ситуаций? Зависят ли возникновение или обработка этих ситуаций от программиста, написавшего эту программу? (1-9) 2. Чем отличаются блоки Try..Except и Try..Finally? В каких случаях какой блок используется в программе? (1-9) 3. Как проверить, не произошло ли ошибки при открытии/создании файла? Как вывести собственное сообщение об этой ошибке и избежать аварийного завершения программы? (1-9) 4. Как проверить, удалось ли считать значения исходных данных из нестандартного текстового файла? Как вывести собственное сообщение об этой ошибке, если не удалось, и не забыть закрыть файл? (1-9) 5. В каких ещё ситуациях помимо работы с файлами можно использовать обработку исключительных ситуаций? Приведите пример. (1-9) 6. Поясните указанную преподавателем

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
	строку своей программы, спецификации. (1-9)

Описание шкалы оценивания:

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: Оценка "отлично" выставляется, если задание выполнено в полном объеме или выполнено преимущественно верно и не содержит лишних действий, и студент легко отвечает на вопросы по программе. Недочеты исправлены быстро и с первой попытки

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: Оценка "хорошо" выставляется, если задание выполнено в полном объеме и выполнено преимущественно верно, и может ответить преимущественно верно на заданные контрольные вопросы, но не все недочеты устранены на момент простановки оценки. И для устранения недочетов потребовалось не более двух попыток

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется, если задание преимущественно выполнено, но есть несколько замечаний и для устранения недочетов потребовалось более двух попыток, либо не на все вопросы студент может ответить, либо время сдачи работы просрочено на пять недель и более по неуважительной причине

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: Оценка "неудовлетворительно" выставляется, если задание выполнено неверно или преимущественно не выполнено, или студент не понимает собственный сдаваемый код и не может ответить на большинство контрольных вопросов

КМ-6. КМ1-6 Процедуры и функции, их параметры (лаб.6,10,11, Паскаль)

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Домашнее задание

Вес контрольного мероприятия в БРС: 18

Процедура проведения контрольного мероприятия: Рассчитать оценку за выполнение и защиту Лабораторных работ 6,10,11, направленных на изучение процедур и функций. Для каждой лабораторной студент 5-7 минут демонстрирует на лабораторном занятии свой код на заранее разработанных своих тестах и отвечает на контрольные вопросы. Если не успевает, то может сдать спецификацию с программой на проверку в бумажном или электронном виде, но при наличии существенных ошибок они обсуждаются на лабораторном занятии за компьютером. Если сдана лабораторная работа 10 (1 семестр), то лабораторная работа 6 (1 семестр) может быть зачтена преподавателем с той же оценкой, поскольку Лабораторная работа 10 включает в себя выполнение лабораторной 6.

Краткое содержание задания:

Выполнить последовательно за 2-3 недели и защитить Лабораторные работы 6,10,11:

Лабораторная работа 6. Экстремум с двумя условиями. Выделение подзадач.

Разработать нисходящим способом алгоритм, выделив несколько подзадач (в том числе поиск элемента по условию 1 и по условию 2, поиск экстремума), и написать программу на Паскале, создав консольное приложение для решения задачи поиска минимального

или максимального значения элемента и/или его номера среди элементов одномерного целочисленного массива, удовлетворяющих двум условиям (см. перечень задач). При этом:

- 1) Ввод исходных данных выполнить из нестандартного текстового файла.
- 2) Вывод исходных данных и результатов также выполнить в нестандартный текстовый файл.
- 3) Имена нестандартных текстовых файлов передавайте в программу через ее параметры.
- 4) Аномальные ситуации можно не рассматривать.
- 5) Создайте функциональные тесты и проверьте работу программы в каждой из пяти-шести возможных ситуаций. Не забывайте для каждого из тестов здесь и далее создавать отдельные текстовые файлы с исходными данными.

Примеры задач по вариантам (6-10):

6. Задан целочисленный одномерный массив A из N элементов. Найти номер последнего максимального значения среди отрицательных элементов, расположенных после первого элемента, равного T . Если нет элементов со значением T , искать с начала массива. (Подсказка: «Сначала найти первый, равный T , и последний отрицательный, а затем уже искать максимум либо вывести сообщение о причинах невозможности поиска.»)
7. Задан целочисленный одномерный массив A из N элементов. Найти номер последнего максимального элемента среди элементов, меньших T_1 , и расположенных до первого элемента, большего T_2 . (Подсказка: «Сначала найти первый, меньший T_1 , и первый, больший T_2 , а затем уже искать максимум либо вывести сообщение о причинах невозможности поиска.»)
8. Задан целочисленный одномерный массив A из N элементов. Найти значение максимального отрицательного элемента среди четных (по значению) элементов, расположенных до последнего нечетного (по значению) элемента. (Подсказка: «Сначала найти первый отрицательный+четный и последний нечетный, а затем уже искать максимум либо вывести сообщение о причинах невозможности поиска.»)
9. Задан целочисленный одномерный массив A из N элементов. Найти номер первого минимального элемента среди элементов, больших T_1 , и расположенных после первого элемента, равного T_2 . Если нет элементов со значением T_2 , то искать с начала массива. (Подсказка: «Сначала найти первый, равный T_2 , и последний, больший T_1 , а затем уже искать минимум либо вывести сообщение о причинах невозможности его поиска.»)
10. Задан целочисленный одномерный массив A из N элементов. Найти номер последнего максимального элемента среди четных (по значению) элементов и расположенных до первого нечетного элемента, значение которого к тому же лежит в диапазоне $[c,d]$. Если нет нечетных элементов со значением из диапазона $[c,d]$, искать до конца массива. (Подсказка: «Сначала найти первый четный (по значению, а не по индексу) и первый нечетный со значением из диапазона $[c,d]$, а затем уже искать максимум либо вывести сообщение о причинах невозможности поиска.»)

Лабораторная работа 10 (1 семестр). Процедуры с параметрами в лаб.6

Разработать алгоритм и написать программу на Паскале, создав консольное приложение для решения задачи из Лабораторной работы №6, выделив подзадачи проверки условий и поиска экстремума и оформив их в виде процедур. При этом:

- 1) Ввод исходных данных выполнить из нестандартного текстового файла.
- 2) Вывод исходных данных и результатов также выполнить в нестандартный текстовый файл.
- 3) Имена нестандартных текстовых файлов передавайте в программу через ее параметры.
- 4) Аномальные ситуации можно не рассматривать.
- 5) Внутри процедур (функций) не использовать глобальных переменных напрямую: вся связь с вызывающей программой только через интерфейс процедуры – её параметры. Для удобства самопроверки этого факта размещайте описания процедур (функций) до раздела описания глобальных переменных var.
- 6) Не смешивайте внутри процедур (функций) решение задачи с вводом или выводом: либо только ввод и/или вывод, либо только решение (поиск, вычисления и т.д.) без ввода дополнительных значений и без вывода результата.
- 7) Для всесторонней проверки работы программы не забудьте создать тесты для всех альтернативных решений.

Условия задач по вариантам см выше в лаб.6.

Лабораторная работа 11 (1 семестр). Процедуры и функции, Повторные вызовы с другими параметрами

Написать две программы обработки трех двумерных массивов для решения задачи своего варианта из нижеприведенного перечня. В первой версии программы использовать только процедуры общего вида. Во второй версии программы как минимум

решение задания из лабораторной работы 8 должно быть функцией (вместо процедуры общего вида). При этом:

- 1) Ввод исходных данных выполнить из нестандартного текстового файла. Одного или нескольких.
- 2) Вывод исходных данных и результатов также выполнить в нестандартный текстовый файл.
- 3) Имена нестандартных текстовых файлов передавайте в программу через ее параметры.
- 4) Аномальные ситуации можно не рассматривать.
- 5) Для решения задачи выделить не менее двух подзадач, не считая подзадачи ввода, вывода и проверки аномалий. Все подзадачи оформить в виде процедур общего вида (в первой версии программы) и/или функций (во второй версии программы).
- 6) Внутри процедур (функций) не использовать глобальных переменных напрямую: вся связь с вызывающей программой только через интерфейс процедуры – её параметры. Для удобства самопроверки этого факта размещайте описания процедур (функций) до раздела описания глобальных переменных var.
- 7) Не совмещайте внутри процедур (функций) решение задачи с вводом или выводом: либо только ввод и/или вывод, либо только решение (поиск, вычисления и т.д.) без ввода дополнительных значений и без вывода результата.
- 8) Для всесторонней проверки работы программы не забудьте создать тесты для всех альтернативных решений с одним или несколькими экстремумами.

Примеры задач по вариантам для разделения на подзадачи:

9. Даны три двумерных массива: $A (na \times na)$, $B (nb \times nb)$ и $C (nc \times nc)$. Для каждого массива A,B,C решить задачу из ЛР №8 и затем определить, в каком(их) из них наибольшее количество упорядоченных по возрастанию строк. Если ни в одном из массивов нет упорядоченных строк, вывести сообщение об этом. Оформить в виде процедур ввод и вывод матрицы, всё решение задачи из ЛР №8 для отдельной матрицы и, по желанию, поиск максимума. Размеры и значения элементов массивов вводить из одного общего или трех разных текстовых файлов, результат и исходные данные также вывести в пользовательский текстовый файл.
10. Даны три двумерных массива: $A (na \times na)$, $B (nb \times nb)$ и $C (nc \times nc)$. Для каждого массива A,B,C решить задачу из ЛР №8 и затем определить, в каком(их) из них наибольшее произведение элементов, матрицы. Оформить в виде процедур ввод и вывод матрицы, всё решение задачи из ЛР №8 для отдельной матрицы, вывод созданного массива и, по желанию, поиск максимума. Размеры и значения элементов массивов вводить из одного общего или трех разных текстовых файлов, результат и исходные данные также вывести в пользовательский текстовый файл.
11. Даны три двумерных массива: $A (na \times na)$, $B (nb \times nb)$ и $C (nc \times nc)$. Для каждого массива A,B,C решить задачу из ЛР №8 и затем определить, в каком(их) из массивов найденное среднее арифметическое наименьшее. Если ни в одном из массивов нет возможности найти среднее арифметическое среди указанных в задаче из ЛР №8 элементов, вывести сообщение об этом. Оформить в виде процедур ввод и вывод матрицы, всё решение задачи из ЛР №8 для отдельной матрицы и, по желанию, поиск минимума. Размеры и значения элементов массивов вводить из одного общего или трех разных текстовых файлов, результат и исходные данные также вывести в пользовательский текстовый файл.
12. Даны три двумерных массива: $A (na \times na)$, $B (nb \times nb)$ и $C (nc \times nc)$. Для каждого массива A,B,C решить задачу из ЛР №8 и затем определить, в каком(их) из массивов A,B,C наибольшее количество измененных элементов. Если окажется, что ни в одном из массивов нет измененных элементов, вывести сообщение об этом. Оформить в виде процедур ввод и вывод матрицы, всё решение задачи из ЛР №8 для отдельной матрицы и, по желанию, поиск максимума. Размеры и значения элементов массивов вводить из одного общего или трех разных текстовых файлов, результат и исходные данные также вывести в пользовательский текстовый файл.
13. Даны три двумерных массива: $A (na \times na)$, $B (nb \times nb)$ и $C (nc \times nc)$. Для каждого массива A,B,C решить задачу из ЛР №8 и затем определить, в каком(их) из них наибольшее общее число инверсий. Если ни в одном из массивов нет инверсий, вывести сообщение об этом. Оформить в виде процедур ввод и вывод матрицы, всё решение задачи из ЛР №8 для отдельной матрицы, вывод созданного массива и, по желанию, поиск максимума. Размеры и значения элементов массивов вводить из одного общего или трех разных текстовых файлов, результат и исходные данные также вывести в пользовательский текстовый файл.

Контрольные вопросы/задания:

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
<p>Уметь: выделять подзадачи и описывать процедуры и их параметры, вызывать процедуры на языке Паскаль/Delphi</p>	<p>1.Какие виды параметров имеются в языке Паскале? В каких видах параметров используется передача по значению, а в каких – по ссылке? Например, четыре разных параметра в заголовке: procedure quest1(const p1:real; p2:real; var p3:real; out p4: real); (1-10)</p> <p>2.Чем будут отличаться параметры, перед одним из которых указано слово const, а перед вторым – ничего? Например, в заголовке: procedure question2(const p1: real; p2: real); (1-10)</p>

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
	<p>3. Чем будут отличаться параметры, перед одним из которых указано слово <code>var</code>, а перед вторым – ничего? Например, в заголовке: <code>procedure question3(p2: real; var p3: real);</code> (1-10)</p> <p>4. Зачем нужны параметры-константы? Как их описать? (1-10)</p> <p>5. Как можно использовать параметры-значения? Как их описать? (1-10)</p> <p>6. Как можно использовать параметры-переменные? Как их описать? (1-10)</p> <p>7. Зачем в процедуре создается копия исходного значения при передаче параметра по значению? (1-10)</p> <p>8. Сколько в вашей задаче альтернативных ответов и какие? (1-10)</p>
<p>Уметь: выделять подзадачи и разрабатывать алгоритмы на проверку наличия или отсутствия подходящих по условию значений</p>	<p>1. Какие альтернативные решения есть в вашей задаче? Сколько положительных решений? Сколько отрицательных? (1-6)</p> <p>2. Из решения каких подзадач складывается решение всей вашей задачи? (1-6)</p> <p>3. Чем может помочь вывод промежуточных результатов? Вывод исходных данных? (1-6)</p> <p>4. Чем могут помочь при отладке разные ответы для каждого из альтернативных решений? (1-6)</p> <p>5. Для всех ли альтернативных решений нужно создать тестовые наборы? Почему? (1-6)</p> <p>6. Поясните указанную преподавателем строку своей программы. (1-6)</p>
<p>Уметь: повторно использовать процедуры с параметрами, отличать и описывать процедуры общего вида и функции на языке Паскаль/Delphi</p>	<p>1. Возвращает ли процедура общего вида результат в точку вызова? Или только управление? (1-11)</p> <p>2. Возвращает ли функция результат в точку вызова? Или только управление? (1-11)</p> <p>3. Как указать значение, которое функция вернула в точку вызова? (1-11)</p> <p>4. Можно ли передавать в функцию параметры по ссылке (<code>var/out</code>) или только по значению? (1-11)</p> <p>5. Из любой ли функции можно сделать процедуру общего вида? Как это сделать? Изменится ли вызов? (1-11)</p> <p>6. Из любой ли процедуры общего вида можно сделать функцию? Как это сделать? Изменится ли вызов? (1-11)</p> <p>7. Поясните указанные преподавателем строки своей программы. (1-11)</p>

Описание шкалы оценивания:

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: Если минимальная оценка по Лаб.6,10,11 это "5". Оценка по Лаб.6,10,11: оценка "отлично" выставляется, если задание выполнено в полном объеме или выполнено преимущественно верно и не содержит лишних действий, и

студент легко отвечает на вопросы по программе и спецификации. Недочеты исправлены в указанные сроки с первой попытки

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: Если минимальная из трёх оценок по Лаб.6,10,11 это "4". Оценка "хорошо" по Лаб.6,10,11 выставляется, если задание выполнено в полном объеме и выполнено преимущественно верно, и может ответить преимущественно верно на заданные контрольные вопросы, но не все недочеты устранены на момент простановки оценки. И для устранения недочетов потребовалось не более двух попыток

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: Если минимальная из трёх оценок по Лаб.6,10,11 это "3". Оценка "удовлетворительно" по Лаб.6,10,11 выставляется, если задание преимущественно выполнено, но есть несколько замечаний и для устранения недочетов потребовалось более двух попыток и не на все вопросы студент может ответить, либо время сдачи работы просрочено на три недели и более по неуважительной причине

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: Если минимальная из трёх оценок по Лаб.6,10,11 это "2".. Оценка "неудовлетворительно" по Лаб.6,10,11 выставляется, если задание выполнено неверно или преимущественно не выполнено, или студент не понимает собственный сдаваемый код и не может ответить на большинство контрольных вопросов.

Если студент не показал ни разу результат лаб.работы (10-11), не приступил к их выполнению, то "0".

КМ-7. КМ1-7 Контрольная работа 2 (1 семестр). Процедуры и параметры.

Формы реализации: Письменная работа

Тип контрольного мероприятия: Контрольная работа

Вес контрольного мероприятия в БРС: 20

Процедура проведения контрольного мероприятия: Озвучивается задания и выдаются условия задач по вариантам. Через 1 час и 30 минут (2 ак.часа) студенты должны сдать написанный темной яркой авторучкой на листах белой бумаги программный код на языке Delphi с процедурами и блок-схему верхнего уровня А0.

Краткое содержание задания:

1.Разработать нисходящим способом алгоритм решения задачи, начертив одну блок-схему верхнего уровня с выделенными подзадачами

2. Написать полностью программный код с 2-3 процедурами общего вида или функциями для решения задачи и 1-2 процедурами для ввода-вывода. При этом:

2.1) Все процедуры должны иметь параметры и не должны использовать глобальные переменные.

2.2) Ввод и вывод не совмещать в одной процедуре с обработкой (поиском, обработкой, проверкой и т.д.)

Примеры задач для нисходящей разработки и решения:

1. Если все элементы K -го столбца матрицы A ($n \times n$, $1 < n \leq 10$) отрицательны (K запросить, $1 \leq K \leq n$), найти значения максимального элемента ниже главной диагонали матрицы и значение минимального элемента выше главной диагонали. В противном случае вывести сообщение «Не все отрицательны».
2. Если суммы всех строк матрицы B ($n \times m$) положительны, найти индексы минимального положительного элемента матрицы, иначе вывести сообщение «Не все суммы положительны».
3. Если в матрице C ($n \times m$) первая и последняя строки совпадают (поэлементно), то найти произведение и сумму значений элементов всех остальных строк матрицы, иначе вывести сообщение «Не совпадают».
4. Если все суммы значений элементов строк матрицы D ($n \times m$) больше произведений элементов этих строк, вывести сообщение «все суммы больше», иначе вывести номер первой строки, для которой условие не выполняется и среднее арифметическое ее элементов.

Контрольные вопросы/задания:

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
<p>Уметь: разрабатывать алгоритм и записывать программу на языке процедурного программирования (Паскаль/Delphi)</p>	<ol style="list-style-type: none"> 1. Сколько в заданной задаче подзадач? Какие у них условия? Какие данные нужны на входе и выходе? Какой порядок их выполнения? 2. Какие параметры нужны вашим подзадачам? 3. Какие параметры можно передать процедурам по значению, а какие необходимо передать по ссылке? 4. Следует ли описать пользовательские типы? Определить константы? 5. Как осуществляются вызовы процедур? 6. В каком порядке надо передать параметры в процедуры? 7. Должны ли соответствовать типы и порядок формальных и фактических параметров? В каких случаях?

Описание шкалы оценивания:

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: Оценка "отлично" выставляется, если задание выполнено в полном объеме (блок-схема, процедуры, параметры, программа, алгоритмы) или выполнено преимущественно верно с парой небольших описок или неточностей

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: Оценка "хорошо" выставляется, если задание выполнено в полном объеме (блок-схема, процедуры, параметры, программа, алгоритмы) или выполнено преимущественно верно, но есть две-три ошибки или замечания к работе о несоблюдении требований, или задание выполнено в полном объеме, но отсутствует блок-схема

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется, если студент смог верно выделить подзадачи, но блок-схема, описания основной программы, процедур и параметров присутствуют, но содержат много ошибок, при этом общий смысл и ход решения задачи верен

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: Оценка "неудовлетворительно" выставляется, если задание выполнено неверно или преимущественно не выполнено, или вовсе отсутствует большая часть работы

КМ-8. КМ1-8 Итоговое индивидуальное задание (1 семестр). Выполнение и защита типового расчёта.

Формы реализации: Защита задания

Тип контрольного мероприятия: Расчетно-графическая работа

Вес контрольного мероприятия в БРС: 15

Процедура проведения контрольного мероприятия: Студент показывает свой отчет-спецификацию и в течении 10-20 минут отвечает на контрольные вопросы о спецификации, предлагает пути устранения найденных в спецификации и в программе ошибок. При необходимости возможна демонстрация студентом правильности создания, работы и решения своей многомодульной программы на компьютере, ее отладка. Работа по требованию преподавателя может быть заранее сдана на проверку для сокращения времени обсуждения на занятии.

Краткое содержание задания:

Решить задачу своего варианта из нижеприведенного перечня, создав **многомодульное** консольное приложение для MS Windows. При этом:

- 1) Ввод исходных данных выполнить из нестандартного текстового файла.
- 2) Вывод исходных данных и результатов также выполнить в нестандартный текстовый файл.
- 3) Использовать только статические массивы, не динамические. В условиях задач N, M, K – переменные, а не константы. Максимальные размеры массивов выбрать самим, но не менее 5 строк и 5 столбцов в двумерных и не менее 6-10 элементов в одномерных массивах.
- 4) Обработать аномальные ситуации, связанные с возможным выходом значений каждого из исходных данных за определенные вами в таблице данных диапазоны. По желанию можно также добавить обработку исключительных ситуаций, связанных с открытием файлов и чтением из файла.
- 5) Для решения задачи выделить не менее двух подзадач, не считая подзадачи ввода, вывода и проверки аномалий. Все подзадачи оформить в виде процедур общего вида или функций.
- 6) Внутри процедур (функций) не использовать глобальных переменных напрямую: вся связь с вызывающей программой только через интерфейс процедуры – её параметры.
- 7) Не совмещать внутри процедур (функций) решение задачи с вводом или выводом: либо только ввод и/или вывод, либо только решение (поиск, вычисления и т.д.) без ввода дополнительных значений и без вывода результата.
- 8) Все процедуры (функции) расположить не в коде головного модуля программы (program), а в отдельных модулях (unit), в одном или нескольких.
- 9) Создать отчет по данной работе в рукописном или печатном виде, написав с титульный лист и полную спецификацию из всех 12 пунктов, включая структурные тесты, для уровня абстракций A0, и для абстракций всех остальных уровней в укороченной форме (пункты 1-4 и 9-11(12)) или полностью. При использовании восходящего тестирования привести также коды драйверов для каждой из подзадач, и отдельные функциональные и структурные тесты.

Примеры условий задач по вариантам:

3. Даны три последовательности: X_1, X_2, \dots, X_k ; Y_1, Y_2, \dots, Y_k и M_1, M_2, \dots, M_k . Каждая тройка элементов (X_i, Y_i, M_i) представляет параметры одной из K материальных точек, лежащих в плоскости XOY : абсциссу X_i , ординату Y_i и массу M_i . Если абсциссы и ординаты всех точек положительны, найти (XC, YC) – координаты центра тяжести данной системы масс по формулам:

$$XC = \frac{\sum_{i=1}^k X_i M_i}{\sum_{i=1}^k M_i}; YC = \frac{\sum_{i=1}^k Y_i M_i}{\sum_{i=1}^k M_i}$$

4. Дана матрица P с двумя строками и K столбцами, каждым столбцом которой задана абсцисса и ордината одной из K точек плоскости. Если нет ни одной пары точек, расстояние между которыми меньше заданной величины R , заменить на ноль в матрице P все отрицательные абсциссы точек, увеличив ординаты этих точек на R .
5. Дана матрица A из N строк и M столбцов. Если среднее арифметическое матрицы A положительно, задать элементам C_1, C_2, \dots, C_i значения тех элементов матрицы A , которые больше этого среднего арифметического.
6. Дана матрица B из N ($N > 1$) строк и N столбцов. Если в матрице B элемент с максимальным значением (среди всех элементов этой матрицы) лежит выше главной диагонали, найти сумму элементов матрицы, лежащих ниже главной диагонали.
7. Дана матрица A из N строк и M столбцов. Найти среднее арифметическое элементов матрицы A , и, если матрица A не содержит ни одного отрицательного элемента, изменить элементы матрицы путем вычитания из них этого среднего арифметического.
8. Дана матрица B из M строк и M столбцов. Если все элементы главной диагонали матрицы B отрицательны, разделить все элементы матрицы на максимальный по абсолютной величине элемент матрицы.
9. Дана матрица A из N строк и N столбцов. Если разность максимального и минимального элемента матрицы A превышает заданную величину E , заменить в матрице A все отрицательные элементы нулями, а положительные единицами и подсчитать число выполненных замен.

Контрольные вопросы/задания:

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
<p>Знать: функциональные возможности программных средств по созданию многомодульных проектов и отладке программ на процедурном языке программирования, о возможности пошагового выполнения программ, о возможности передаче параметров программе для функционального и структурного тестирования (на основе черного и белого ящиков)</p>	<ol style="list-style-type: none"> 1. Как в среде программирования создать многомодульное приложение на языке Delphi? (12) 2. Как подключить один модуль к другому? (12) 3. Какие есть обязательные разделы у модуля на языке Delphi? Дополнительные? Какие использованы в работе? (12) 4. Чем отличаются разделы модуля Interface и Implementation на языке Delphi? (12) 5. Как отличаются по способу построения функциональные и структурные тесты? Что такое “белый” и “черный” ящики? (12) 6. Как в среде программирования пошагово выполнять программу? (12) 7. Как при пошаговом выполнении программы узнать значения переменных? (12) 8. Как узнать, какая ошибка и в какой строке возникла в программе? (12) 9. Как описание процедуры перенести в отдельный модуль (файл программы)? (12) 10. Поясните указанную преподавателем строку своей программы или отчета-спецификации. (12)

Описание шкалы оценивания:

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: Оценка "отлично" выставляется, если задание выполнено в полном объеме (многомодульное приложение, блок-схемы, процедуры, параметры, функциональные и структурные тесты) или выполнено преимущественно верно с парой небольших описок или неточностей, которые студент быстро и сразу может исправить. Студент легко и верно или с небольшими неточностями отвечает на все заданные контрольные вопросы

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: Оценка "хорошо" выставляется, если задание выполнено в полном объеме (многомодульное приложение, блок-схемы, процедуры, параметры, функциональные и структурные тесты) или выполнено преимущественно верно, но есть одна-две ошибки или замечания к работе о несоблюдении требований к ее выполнению. И для устранения недочетов потребовалось не более двух попыток. Студент верно или с небольшими неточностями отвечает на все заданные контрольные вопросы

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется, если задание преимущественно выполнено, но есть несколько замечаний и для устранения недочетов потребовалось более двух попыток, либо не на все вопросы студент может ответить и замечания устранить

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: Оценка "неудовлетворительно" выставляется, если задание выполнено неверно или преимущественно не выполнено, или студент не понимает собственный сдаваемый код и спецификацию к нему и не может ответить на большинство контрольных вопросов

2 семестр

КМ-1. КМ2-1 Закрепление (Лаб.1-2, Паскаль)

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Программирование (код)

Вес контрольного мероприятия в БРС: 12

Процедура проведения контрольного мероприятия: Рассчитать оценку за выполнение и защиту Лабораторных работ 1-2, направленных на закрепление большинства тем прошлого семестра: модули, процедуры и функции, параметры, массивы, записи, множества, перечисляемый тип, обменная сортировка, символы, и новых тем: строки, типизированный файл. Для каждой лабораторной студент 5-7 минут демонстрирует на лабораторном занятии свой код на заранее разработанных своих тестах и отвечает на контрольные вопросы. Если не успевает, то может сдать спецификацию с программой на проверку в бумажном или электронном виде, но при наличии существенных ошибок они обсуждаются на лабораторном занятии за компьютером.

Краткое содержание задания:

Выполнить последовательно за 3-4 недели и защитить Лабораторные работы 1-2:

Лабораторная 1. Строки и множества:

1 часть. (2 ак. часа). Создать в среде программирования Паскаля (Free Pascal Lazarus, PascalABC.NET или Borland Developer Studio 2006, Embarcadero Delphi Community Edition) приложение по указанному примеру

А) Ознакомится с примером и спецификацией приведенной ниже.

- Б) Создать консольное приложение, скопировать в него соответствующий код.
 В) Создать модуль (unit) для Delphi for Win32, скопировать в него код модуля, переименовать файл и связать его с программой, убрав старую связь.
 Г) Отладить, показать преподавателю.

2 часть. (2 ак. часа). Исправить приведенный пример, адаптировав его для решения своей задачи сортировки по варианту.

Общая задача. Упорядочить символы строки, расположив символы в указанном в нижеследующей таблице порядке (по вариантам) **двумя способами**: с созданием своей таблицы символов и без ее создания (со сложным условием сравнения и множествами). Провести **анализ аномалий**: строка не пустая и других символов (кроме указанных) нет.

При выводе результата в начало строки и в конец **добавить символ "** (двойная кавычка), чтобы указать начало и конец строки, что особенно актуально для строк из пробелов и других непечатаемых символов (s:= +<нужный символ>+s+<нужный символ или строка>).

Указание 1. Для проверки и сортировки использовать **процедуры**, расположенные в **отдельном модуле**

Указание 2. Для проверки принадлежности символа к одной из трёх групп **использовать множества**.

Указание 3. Изменить(упорядочить) заданную строку по указанному в каждом варианте правилу **«на месте» без создания дополнительной строки (или массива, файла и т.д.)**.

Замечания:

Символы упорядочены согласно **таблице кодов символов** и их порядок почти совпадает с алфавитным. Таблицы кодов для ОС *Windows* и *DOS* не совпадают, поэтому на экран символы кириллицы выводятся не корректно. В таблице кодов для *Windows* сначала идут цифры (48..57), символы латинского алфавита (ЗАГЛАВНЫЕ – 65..90, прописные – 97..122), затем символы кириллицы (ЗАГЛАВНЫЕ – 192..223 и прописные – 224..255, Ё – 168, ё – 184).

Символьные данные можно сравнивать обычным образом:

'0'<'1'<...<'F'<'G'<...<'f'<'g'<...<'Г'<'Д'<...<'я' – при этом сравниваются **коды символов** по текущей таблице символов. Только буквы Ё и ё надо сравнивать отдельно, чтобы поставить их в нужное место алфавита в упорядоченной строке.

Для изменения *регистра* символов в строке при сравнении используйте *функции ANSISupperCase* и *ANSILowerCase* из модуля *SysUtils*. Для символов *латинского* алфавита ещё есть четыре функции: для строки – это *UpperCase* и *LowerCase*, и для одного символа – *UpCase* и *LowCase*

Как отсортировать символы строки, чтобы в ней символы шли в следующем порядке (по вариантам):

№	Сначала идут символы	Затем символы	А в конце – символы
1	Латинские строчные в алфавитном порядке (a..z)	Кириллицы строчные в исходно заданном порядке, включая ё	Латинские ЗАГЛАВНЫЕ в порядке, обратном алфавитному (Z..A)
2	Пробелы	Латинские строчные в исходно заданном порядке	Кириллицы ЗАГЛАВНЫЕ в алфавитном порядке (А..Ё..Я)
3	Кириллицы строчные в алфавитном порядке (а..ё..я)	Пробелы	Цифры в исходно заданном порядке
4	Латинские строчные в алфавитном порядке (a..z)	Латинские ЗАГЛАВНЫЕ в порядке обратном алфавитному (Z..A)	Кириллицы ЗАГЛАВНЫЕ в исходно заданном порядке, включая Ё
5	Латинские строчные в порядке, обратном алфавитному (z..a)	Пробелы	Латинские ЗАГЛАВНЫЕ в алфавитном порядке (A..Z)
6	Пробелы	Латинские строчные в порядке, обратном алфавитному (z..a)	Латинские ЗАГЛАВНЫЕ в алфавитном порядке (A..Z)
7	Кириллицы строчные в порядке обратном алфавитному (я..ё..а)	Кириллицы ЗАГЛАВНЫЕ в исходно заданном порядке, включая Ё	Пробелы
8	Кириллицы строчные в исходно заданном порядке, включая ё	Кириллицы ЗАГЛАВНЫЕ в алфавитном порядке (А..Ё..Я)	Пробелы

Figure 1 Примеры заданий по вариантам

Лабораторная 2. Типизированный файл:

В типизированном файле, хранящем записи вида **C1**, найти (и вывести) **C2**, и переставить её по правилу **C3** (условия **C1**, **C2** и **C3** – см. в таблице по вариантам).

Для *создания* типизированного файла предусмотреть две возможности:

- считывание данных из заранее подготовленного нестандартного текстового файла,
- добавление данных в типизированный файл в режиме диалога.

Предусмотреть также возможность *просмотра* данных из типизированного файла.

Для написания своей программы воспользуйтесь *примером программы*, приведенным на сайте после вариантов заданий. Для выбора нужного действия в программе использовано *меню*.

Выполнить задание по своего варианта варианта, например:

В столбце **Тип**: п – перечисляемый, с – строковый, ч – числовой, * – по вашему выбору.
 ***Дату** можно представить разными способами: одной *строкой* ('31.12.2015' или '2015/12/31' и т.п.), тремя *целыми числами*, двумя *числами и перечисляемым* типом месяца, типом *TDataTime*.

№	Тип	C1	C2	C3
1	п с ч	Вид транспорта (автобус, трамвай и т.д.) Номер маршрута (например, 10, 10А, А) Количество остановок до ближайшего метро	Первую запись о выбранном виде транспорта с наименьшим количеством остановок до метро	С k-ой записью файла
2	с п п ч	Группа (например, А-07-14, ЭР-1-14 и т.д.) Предмет (физика, математика и т.д.) Оценка (отл, хор, удовл, неуд) Количество студентов (из указанной Группы с указанной оценкой по Предмету)	Первую запись о группе с наименьшим количеством неудовлетворительным количеством оценок по выбранному предмету	С последней записью файла
3	п с *	Существо (вампир, оборотень и т.д.) Имя Дата рождения	Последнюю запись с самым старым вампиром	Перед первой записью файла (со сдвигом)
4	п с ч	Вид транспорта (автобус, трамвай и т.д.) Номер маршрута (например, 10, 10А, А) Количество остановок до ближайшего метро	Последнюю запись о выбранном виде транспорта с наибольшим количеством остановок до метро	С первой записью файла
5	с ч п ч	Событие Число Месяц (январь, февраль и т.д.) Год	Первую запись выбранного месяца с событием, содержащим слово «экзамен» в любом регистре	С k-ой записью файла

Figure 2 Примеры вариантов заданий

Контрольные вопросы/задания:

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
Уметь: использовать и редактировать типизированные файлы из записей на языке Паскаль/Delphi	<ol style="list-style-type: none"> 1.Поясните указанную преподавателем строку своей программы. (1-2) 2.Как можно открыть типизированный файл? Текстовый файл? (2) 3.В чем отличия процедуры ReSet для открытия текстового и типизированного файлов? (2) 4.Есть ли отличия считывания записи из текстового и типизированного файлов? (2) 5.Есть ли отличия записывания записи в текстовый и типизированный файлы? (2) 6.Можно ли дописать информацию в конец текстового файла? Типизированного файла? Как? (2) 7.Можно ли дописать информацию в начало текстового файла? Типизированного файла? Как? (2) 8.Допустим, в типизированном файле находятся три записи. Если перейти в начало файла и записать туда одну запись, сколько записей будет в файле – три или четыре? Почему? (2)
Уметь: обрабатывать строки и множества на языке Паскаль/Delphi	<ol style="list-style-type: none"> 1.Как поменять регистр символов латиницы в строке, не меняя символы кириллицы? (1) 2.Как поменять регистр символов латиницы и кириллицы? (1) 3.Как создать новый модуль? (1) 4.Как добавить существующий модуль в проект? (1)

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
	<p>5.Каким образом можно узнать код символа по своей таблице символов, представляющей собой строку? (1)</p> <p>6.Можно ли было для сортировки строки со сложным порядком использовать другие два изученные нами в прошлом семестре способа сортировки? Обладают ли они свойством устойчивости, чтобы оставить исходно заданный порядок нужной части? (1)</p> <p>7.В каком порядке в таблице символов Windows расположены заглавные и строчные символы латиницы? Кириллицы? Цифры? (1)</p> <p>8.В вашей программе использованы статические или динамические строки? Стоки в программе состоят из однобайтовых или двухбайтовых символов? (1)</p> <p>9.Как описать статическую и как описать динамическую строки? (1)</p>

Описание шкалы оценивания:

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: Если минимальная оценка по Лаб.1-2 это "5".

Оценка по Лаб.1-2: оценка "отлично" выставляется, если задание выполнено в полном объеме или выполнено преимущественно верно и не содержит лишних действий, и студент легко отвечает на вопросы по программе и спецификации. Недочеты исправлены в указанные сроки с первой попытки

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: Если минимальная из двух оценок по Лаб.1-2 это "4". Оценка "хорошо" по Лаб.1-2 выставляется, если задание выполнено в полном объеме и выполнено преимущественно верно, и может ответить преимущественно верно на заданные контрольные вопросы, но не все недочеты устранены на момент простановки оценки. И для устранения недочетов потребовалось не более двух попыток

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: Если минимальная из двух оценок по Лаб.1-2 это "3". Оценка "удовлетворительно" по Лаб.1-2 выставляется, если задание преимущественно выполнено, но есть несколько замечаний и для устранения недочетов потребовалось более двух попыток, либо не на все вопросы студент может ответить, либо время сдачи работы просрочено на три недели и более по неуважительной причине

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: Если одна из Лаб.работ1-2 не выполнена вовсе, либо выполнена на "2". Оценка "неудовлетворительно" по Лаб.1-2 выставляется, если задание выполнено неверно или преимущественно не выполнено, или студент не понимает собственный сдаваемый код и не может ответить на большинство контрольных вопросов.

КМ-2. КМ2-2 Связный линейный список (Лаб..3 , Паскаль)

Формы реализации: Защита задания

Тип контрольного мероприятия: Домашнее задание

Вес контрольного мероприятия в БРС: 10

Процедура проведения контрольного мероприятия: Студент 5-10 минут демонстрирует на лабораторном занятии свой код на заранее разработанных своих тестах и отвечает на контрольные вопросы. Без защиты работа не принимается!.

Краткое содержание задания:

Лабораторная 3:

Описать **собственный тип C1**-связного линейного списка.

Создать и заполнить данными **C1**-связный линейный список, упорядоченный по **C2**, из **N C3 C4** и найти элемент, удовлетворяющий условию **C5**. Вывести созданный список или сообщение: «Список пуст». Для **двусвязных** списков – созданный список выводить **дважды**: в прямом и обратном порядке.

Для заполнения списка предусмотреть: 1) возможность ввода данных в список из **текстового файла**; 2) **генерации** элементов в заданном количестве **N**; 3) добавления одного элемента в список в указанном порядке **C2** в режиме диалога.

Для решения задачи создать процедуры: **добавления первого** элемента в пустой список, **добавления очередного** элемента в требуемом порядке (в начало/конец/середину), **генерация списка** (генерация/ввод и вызов процедуры добавления для каждого элемента списка), **вывода** списка, **поиска** в списке, **освобождения** списка. Для односвязного списка добавить процедуру удаления последних **K** элементов (значение **K** запросить у пользователя в режиме диалога).

Типы списков и способов упорядочивания по вариантам:

№	C1	C2	C3*	C4	C5
1	Одно-	По возрастанию абсолютной величины	Псевдослучайных, от -20 до +55	целых чисел	Номер первого кратного 5
2	Дву-	По возрастанию	Псевдослучайных, от -19,9 до +59,9	вещественных чисел	Номер последнего ненулевого элемента
3	Одно-	По возрастанию кодов символов	Псевдослучайных, от а до я, включая ё	символов кириллицы	Номер первой гласной буквы
4	Дву-	По алфавиту	Псевдослучайных, из массива имен	слов – мужских имен	Номер последнего из имен «Иван»
5	Одно-	По убыванию	Псевдослучайных, от -100 до +100	целых чисел	Номер последнего четного
6	Дву-	По убыванию абсолютной величины	Псевдослучайных, от -99,9 до +99,9	вещественных чисел	Номер первого нулевого элемента
7	Одно-	По убыванию кодов символов	Псевдослучайных, от А до Z	символов	Номер последнего символа «В»

Контрольные вопросы/задания:

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
Уметь: работать с указателями, выделять, освобождать память и создавать динамические связные списки на языке Паскаль/Delphi	<ol style="list-style-type: none">1. Как отличить односвязный список от двусвязного?2. Как переделать односвязный список в двусвязный?3. Куда будет проще и быстрее добавить элемент в начало – в список или одномерный массив? Сколько элементов надо изменить при этом в списке? В массиве?4. Как в односвязном списке выполнить переход к предыдущему элементу, если есть только ссылка на следующий?5. Как в односвязном списке удалить последний элемент?6. Как добавить новый элемент в упорядоченный список, не нарушая порядка?7. Поясните указанную преподавателем строку своей программы.

Описание шкалы оценивания:

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: Оценка "отлично" выставляется, если задание выполнено в полном объеме или выполнено преимущественно верно и не содержит лишних действий, и студент легко отвечает на все вопросы по программе. Недочеты исправлены в

указанные сроки с первой попытки. Есть уверенность в том, что студент сможет свободно решить задачу со списком на экзамене.

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: Оценка "хорошо" выставляется, если задание выполнено в полном объеме и выполнено преимущественно верно, и может ответить преимущественно верно на заданные контрольные вопросы, но не все недочеты устранены на момент простановки оценки. И для устранения недочетов потребовалось не более двух попыток

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется, если задание преимущественно выполнено, но есть несколько замечаний и для устранения недочетов потребовалось более двух попыток, либо не на все вопросы студент может ответить, либо время сдачи работы просрочено на три недели и более по неубажительной причине

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: Оценка "неудовлетворительно" выставляется, если задание выполнено неверно или преимущественно не выполнено, или студент не понимает собственный сдаваемый код и не может ответить на большинство контрольных вопросов

КМ-3. КМ2-3 Знакомство с C/C++ (Лаб.4-8, язык Си)

Формы реализации: Смешанная форма

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 35

Процедура проведения контрольного мероприятия: Рассчитать оценку за последовательное в течение не менее 5 недель выполнение и защиту Лабораторных работ 4-8: от простейшего приложения до более сложного). По каждой лабораторной работе студент 5-7 минут демонстрирует на лабораторном занятии свой код на заранее разработанных своих тестах и отвечает на контрольные вопросы. Если не успевает, то может сдать спецификацию с программой на проверку в бумажном и электронном виде, но при наличии существенных ошибок они обсуждаются на лабораторном занятии за компьютером.

Краткое содержание задания:

Последовательное в течение не менее 5 недель выполнение и защита Лабораторных работ 4-8: от простейшего приложения до более сложного):

Лабораторная 4. Первая программа на Си:

Изучить инструкцию по созданию консольного приложения на C/C++ в среде программирования (RAD Studio).

Для динамического одномерного массива X из N ($0 < N \leq 20$) элементов составить (изучить и исправить пример) алгоритм и программу (консольное приложение на Си) нахождения суммы ряда с заданной точностью E ($0 < E < 0.1$) для каждого из элементов X . **Вывести и использовать рекуррентные соотношения при вычислении очередного члена ряда.** Вычисление суммы $Sum(X_i)$ заканчивается, если модуль очередного слагаемого оказывается меньше заданного значения точности (E), причем для этих рядов (при $|X_i| < 1$) абсолютная величина суммы всех отброшенных членов ряда при этом оказывается меньше E .

Для оценки правильности результата предусмотреть вычисление по контрольной формуле $F(X_i)$.

Результаты вычислений вывести на экран в виде таблицы.

Примеры рядов и функций (по вариантам):

№	Ряд	Контрольная формула
1	$x - \frac{2}{6}x^2 + \frac{2 \cdot 5}{6 \cdot 9}x^3 - \dots \pm \frac{2 \cdot 5 \dots (3i-4)}{6 \cdot 9 \dots 3i}x^i \mp \dots$	$\sqrt[3]{1+x-3}$
22	$1 - \frac{5}{2}x + \frac{5 \cdot 7}{2 \cdot 4}x^2 - \frac{5 \cdot 7 \cdot 9}{2 \cdot 4 \cdot 6}x^3 + \dots \pm \frac{5 \cdot 7 \cdot 9 \dots (2i+3)}{2 \cdot 4 \cdot 6 \dots 2i}x^i \mp \dots$	$\sqrt[3]{1+x-3}$
23	$1 - \frac{3x^2}{2!} + \frac{5x^4}{4!} - \frac{7x^6}{6!} + \dots \pm \frac{(2i+1)x^{2i}}{(2i)!} \mp \dots$	$\cos x$
24	$\frac{x^2}{1 \cdot (2 \cdot 1 - 1)} - \frac{x^4}{2 \cdot (2 \cdot 2 - 1)} + \frac{x^6}{3 \cdot (2 \cdot 3 - 1)} - \dots \pm \frac{x^{2i}}{i \cdot (2i - 1)} \mp \dots$	$2x \cdot \arctg x$
25	$x - \frac{1}{4}x^2 + \frac{1 \cdot 3}{4 \cdot 6}x^3 - \frac{1 \cdot 3 \cdot 5}{4 \cdot 6 \cdot 8}x^4 + \dots \pm \frac{1 \cdot 3 \cdot 5 \dots (2i-3)}{4 \cdot 6 \cdot 8 \dots 2i}x^i \mp \dots$	$2\sqrt{1-x}$
26	$\frac{2x^6}{3!} - \frac{4x^{10}}{5!} + \frac{6x^{14}}{7!} - \frac{8x^{18}}{9!} + \dots \pm \frac{2i \cdot x^{4i+2}}{(2i+1)!} \mp \dots$	$\sin x^2$
27	$\frac{x(3+x)}{3!} - \frac{x^3(5+x)}{5!} + \frac{x^5(7+x)}{7!} - \dots \pm \frac{x^{2i-1}(2i+1+x)}{(2i+1)!} \mp \dots$	$1 - \cos x$

Лабораторная 5. Функции, параметры и модули:

Решить задачу своего варианта из нижеприведенного перечня, создав многомодульное консольное приложение для MS Windows на языке C или C++. При этом:

- 1) Ввод исходных данных выполнить из нестандартного текстового файла.
 - 2) Вывод исходных данных и результатов также выполнить в нестандартный текстовый файл.
 - 3) До конца семестра для работы с файлами использовать только тип FILE, а не классы (fstream, ifstream, ofstream). А также не использовать классы string, list, stack и т.д. Вообще не использовать классы – не свои, не чужие; изучаем процедурное программирование на двух языках высокого уровня.
 - 4) Использовать только динамические массивы, а не статические. В условиях задач N, M, K – переменные, а не константы, и их значения надо также узнать из файла с массивом.
 - 5) Аномальные ситуации можно не обрабатывать.
 - 6) Для решения задачи выделить не менее двух подзадач, не считая подзадач ввода, вывода, выделения/освобождения памяти и проверки аномалий. Все подзадачи оформить в виде функций.
 - 7) Внутри функций не использовать глобальных переменных напрямую: вся связь с вызывающей программой только через интерфейс функции – её параметры и возвращаемое значение.
 - 8) Не смешивать внутри функций решение задачи с вводом или выводом: либо только ввод и/или вывод, либо только решение (поиск, вычисления и т.д.) без ввода дополнительных значений и без вывода результата.
 - 9) Все функции расположить не в коде головного модуля программы (рядом с main), а в отдельных файлах (*.h и *.c/*.cpp), в одном или нескольких.
 - 10) Создать краткий отчет по данной работе в рукописном или печатном виде, из условий задачи и подзадач, блок-схем и функциональных тестов.
- Примеры условий задач по вариантам:

3. Даны три последовательности: X_1, X_2, \dots, X_K ; Y_1, Y_2, \dots, Y_K и M_1, M_2, \dots, M_K . Каждая тройка элементов (X_i, Y_i, M_i) представляет параметры одной из K материальных точек, лежащих в плоскости XOY : абсциссу X_i , ординату Y_i и массу M_i . Если абсциссы и ординаты всех точек положительны, найти (XC, YC) – координаты центра тяжести данной системы масс по формулам:

$$XC = \frac{\sum_{i=1}^K X_i M_i}{\sum_{i=1}^K M_i}; YC = \frac{\sum_{i=1}^K Y_i M_i}{\sum_{i=1}^K M_i}$$

4. Дана матрица P с двумя строками и K столбцами, каждым столбцом которой задана абсцисса и ордината одной из K точек плоскости. Если нет ни одной пары точек, расстояние между которыми меньше заданной величины R , заменить на ноль в матрице P все отрицательные абсциссы точек, увеличив ординаты этих точек на R .
5. Дана матрица A из N строк и M столбцов. Если среднее арифметическое матрицы A положительно, задать элементам C_1, C_2, \dots, C_i значения тех элементов матрицы A , которые больше этого среднего арифметического.
6. Дана матрица B из N ($N > 1$) строк и N столбцов. Если в матрице B элемент с максимальным значением (среди всех элементов этой матрицы) лежит выше главной диагонали, найти сумму элементов матрицы, лежащих ниже главной диагонали.
7. Дана матрица A из N строк и M столбцов. Найти среднее арифметическое элементов матрицы A , и, если матрица A не содержит ни одного отрицательного элемента, изменить элементы матрицы путем вычитания из них этого среднего арифметического.
8. Дана матрица B из M строк и M столбцов. Если все элементы главной диагонали матрицы B отрицательны, разделить все элементы матрицы на максимальный по абсолютной величине элемент матрицы.
9. Дана матрица A из N строк и N столбцов. Если разность максимального и минимального элемента матрицы A превышает заданную величину E , заменить в матрице A все отрицательные элементы нулями, а положительные единицами и подсчитать число выполненных замен.

Лабораторная 6. Дополнительная. Выделение и анализ слов и предложений:

Задание: Решить задачу своего варианта из нижеприведенного перечня, создав консольное приложение для MS Windows на языке C или C++.

Пояснение к задачам: В текстовом файле находится некоторый текст, состоящий из нескольких предложений, расположенных на нескольких строках.

Словом будем считать непустую последовательность любых символов в ANSI кодировке, отделенных друг от друга символами-разделителями: пробел, точка, запятая, двоеточие, точка с запятой, конец строки, перевод каретки, табуляция, конец файла. В число разделителей можно также включить все символы до 32-го по таблице символов и скобки разных видов и т.д.

Предложением будем считать непустую последовательность слов, оканчивающуюся точкой, вопросительным или восклицательным знаком, или символом конца файла.

До и после знаков препинания и между слов может находиться один или несколько пробелов и концов строк.

Примеры задач:

- Дан текстовый файл с несколькими строками. Найти номер последнего предложения, в котором все слова начинаются на букву Б (б), или вывести сообщение, что таких предложений нет.
- Дан текстовый файл с несколькими строками. Найти номер первого предложения, где есть слово, начинающееся и заканчивающееся на букву А (а), или вывести сообщение, что таких предложений нет. Замечание: после нахождения нужного предложения, дальше просматривать файл бессмысленно.
- Дан текстовый файл с несколькими строками. Найти номер последнего предложения, где есть слово, начинающееся и заканчивающееся на букву К (к), или вывести сообщение, что таких предложений нет.
- Дан текстовый файл с несколькими строками. Найти номер первого предложения, где у всех слов первая буква совпадает с последней, или вывести сообщение, что таких предложений нет. Замечание: после нахождения нужного предложения, дальше просматривать файл бессмысленно.

Лабораторная 7. Двоичные файлы:

Задание:

Создать тип данных (в виде структуры на языке Си) в соответствии с условием задачи своего варианта из нижеприведенного перечня, создав многомодульное консольное приложение для MS Windows на языке C или C++, которая в интерактивном режиме (меню) выполняет следующие команды пользователя:

- из текстового файла ввести все данные и создать двоичный файл;
- поиск в двоичном файле по условию задачи;
- коррекция данных в двоичном файле (не при вводе из текстового, а именно редактирование двоичного файла);
- просмотр двоичного файла – вывод данных из файла на экран в виде таблицы.

Указание 1. Для ввода строки из *нескольких* слов, используйте *fgets* вместо *fscanf* или *scanf*

Указание 2. Корректировать не в диалоге, а автоматически на значение ближайшей границы или случайное корректное значение.

Примеры задач:

2. *Исходные данные:* Расписание экзаменов содержит следующую информацию: номер студенческой группы, дата и время экзамена, номер аудитории, наименование дисциплины и фамилия преподавателя. *Условие поиска:* Даты, по которым занят конкретный преподаватель (Запросить фамилию в режиме диалога). *Коррекция:* Номер месяца 1-12, дней не более 28-31 в соответствии с месяцем.
3. *Исходные данные:* Сведения о автомобиле состоят из его марки, номера, фамилии владельца и отметки о прохождении техосмотра. *Условие поиска:* Фамилии владельцев и номера автомобилей определенной марки (Запросить марку в режиме диалога), не прошедших вовремя техосмотр. *Коррекция:* Отметка о прохождении техосмотра включает и дату его прохождения. Техосмотр должен быть пройден не более чем год назад. Номер месяца 1-12, дней не более 28-31 в соответствии с месяцем.
4. *Исходные данные:* В расписании поездов указаны: номер поезда, пункт отправления, пункт назначения, время отправления, время в пути, цена билета. *Условие поиска:* Поезда, следующие из Москвы в Санкт-Петербург; время отправления которых с 7.00 до 9.00. *Коррекция:* Время отправления: минуты 0-59, часы 0-23. Названия пунктов отправления и назначения могут состоять из нескольких слов.

Лабораторная 8. Корни уравнений:

Написать отчет, в котором должны присутствовать:

1 условие задачи;

Двумя методами (см. табл.1) для N значений погрешности ϵ ($0,1; 0,01; 0,001; \dots; 1e-N$, $1 \leq N \leq 10$) вычислить значение корня для двух заданных функций (N_0 и N_0+3 по табл.2) на отрезке $[A, B] \subset (0, 2)$ и вывести их в виде таблицы (для каждой функции отдельную):

Номер функции	Название метода		Название метода	
	Корень	Итераций	Корень	Итераций
0,1				
...				
0,0...1				

При вводе значений анализировать аномалии: A, B, N – числа и $0 < A < B < 2$, $1 \leq N \leq 10$. Перед поиском корня обязательно проверять наличие корня на отрезке $[A, B]$ (разные знаки значения функции на концах отрезка), и в случае отсутствия выдавать сообщение «Корня на отрезке АВ нет».

2 условия заданного варианта задания, две функции и названия методов;

3 для каждого метода:

- описание заданного численного метода;
- блок-схему алгоритма подзадачи вычисления корня;
- программный код функции вычисления корня, в которую передается указатель на нужную функцию;

4 сравнение методов:

- результаты вычислений значения корня для обеих функций для заданных 10 вариантов погрешности и количество выполненных итераций (см. таблицу выше);
- вывод: какой метод и чем считаете лучше и почему?

Для самопроверки: 1) для каждой функции указан примерный корень для проверки написания заданной функции, и 2) корни, найденные разными методами с одинаковой точностью не должны отличаться больше, чем на двойную величину точности.

Таблица 1. Выбор метода

М Е Т О Д Ы	1)Метод деления отрезка пополам			
	2) метод простых итераций	2) метод Ньютона (касательных)	2) метод секущих	2) метод хорд
	1	2	3	4
В	5	6	7	8
А	9	10	11	12
Р	13	14	15	16
И	17	18	19	20
А	21	22	23	24
Н	25	26	27	28
Т	29	30	31	32
Ы	33	34	35	

Таблица 2. Функции по вариантам

№	Формула	№	Формула
1	$\frac{\sqrt[3]{4} - \sin^2(x/10)}{\sqrt{x}} - x;$ 1,35041	16	$(\sin x + \cos x)^2 / \sqrt[3]{33,5^2} + \sqrt{\frac{3}{7}} - x;$ 0,8463
2	$\frac{1}{10}e^{-\cos^2 x} + \frac{\sqrt{x/2}}{\ln(x+1)} - x;$ 1,08306	17	$\sqrt{\ln 7,9 + e^{-x} - \frac{x^2}{11}} - x;$ 1,452
3	$\frac{\ln 44,8 - \sin \sqrt{x}}{2} - x;$ 1,43543	18	$\sqrt[3]{e^x - \frac{6}{7}x \cdot \sin x - 0,3} - x;$ 0,742
4	$1,5 - (\sqrt{x} + \sqrt[3]{x}) / e^{3/2} - x;$ 1,04540	19	$\sqrt{1 - \frac{1}{4}x^2 + 4e^{-x-1,5} - \frac{2}{3}} - x;$ 0,708
5	$1/(\sqrt{5} + \sin(0,1x) + \ln(1+x)) - x;$ 0,38462	20	$\frac{1}{3}(e^{-x} - \sqrt{e^x} + 3,7) - x;$ 0,861

Контрольные вопросы/задания:

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
Уметь: выделять подзадачи и описывать функции и их прототипы, вызывать функции и передавать параметры по значению, адресу и ссылке на языке C C++	<ol style="list-style-type: none"> 1.Какие подзадачи есть в вашей работе и как их описать в виде функций на языке C C++? (5) 2.Где надо описать прототипы функций, а где разместить полное описание? Для чего нужны заголовочные файлы (*.h .hpp) и файлы с исходным кодом (*.c *.cpp)? (5) 3.Как передать параметры простого типа по значению? По адресу? По ссылке в C++? (5) 4.Как передать одномерные массивы (указатели) по значению? По адресу? По ссылке в C++? (5) 5.Как передать двухмерные динамические массивы (массив указателей) по значению? По адресу? По ссылке в C++? (5) 6.Как передать двухмерные статические массивы (указатели) по значению? По адресу? По ссылке в C++? (5) 7.В каких случаях используется передача параметров простого типа по значению? По адресу? По ссылке в C++? (5) 8.В каких случаях используется передача параметров-массивов или других указателей по

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
<p>Уметь: работать со строками на языке C/C++, разделять текст на символы, слова и предложения базовыми функциями</p>	<p>значению? По адресу? По ссылке в C++? (5)</p> <ol style="list-style-type: none"> 1.Каким способом вы бы воспользовались для выделения слов при решении задачи – посимвольным или функцией выделения токенов? (6) 2.Какие разделители вы бы выбрали для разделения слов и предложений? (6) 3.Как считать символ из текстового файла? (6) 4.Как считать одно слово/токен из текстового файла? (6) 5.Как считать строку из нескольких слов из текстового файла? (6) 6.Как описать статическую строку? Динамическую? (6) 7.Как выделить память для хранения строки? Как ее освободить? (6) 8.Какой минимальной длины должна быть строка, предназначенная для хранения строк из одного символа – «Я»? (6) 9.Как вычислить длину строки? (6)
<p>Уметь: разрабатывать алгоритмы для обработки двоичных файлов и структур данных на языке C/C++</p>	<ol style="list-style-type: none"> 1.Можно ли считать структуру/запись целиком из текстового файла? Двоичного? Типизированного? Укажите как (7) 2.Как указать тип открываемого файла? Имя файла? (7) 3.Как узнать удачей ли закончилась функция чтения или записи в двоичный/бестиповой файл? (7) 4.С помощью какого оператора выполняется чтение из двоичного/бестипового файла? (7) 5.С помощью какого оператора выполняется запись в двоичный/бестиповой файл? (7) 6.Как узнать текущую позицию в файле? (7) 7.Как изменить текущую позицию в нетекстовом открытом файле? (7)
<p>Уметь: разрабатывать алгоритмы по их словесному описанию и использовать указатели на функции в языке C/C++</p>	<ol style="list-style-type: none"> 1.Поясните указанную преподавателем строку своей программы или спецификации. (4-8) 2.Как описать указатель на функцию в языке Си? (8) 3.Нужно ли обязательно знать формулу обратной функции $g(x)=g(y)$ для функции $y=f(x)$, чтобы вычислить корень с заданной точностью? (8) 4.Будут ли совпадать корни, найденные разными методами с заданной погрешностью E? (8) 5.Какой из методов не учитывает особенности функции и выполняет одинаковое количество итераций для одинаковой длины исходно заданного отрезка и одинакового значения точности вне зависимости от заданной функции? (8)

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
	6. Как вывести в корнях и значениях точности (0,1; 0,01; 0,001; ...) столько знаков в дробной части, каков номер строки таблицы? Пример ищите в Лабораторной работе 4. (8)
Уметь: использовать современное программное средство разработки программ для решения поставленной задачи на языке C/C++	1. Как создать консольное приложение на C/C++? Где расположить функцию main? (4) 2. Как вывести рекуррентное соотношение, чтобы избежать переполнения при вычислениях большого произведения? (4) 3. Как указать формат при выводе числа типа double? (4) 4. Как в формате вывода использовать переменное число знаков z вместо числа-константы? (4) 5. Как выделить память для N элементов одномерного динамического массива? (4) 6. Как освободить память для N элементов одномерного динамического массива? (4) 7. Как правильно записать вычисление одной трети, чтобы результат был вещественным, а не целым и нулевым? (4)

Описание шкалы оценивания:

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: Если минимальная из оценок за Лабораторные 4,5,7,8 это "5", при этом оценка "4"- "5" за лаб.6 позволяет одну из оценок за лаб.4,5,7,8 при подсчете увеличить на один балл. Оценка "отлично" выставляется за Лабораторную работу, если задание выполнено в полном объеме или выполнено преимущественно верно и не содержит лишних действий, и студент легко отвечает на вопросы по программе. Недочеты исправлены в указанные сроки с первой попытки.

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: Если минимальная из оценок за Лабораторные 4,5,7,8 это "4", при этом оценка "4"- "5" за лаб.6 позволяет одну из оценок за лаб.4,5,7,8 при подсчете увеличить на один балл. Оценка "хорошо" выставляется за Лабораторную работу, если задание выполнено в полном объеме и выполнено преимущественно верно, и может ответить преимущественно верно на заданные контрольные вопросы, но не все недочеты устранены на момент простановки оценки. И для устранения недочетов потребовалось не более двух попыток

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: Если минимальная из оценок за Лабораторные 4-8 это "3". Оценка "удовлетворительно" выставляется, если задание преимущественно выполнено, но есть несколько замечаний и для устранения недочетов потребовалось более двух попыток, либо не на все вопросы студент может ответить, либо время сдачи работы просрочено на три недели и более по неуважительной причине. За лаб.6 оценка выставляется по желанию студента (необязательное задание, только для хорошистов и отличников)

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: Если не выполнена хотя бы одна из Лабораторных работ 4,5,7,8 или выполнена на "2": Оценка "неудовлетворительно"

выставляется за лабораторную работу, если задание выполнено неверно или преимущественно не выполнено, или студент не понимает собственный сдаваемый код и не может ответить на большинство контрольных вопросов

КМ-4. КМ2-4 Рекурсия (Лаб.9-10, C|C++)

Формы реализации: Защита задания

Тип контрольного мероприятия: Домашнее задание

Вес контрольного мероприятия в БРС: 18

Процедура проведения контрольного мероприятия: Расчет оценки за выполнение в течение предыдущих 2-3 недель и защиту Лаб.9-10: рекурсивный алгоритм и рекурсивный тип данных: Для каждого задания Лаб.9-10 студент 5-7 минут демонстрирует на лабораторном занятии свой код на заранее разработанных своих тестах, показывает полное дерево вызовов своей рекурсивной функции и отвечает на контрольные вопросы. Если не успевает, то может сдать спецификацию с программой на проверку в бумажном и электронном виде, но при наличии существенных ошибок они обсуждаются на лабораторном занятии за компьютером.

Краткое содержание задания:

В течении 2-3 недель выполнить и защитить программы по заданиям на Лабораторные работы 9-10 про рекурсивный алгоритм и рекурсивный тип данных и стек.

Лабораторная 9. Рекурсивный алгоритм:

Задание. В целях изучения рекурсивного способа описания алгоритмов и использования динамических одномерных массивов написать консольное приложение на *Delphi*, с рекурсивной процедурой/функцией для решения задачи, условие которой приведено ниже. При этом:

- 1.Использовать динамический открытый одномерный массив. Запросить у оператора его длину N , перед выделением памяти для размещения значений элементов массива;
- 2.Ввод исходных данных и вывод результатов осуществить из/в текстовый файл (стандартный либо нестандартный – по выбору студента);
- 3.Изобразить полное дерево рекурсивных вызовов для $N=7+\langle \text{номер учебной группы} \rangle$, например, для А-4-XX $N=7+4=11$. Подсчитать для этого дерева глубину вызовов и объем рекурсии - это характеристики сложности рекурсивных вычислений для конкретного набора параметров. Если деревья могут быть разными, начертить два дерева: наибольшее и любое другое.

Примеры задач:

2.Для заданного одномерного массива A из N элементов найти количество элементов массива, для которых выполняется условие $\sin A_i < \cos A_i$. В рекурсивной функции каждый раз отделять последнюю треть от первых двух третей рассматриваемой части массива и применять эту же функцию к обеим частям. Рекурсивные вызовы заканчивать, когда останется только один элемент в рассматриваемой части массива.

Например, для $N=12$:

$$\underset{i=1}{\overset{12}{\text{Count}}}(\sin A_i < \cos A_i) = \underset{i=1}{\overset{8}{\text{Count}}}(\sin A_i < \cos A_i) + \underset{i=9}{\overset{12}{\text{Count}}}(\sin A_i < \cos A_i)$$

3.Для заданного одномерного массива B из N элементов найти номер минимального значения выражения $B_i^2 + B_i \sin B_i$. В рекурсивной функции каждый раз отделять последнюю треть от первых двух третей рассматриваемой части массива и применять эту же функцию к обеим частям. Рекурсивные вызовы заканчивать, когда останется только один элемент в рассматриваемой части.

Например, для $N=9$: $\underset{i=1}{\overset{9}{\text{Min}}}(B_i^2 + B_i \sin B_i) = \min(\underset{i=1}{\overset{6}{\text{Min}}}(B_i^2 + B_i \sin B_i), \underset{i=7}{\overset{9}{\text{Min}}}(B_i^2 + B_i \sin B_i))$

Лабораторная 10. Стек:

Задание: смоделировать стек на основе собственного типа односвязного списка и решить на этой структуре задачу по своему варианту. Создать интерактивное приложение с возможностью выбора команды: а) ввод данных в стек(и) из стандартного или нестандартного текстового файла(ов); б) вывод данных из всех стеков на экран (стандартный текстовый файл); в) решение задачи; г) освобождение стеков; д) выход. Пример задач:



Контрольные вопросы/задания:

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
<p>Уметь: моделировать информационно-логическую структуру Стек (LIFO) на основе динамического связного списка на языке C/C++</p>	<ol style="list-style-type: none"> 1.Пояснить указанную преподавателем строку программы. (9-10) 2.Как описать тип элемента собственный тип односвязного линейного списка? Какие операции надо написать, чтобы работать с ним как со стеком? (10) 3.Как добавить элемент в стек? (10) 4.Как удалить элемент из стека? (10) 5.Как определить пустоту стека? (10) 6.Как перебирать элементы стека, используя один дополнительный стек? (10) 7.В чём отличия стека от линейного списка? от одномерного массива? (10)
<p>Уметь: разрабатывать рекурсивные алгоритмы и строить полное дерево вызовов, высчитывать глубину вызовов и объем рекурсии</p>	<ol style="list-style-type: none"> 1.Какой алгоритм вычисления суммы одномерного массива эффективней – рекурсивный или циклический? На что тратится память при вызове функций? (9) 2.Почему переполняется стек вызовов? Как этого можно избежать? (9) 3.Каков индекс первого по порядку элемента динамического массива? Статического? Можно ли его изменить на Си или в Delphi? (9) 4.Покажите, где в вашем рекурсивном алгоритме терминальная ветвь, а где – рабочая? (9) 5.Как выделить память для одномерного массива в Си и в Delphi? Как освободить ее? (9)

Описание шкалы оценивания:

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: Если минимальная из двух оценок за Лаб.9-10 это "5". Оценка "отлично" выставляется за лабораторную работу, если задание выполнено в полном объеме или выполнено преимущественно верно и не содержит лишних действий, и студент легко отвечает на вопросы по программе. Недочеты исправлены в указанные сроки с первой попытки. Если есть уверенность, что студент легко сможет решить аналогичную задачу со стеком на экзамене.

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: Если минимальная из двух оценок за Лаб.9-10 это "4". Оценка "хорошо" выставляется за лабораторную работу, если задание выполнено в полном объеме и выполнено преимущественно верно, и может ответить преимущественно

верно на заданные контрольные вопросы, но не все недочеты устранены на момент простановки оценки. И для устранения недочетов потребовалось не более двух попыток

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: Если минимальная из двух оценок за Лаб.9-10 это "3". Оценка "удовлетворительно" выставляется за лабораторную работу, если задание преимущественно выполнено, но есть несколько замечаний и для устранения недочетов потребовалось более двух попыток, либо не на все вопросы студент может ответить, либо время сдачи работы просрочено на три недели и более по неважительной причине

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: Если одна из двух работ Лаб.9-10 ещё не выполнена или выполнена/защищена на "2". Оценка "неудовлетворительно" выставляется за лабораторную работу, если задание выполнено неверно или преимущественно не выполнено, или студент не понимает собственный сдаваемый код и не может ответить на большинство контрольных вопросов.

КМ-5. КМ2-5 Итоговое индивидуальное задание (2 семестр)

Формы реализации: Защита задания

Тип контрольного мероприятия: Расчетно-графическая работа

Вес контрольного мероприятия в БРС: 25

Процедура проведения контрольного мероприятия: Студент показывает свой отчет-спецификацию и в течении 10-20 минут отвечает на контрольные вопросы о спецификации, предлагает пути устранения найденных в спецификации и в программе ошибок. При необходимости возможна демонстрация студентом правильности создания, работы и решения своей многомодульной программы на компьютере, ее отладка. Работа по требованию преподавателя может быть заранее сдана на проверку для сокращения времени обсуждения на занятии.

Краткое содержание задания:

Выполнение и защита индивидуального задания на моделирование нового типа(список, очередь, стек, дек, бинарное дерево, матрица) двумя способами на основе двух других типов(одно- и двух-мерный массив, линейный и нелинейный связный список, типизированный/бестиповой или двоичный файл). Написание программ (на одном(или двух для эталонной группы) из двух изученных языков высокого уровня: Паскаль или Си) и отчета по проделанной работе. Например, смоделировать (на основе знаний этого семестра и этой дисциплины: только процедурное программирование, без объектно-ориентированного) новый тип "Стек" на основе динамического массива и на основе типизированного файла и затем решить задачу со стеком, используя смоделированный тип данных и операции для работы с ним.

Контрольные вопросы/задания:

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
Знать: процедурное программирование на двух языках высокого уровня - Паскаль/Delphi и C/C++	1. Когда при моделировании типа удобно использовать базовый тип - динамический массив? Насколько просты функции добавления, удаления, обработка? 2. Когда при моделировании типа удобно использовать базовый тип - односвязный линейный список? Насколько просты функции добавления, удаления, обработка? 3. Когда при моделировании типа удобно

Запланированные результаты обучения по дисциплине	Вопросы/задания для проверки
	<p>использовать базовый тип - двусвязный линейный список? Насколько просты функции добавления, удаления, обработка?</p> <p>4.Когда при моделировании типа удобно использовать базовый тип - двусвязный НЕлинейный список? Или даже трёхсвязный? Насколько просты функции добавления, удаления, обработка?</p> <p>5.Когда при моделировании типа удобно использовать базовый тип - НЕтекстовый файл? Насколько просты функции добавления, удаления, обработка?</p> <p>6.Реализация, выполненная на каком базовом типе будет работать быстрее: на основе динамического массива или нетекстового файла? Какая память быстрее?</p> <p>7.С помощью какого типа можно смоделировать более длинную линейную структуру данных: на основе массива, списка или файла? Какая память объемнее?</p>

Описание шкалы оценивания:

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: Оценка "отлично" выставляется, если задание выполнено в полном объеме или выполнено преимущественно верно и не содержит лишних действий, и студент легко отвечает на вопросы по программе. Недочеты исправлены в указанные сроки с первой попытки. На вопросы даны развернутые ответы при защите работы

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: Оценка "хорошо" выставляется, если задание выполнено в полном объеме и выполнено преимущественно верно, и может ответить преимущественно верно на заданные контрольные вопросы, но не все недочеты устранены на момент простановки оценки. На все вопросы даны верные ответы при защите, но неполные или содержат неточности и нечеткости

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется, если задание преимущественно выполнено, но есть несколько замечаний и для устранения недочетов потребовалось более двух попыток, либо не на все вопросы студент может ответить, либо выполнена только половина задания (один из способов), либо время сдачи работы просрочено на три недели и более по неуважительной причине и идёт экзаменационная сессия

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: Оценка "неудовлетворительно" выставляется, если задание выполнено неверно или преимущественно не выполнено, или студент не понимает собственный сдаваемый код и не может ответить на большинство контрольных вопросов.

СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

1 семестр

Форма промежуточной аттестации: Экзамен

Пример билета

1. Алгоритм и его основные свойства.
2. Классификация типов данных в Паскале. Типы с плавающей точкой.
3. Разработать нисходящим способом и написать программный код с несколькими процедурами/функциями и статическими массивами для предложенной задачи.

Задача:

Заданы матрицы $A(n \times m)$ и $B(s \times t)$, массив $C(k)$.

Если максимальный из элементов массива C больше хотя бы одного из элементов матрицы A и хотя бы одного из элементов матрицы B , то вывести сообщение «Максимум больше», иначе в каждой из матриц найти произведение нечетных элементов.

Процедура проведения

Экзаменуемый получает экзаменационный билет с двумя теоретическими вопросами и одной задачей.

За 60 минут он должен кратко записать ответы на два первых вопроса и для решения задачи: начертить блок-схему верхнего уровня с выделенными подзадачами и написать программный код на языке Delphi полностью или основную программу плюс одну-две (указанные экзаменатором по блок-схеме студента) подзадачи в виде процедур с параметрами и заголовки всех остальных процедур.

Если за 60 минут студент еще не полностью готов к ответу, экзаменатор может ему дать дополнительные 30 минут для подготовки, если большая часть ответа уже написана, или немного упростить задачу, если экзаменуемый затрудняется с решением.

При опросе студента экзаменатор может задать вопросы по всей первой части предмета, если ответов по вопросам билета ему не хватило для точной оценки знаний студента.

Требования к решению задачи:

1. Все процедуры/функции должны иметь ПАРАМЕТРЫ и не должны использовать глобальные переменные.
2. Не совмещайте в теле процедур решение (поиск, вычисления) с вызовом процедур с вводом или выводом.
3. Каждую подзадачу надо оформить отдельной процедурой/функцией. При этом тело достаточно написать у той, что укажет экзаменатор по блок-схеме, для остальных подзадач достаточно написать заголовки процедур(функций) с параметрами.
4. Можно использовать только статические массивы на экзамене за первый семестр

1. Перечень компетенций/индикаторов и контрольных вопросов проверки результатов освоения дисциплины

1. Компетенция/Индикатор: ИД-1_{ОПК-8} Демонстрирует знание основных языков программирования и работы с базами данных, операционных систем и оболочек, современных программных сред разработки информационных систем и технологий

Вопросы, задания

1. Описание констант, переменных и пользовательских типов. Области видимости констант и переменных
2. Имена процедур и функций как фактические параметры процедур (Процедурный тип)
3. Массивы и записи как формальные параметры процедур и функций
4. Классы формальных параметров: параметры-константы, параметры-значения и параметры переменные. Ключевые слова `const`, `var`, `out` при описании параметров
5. Описание и вызов функций. Отличия от процедур общего вида
6. Описание и вызов процедур. Отличия от функций
7. Назначение и отличия процедур общего вида и функций
8. Тип множество: описание, ввод, вывод, операции над множествами
9. Тип запись: описание, ввод, вывод. Оператор `With`. Запись с вариантами
10. Тип массив (статический): описание, ввод, вывод. Форматный вывод
11. Порядковые типы. Перечисляемый тип
12. Логический тип. Логические операторы и операции сравнения
13. Порядковые типы. Символьный тип. Таблицы символов
14. Порядковые типы. Целые типы в Паскале, тип диапазон
15. Классификация типов данных в Паскале. Типы с плавающей точкой (`Double`, `Single`, `Real`, `Extended`)
16. Операторы ветвления с единственным и множественным выбором: `if` и `case`
17. Операторы ветвления с единственным и множественным выбором: `if` и `case`
18. Виды циклов в языке Паскале. Итерационные циклы (`repeat until` и `while do`)
19. Виды циклов в языке Паскале. Параметрические циклы с шагом `+1` и `-1`
20. Базовые структуры алгоритмов (базис Вирта, базис Дейкстры). В виде блок-схем и на Паскале
21. Алгоритм и его основные свойства
22. Описание констант структурированных типов: массивов, записей и множеств
23. Спецификация задачи: состав и назначение

Материалы для проверки остаточных знаний

1. (Delphi) Какой из нижеприведенных четырех типов является простым порядковым типом? (выберите только один ответ)

Type

TA = array [0..9] of char;

TB = set of 0..9;

TC = (sun, mon, tue, wed, thu, fri, sat);

TD = record x, y: integer end;

Ответы:

1) TA

2) TB

3) TC

4) TD

Верный ответ: 3

2. (Delphi) Какой вид цикла использован в приведенном коде? (выберите только один ответ)

m:=a[1];

```
for i:=2 to n do
  if m>a[i] then m:=a[i];
```

Ответы:

- 1) здесь нет цикла
- 2) итерационный цикл с предусловием
- 3) итерационный цикл с постусловием
- 4) цикл с параметром

Верный ответ: 4

3.(Delphi) Какой вид цикла использован в приведенном коде? (выберите только один ответ)

```
m:=a[1]; i:=2;
repeat
  if m>a[i] then m:=a[i];
  inc(i);
until i>n;
```

Ответы:

- 1) цикл if
- 2) итерационный цикл с предусловием
- 3) итерационный цикл с постусловием
- 4) цикл с параметром

Верный ответ: 3

4.(Delphi) Значения переменной Z будут уменьшаться или возрастать, если точно известно, что тело этого цикла выполнится четыре раза? (выберите только один ответ)

```
i:=1;
for Z:=N1 downto N2 do
  if Z mod 2 = 0 then i:=i+2;
```

Ответы:

- 1) уменьшаться
- 2) возрастать

Верный ответ: 1

2. Компетенция/Индикатор: ИД-2_{ОПК-8} Применяет языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ

Материалы для проверки остаточных знаний

1. Какими способами может быть открыт текстовый файл? (выберите только один ответ)

Ответы:

- 1) только для чтения, только для записи с предварительным созданием, одновременно для чтения и записи
- 2) только для чтения, только для записи с предварительным созданием, только для дозаписи в конец существующего файла

Верный ответ: 2

3. Компетенция/Индикатор: ИД-3_{ОПК-8} Применяет методы и технологии отладки и оптимизации программного обеспечения

Вопросы, задания

1. Составление функциональных и структурных тестов
2. Способы структуризации алгоритмов. Сведение алгоритма с досрочным выходом из цикла for к циклам while и repeat (метод флажка)
3. Критерии качества программного продукта

Материалы для проверки остаточных знаний

1.(Delphi) Чем отличаются блоки try..except и try..finally? (выберите только один ответ)

Ответы:

- 1) блок except выполняется только в случае возникновения ошибки времени выполнения программы в соответствующем блоке try, а блок finally - всегда
- 2) блок finally выполняется только в случае возникновения ошибки в блоке try во время выполнения программы, а блок except - в любом случае

Верный ответ: 1

2.Продолжите фразу. Тесты - это ... (выберите только один ответ)

Ответы:

- 1) любые исходные данные
- 2) исходные данные и ожидаемый результат

Верный ответ: 2

3.Продолжите предложение: Для составления функциональных тестов... (выберите только один ответ)

Ответы:

- 1) не нужен текст программы, а только постановка задачи и информация о входных данных и результатах - это принцип тестирования "черного ящика"
- 2) нужен текст программы или информация о принципах ее работы - это тестирование "белого ящика"

Верный ответ: 1

4. Компетенция/Индикатор: ИД-1_{ОПК-9} Демонстрирует знание методов и способов использования программных средств для решения практических задач

Вопросы, задания

- 1.Модули в Паскале: назначение, описание, использование. Обязательные и дополнительные разделы
- 2.Этапы разработки программ. Цикл разработки программного обеспечения
- 3.Суть методов нисходящего и восходящего проектирования

Материалы для проверки остаточных знаний

1.(Delphi) В чем отличие процедур общего вида от функций? (выберите только один ответ)

Ответы:

- 1) процедурам не нужны параметры
- 2) функция возвращает в точку вызова не только управление, но и результат вычислений
- 3) процедура общего вида возвращает в точку вызова не только управление, но и результат вычислений

Верный ответ: 2

2.(Delphi) Чтобы обособить подзадачи, сократив код вызывающей их программы, и получить возможность повторного использования кода нужны ____ . (выберите только один ответ)

Ответы:

- 1) типы, константы и переменные
- 2) массивы, записи и множества
- 3) процедуры, функции и модули

Верный ответ: 3

5. Компетенция/Индикатор: ИД-2_{ОПК-9} Использует программные средства для решения практических задач

Вопросы, задания

1. Разработать нисходящим способом и написать программный код на Delphi с несколькими процедурами/функциями и статическими массивами для предложенной задачи. Примеры задач:

1. Заданы матрицы $A(n \times m)$ и $B(s \times t)$, массив $C(k)$.

Если максимальный из элементов массива C больше хотя бы одного из элементов матрицы A и хотя бы одного из элементов матрицы B , то вывести сообщение «Максимум больше», иначе в каждой из матриц найти произведение нечетных элементов.

2. Заданы матрицы $A(n \times l)$ и $B(m \times t)$, массив $C(k)$.

Если в массиве C все элементы кратны пяти, то вывести сообщение «Все кратны», иначе в обеих матрицах проверить наличие элемента, равного первому некрратному пяти элементу массива C : в матрице A – в верхней половине, а в матрице B – в нижней.

3. Заданы символьные матрицы $A(n \times n)$ и $B(\text{txt})$.

Если максимальный (по коду символа) из элементов, лежащих выше главной диагонали матрицы A меньше максимального из элементов, лежащих выше главной диагонали матрицы B , то вывести сообщение «Максимум A меньше», иначе на главной диагонали матрицы B найти номер первого символа 'Q', или вывести сообщение, что такого символа там нет

2. Нисходящее и восходящее тестирование программ. Достоинства и недостатки.

Использование заглушек и драйверов

3. Текстовый файл: описание файловой переменной, основные операции. Использование параметров программы для передачи программе имен файлов

Материалы для проверки остаточных знаний

1. Какой способ поиска и локализации ошибок отсутствует в среде разработки программ? (выберите только один ответ)

Ответы:

- 1) пошаговое выполнение программы
- 2) точки останова
- 3) просмотр значений переменных в ходе пошаговой работы программы
- 4) сравнение программного кода с условием задачи

Верный ответ: 4

II. Описание шкалы оценивания

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: оценки «отлично» заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание материалов изученной дисциплины, умение свободно выполнять задания, предусмотренные программой, усвоивший основную и знакомый с дополнительной литературой, рекомендованной программой. Как правило, оценка «отлично» выставляется студенту, проявившему творческие способности в понимании, изложении и использовании материалов изученной дисциплины, безупречно ответившему не только на вопросы билета, но и на дополнительные вопросы в рамках основной программы дисциплины экзамена, правильно выполнившему практическое задание

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: оценки «хорошо» заслуживает студент, обнаруживший полное знание материала изученной дисциплины, успешно выполняющий предусмотренные в программе задания, усвоивший основную литературу, рекомендованную в программе. Как правило, оценка «хорошо» выставляется студенту, показавшему систематический характер знаний по дисциплине, ответившему на все вопросы билета, правильно выполнившему практическое задание, но допустившему при этом не принципиальные и небольшие ошибки и сумевшему самостоятельно их исправить после указания на них

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: оценки «удовлетворительно» заслуживает студент, обнаруживший знание материала изученной дисциплины в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справляющийся с выполнением заданий, предусмотренных программой. Как правило, оценка «удовлетворительно» выставляется студентам, допустившим погрешность или неполноту в ответе на теоретические вопросы или отсутствие ответа на один из теоретических вопросов билета и/или принципиальные ошибки при выполнении практических заданий, но обладающим необходимыми знаниями для их устранения под руководством преподавателя, либо неправильно выполнившему практическое задание, но с помощью подсказок преподавателя сумевшему доделать часть большую часть работы или по указанию экзаменатора выполнившим другие практические задания из того же раздела дисциплины, знающий базовые типы и управляющие структуры языка (Паскаль, Delphi) и базовые алгоритмы обработки массивов

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: оценка «неудовлетворительно» выставляется студенту, обнаружившему серьезные пробелы в знаниях основного материала изученной дисциплины, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий. Как правило, оценка «неудовлетворительно» ставится студентам, не ответившим на теоретических вопросы билета и дополнительные вопросы, и неправильно выполнившему практическое задание, а также не сумевшему решить и более упрощенную экзаменатором версию даже с помощью подсказок преподавателя, не знающему базовые типы и управляющие структуры языка программирования (Паскаль/Delphi) и базовые алгоритмы обработки разных структур данных. Неправильное или правильное выполнение только практического задания не является однозначной причиной для выставления оценки.

Требуется устный ответ по написанному при подготовке к ответу. Оценка «неудовлетворительно» выставляется также, если студент: - после начала экзамена отказался его сдавать; - нарушил правила сдачи экзамена (списывал, подсказывал, обманом пытался получить более высокую оценку и т.д.).

III. Правила выставления итоговой оценки по курсу

Оценка определяется в соответствии с Положением о балльно-рейтинговой системе для студентов НИУ «МЭИ» на основании семестровой и экзаменационной составляющих. В приложение к диплому выносятся оценка за 2 семестр.

2 семестр

Форма промежуточной аттестации: Экзамен

Пример билета

1. Рекурсия. Рекурсивный алгоритм, дерево рекурсивных вызовов, глубина вызовов и объем

рекурсии. Стек рекурсивных вызовов: переполнение и способы его предотвращения. Взаимно рекурсивные процедуры/функции. Предварительное описание процедур/функций в Паскале (forward) и прототипы в С. Приведите пример описания взаимно рекурсивных процедур/функций.

2. Написать процедуру/функцию на Delphi или С для решения нижеприведенной задачи. Исходные данные передать в нее с помощью параметров, также вернуть результат. При необходимости выделить дополнительные процедуры/функции. Описать необходимые типы данных.

Задача:

Дан адрес первого элемента линейного списка. Проверить, что в заданном списке, хранящем записи вида (ФИО, Оценка по физике, Оценка по математике, Оценка по программированию), есть хотя бы один отличник (5, 5 и 5), и, если есть, то подсчитать средние баллы по каждому из предметов.

Процедура проведения

Экзаменуемый получает экзаменационный билет с одним теоретическим вопросом и одной задачей. За 60 минут он должен кратко записать ответы на первый вопрос, приведя фрагмент кода для примера, и для решения задачи: начертить блок-схему верхнего уровня с выделенными подзадачами и написать программный код указанной функции полностью или основную функцию плюс полный код двух вспомогательных процедур/функций с параметрами и заголовки/прототипы всех остальных вспомогательных процедур/функций. Если за 60 минут студент еще не полностью готов к ответу, экзаменатор может ему дать дополнительные 30 минут для подготовки, если большая часть ответа уже написана, или немного упростить задачу, если экзаменуемый затрудняется с решением. При опросе студента экзаменатор может задать вопросы по всему курсу, если ответов по вопросам билета ему не хватило для точной оценки знаний студента.

Требования к решению задачи:

1. Входные и выходные данные передавать в процедуру и возвращать, используя ПАРАМЕТРЫ: для файлов это – имя файла, для остальных структур – адрес первого элемента структуры.
2. Заданные файлы и структуры (списки, стеки) могут быть ПУСТЫМИ – нужно предусмотреть проверку. Но, если в файле ЕСТЬ данные, то считать их корректно заданными.
3. При необходимости надо написать ВСПОМОГАТЕЛЬНЫЕ ПРОЦЕДУРЫ/ФУНКЦИИ, например, добавления элемента в список/стек. Если их много, то заголовки/прототипы всех, и тело хотя бы двух из них.
4. Все описанные типы и процедуры/функции расположите В ОТДЕЛЬНОМ МОДУЛЕ (unit).
5. Если задача с матрицей, то ее надо нельзя решать на Delphi - это уже было в экзамене за 1 семестр. Только С|С++ и всю функцию main, выделив не менее трех вспомогательных функций с параметрами.

Пример заголовка/прототипа процедуры/функции на Delphi и С, где дан файл (его имя), а результат – список (адрес первого элемента списка):
Procedure Primer(const FileName: string; var ListN: PElem);

TElem* Primer(const char* FileName);

1. Перечень компетенций/индикаторов и контрольных вопросов проверки результатов освоения дисциплины

1. Компетенция/Индикатор: ИД-1_{ОПК-8} Демонстрирует знание основных языков программирования и работы с базами данных, операционных систем и оболочек, современных программных сред разработки информационных систем и технологий

Вопросы, задания

- 1. Указатели в Паскале.** Типизированные и бестиповые указатели: описание переменных, выделение и освобождение памяти. Операции \wedge и $@$. Использование указателей в качестве параметров процедур. Напишите программный код: создать указатель на запись, состоящую из двух полей: фамилия и возраст, присвойте значения полям записи, выведите эти значения в стандартный или нестандартный текстовый файл.
- 2. Язык Си.** Управляющая структура языка C – «ветвление»: одинарный и множественный выбор. Опишите синтаксис и изобразите блок-схемы, сравните с аналогами в языке Паскаль. Операция $?$: и операции отношения (сравнения)
- 3. Записи в Паскале, структуры и объединения в Си:** сходство и различия, синтаксис описания типов и переменных.
- 4. Массивы в Си.** Динамические и статические одно- и двух- мерные массивы. Расположение элементов массивов в памяти. Выделение и освобождение памяти. Двухмерные массивы с разным количеством элементов в разных строках. Описание параметров типа массив у функций.
- 5. Массивы в Паскале.** Динамические одно- и двух- мерные массивы. Расположение элементов массивов в памяти. Организация динамического массива с помощью стандартного типа – открытого массива. Выделение и освобождение памяти. Двухмерные массивы с разным количеством элементов в разных строках. Описание параметров процедур типа массив.
- 6. Массивы в Паскале.** Динамические и статические одномерные массивы. Расположение элементов массивов в памяти. Открытый массив как тип с управляемым временем жизни. Организация динамического одномерного массива с помощью стандартного типа – открытого массива - и с помощью самостоятельного распределения памяти с использованием процедур GetMem + FreeMem и работы с указателями.
- 7. Указатели в Си.** Типизированные и бестиповые указатели: описание переменных, выделение и освобождение памяти. Операции $*$ и $\&$. Использование указателей в качестве параметров функции. Напишите программный код: создать указатель на структуру, состоящую из двух полей: фамилия и возраст, присвойте значения полям структуры, выведите эти значения в стандартный или нестандартный текстовый файл.
- 8. Процедурный тип в Паскале и указатель на функцию в Си.** Использование имени функции в качестве параметра другой функции.
- 9. Строки в Си:** описание, ввод, вывод. Отличия от строк Delphi. Индекс первого символа строки. Статические строки и динамические строки: описание и выделение памяти. Описание параметров процедур строковых типов. Напишите программный код: подсчитать количество цифр в заданной строке.
- 10. Строки в Паскале.** Использование типов ShortString и ANSISString: описание, ввод, вывод. Нулевой элемент, если есть. Индекс первого символа строки. Статические строки и динамические строки с управляемым временем жизни. Описание параметров процедур строковых типов. Напишите программный код: подсчитать количество цифр в заданной строке.

Материалы для проверки остаточных знаний

1.(C|C++) Какая из нижеприведенных функций является рекурсивной? (выберите только один ответ)

```
int F1(int x){
    if (x > 0) return x*F1(x-1);
    else return 1;
}
int F2(int x){
    if (x < 0) return 1;
    else return x+F2(x-1);
}
```

Ответы:

- 1) Обе рекурсивные (Прямая рекурсия)
- 2) Обе рекурсивные (Косвенная рекурсия)
- 3) F1 – рекурсивная, F2 – нет
- 4) F2 – рекурсивная, F1 – нет
- 5) Обе не являются рекурсивными

Верный ответ: 1

2.(C|C++) Сколько раз выполниться указанный цикл? (выберите только один ответ)
float s=0; for(int i=5; i>0; s+=i, i-=2);

Ответы:

- 0
- 1
- 3
- 5

Верный ответ: 3

3.(C|C++) Сколько раз выполниться указанный цикл? (выберите только один ответ)
float s=0; for(int i=5; 0>i; s+=i, i-=2);

Ответы:

- 0
- 1
- 3
- 5

Верный ответ: 0

4.(C|C++) Какие числа будут прибавлены к значению переменной s в процессе выполнения указанного цикла? (выберите только один ответ)
for(int i=5; 10>i; i+=2, s+=i);

Ответы:

- 1)5,7,9
- 2)7,9,11
- 3)5,6,7,8,9
- 4)6,7,8,9,11

Верный ответ: 2

2. Компетенция/Индикатор: ИД-2_{ОПК-8} Применяет языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ

Вопросы, задания

1. **Типы файлов** в Си. Расскажите о текстовом файле подробнее: описание файловой переменной, открытие, чтение, запись, закрытие. Форматированный вывод. Особенности

строения и работы с текстовым файлом. Проверка ошибок при открытии файла. Напишите программный код: в текстовом файле file1.txt находятся числа по одному в строке - найти сумму первого и последнего из них.

2. Типы файлов в Паскале. Расскажите о бестиповом файле подробнее: описание файловой переменной, открытие, чтение, запись, закрытие. Особенности строения и работы с бестиповым файлом. Проверка ошибок при работе с файлом. Напишите программный код: создать копию исходного файла file1.dan

3. Типы файлов в Паскале. Расскажите о типизированном файле подробнее: описание файловой переменной, открытие, чтение, запись, закрытие. Особенности строения и работы с типизированным файлом. Проверка ошибок при работе с файлом. Напишите программный код: в типизированном файле file1.dat хранятся числа - найти сумму первого и последнего из них, и поменять их местами.

4. Типы файлов в Паскале. Расскажите о текстовом файле подробнее: описание файловой переменной, открытие, чтение, запись, закрытие. Форматированный вывод. Особенности строения и работы с текстовым файлом. Проверка ошибок при работе с файлом. Напишите программный код: в текстовом файле file1.txt находятся числа по одному в строке - найти сумму первого и последнего из них.

5. Типы файлов в Си. Расскажите о двоичном файле подробнее: описание файловой переменной, открытие, чтение, запись, закрытие. Особенности строения и работы с двоичным файлом. Проверка ошибок при открытии файла. Напишите программный код: в двоичном файле file1.dat хранятся числа типа int - найти сумму первого и последнего из них, и поменять их местами.

6. Написать процедуру/функцию на Паскале или С (задачи с массивами - только на С) для решения нижеприведенной задачи.

Исходные данные передать в нее с помощью параметров, также вернуть результат. При необходимости выделить дополнительные процедуры/функции. Описать необходимые типы данных.

1. Дан адрес первого элемента линейного списка. Проверить, что в заданном списке, хранящем

записи вида (ФИО, Оценка по физике, Оценка по математике, Оценка по программированию),

есть хотя бы один отличник (5, 5 и 5), и, если есть, то подсчитать средние баллы по каждому из предметов.

2. Даны адреса вершин двух стеков Stack1 и Stack2, хранящих сведения о цветах (Название,

Цена), и критическая цена К. Измените стеки, переложив всю информацию о цветах, дешевле К

во второй стек, а остальных – в первый.

3. Дано имя файла. Из данных вида (Наименование товара, Производитель, Цена, Количество),

хранящихся в типизированном (Delphi)/двоичном (С) файле, с заданным именем, создать упорядоченный односвязный линейный список, добавляя элементы в порядке увеличения

цены, а при равной цене – по наименованию товара в алфавитном порядке. Вернуть адрес

первого элемента созданного списка.

4. Если в прямоугольной матрице A ($N \times M$) сумма значений элементов первой строки матрицы A и сумма значений элементов последней строки матрицы A не отрицательны, определить индексы последнего из нечетных отрицательных элементов матрицы A , если такой есть.

5. Задача с самостоятельным выбором структуры данных:

Дана последовательность (на выбор студента: динамический массив (только на языке C++), список (C++, Паскаль), стек (C++, Паскаль), файл двоичный (C++) или типизированный (Паскаль)), хранящая информацию о результатах прохождения теста и состоящая из данных вида:

Имя,

Фамилия

и целочисленный балл от 0 до 100.

Проверить, все ли тестируемые смогли набрать хотя бы 50 баллов,

и, если все, то вычислить процент тех, чья фамилия начинается на ту же букву, что и у Вас, сдающего экзамен по "Программированию" за 2 семестр,

а если не все, то вычислить средний балл среди тех тестируемых, чье имя полностью совпадает с Вашим

Материалы для проверки остаточных знаний

1. СТЕК - это... (выберите только один ответ)

Ответы:

- 1) базовый тип данных
- 2) информационно-логическая линейная структура с дисциплиной доступа LIFO (Last In First Out), которая может быть смоделирована на основе базовых структур, например, на основе одномерного массива, связного линейного списка, типизированного или двоичного файла

Верный ответ: 2

2. ОЧЕРЕДЬ - это линейная последовательность элементов, с порядком обработки: (выберите только один ответ)

Ответы:

- 1) FIFO (First In First Out - Кто первым пришел, тот и обслуживается первым)
- 2) LIFO (Last In First Out - Кто последним добавлен, тот удаляется первым)

Верный ответ: 1

3. Компетенция/Индикатор: ИД-3опк-8 Применяет методы и технологии отладки и оптимизации программного обеспечения

Вопросы, задания

1. Информационно-логические структуры. Стек. Реализация своего типа «стек» на Паскале или Си на основе своего ссылочного типа «линейный список»: описание типа, допустимые операции. Напишите программный код: найти сумму всех положительных элементов стека, используя только допустимые для стека операции.

2. Информационно-логические структуры. Список. Односвязный и двусвязный линейные списки. Отличия массивов от списков: достоинства и недостатки этих структур. Реализация своего ссылочного рекурсивного типа «линейный список» на Паскале или Си: описание типа, допустимые операции. Напишите программный код: удаление всех элементов списка с освобождением памяти.

3.Рекурсия. Рекурсивный алгоритм, дерево рекурсивных вызовов, глубина вызовов и объем рекурсии. Стек рекурсивных вызовов: переполнение и способы его предотвращения. Взаимnoreкурсивные процедуры/функции. Предварительное описание процедур/функций в Паскале (forward) и прототипы в Си. Приведите пример описания взаимnoreкурсивных процедур/функций.

4.Рекурсия. Рекурсивный алгоритм, дерево рекурсивных вызовов, глубина вызовов и объем рекурсии. Линейная и каскадная рекурсии. Стек рекурсивных вызовов: переполнение и способы его предотвращения. Приведите пример линейной и каскадной рекурсии.

Материалы для проверки остаточных знаний

1.(Паскаль) Какая из нижеприведенных функций является рекурсивной? (выберите только один ответ)

```
function F1(x: integer): integer;
begin
  if x > 0 then F1:=x*(x-1) else F1:=1;
end;
function F2(x: integer): integer;
begin
  if x < 0 then F2:=1 else F2:=x+F2(x-1);
end;
```

Ответы:

- 1) F1 – рекурсивная, F2 – нет
- 2) F2 – рекурсивная, F1 – нет
- 3) обе рекурсивные
- 4) обе не рекурсивные

Верный ответ: 2

2.(Паскаль) Какая из нижеприведенных функций является рекурсивной? (выберите только один ответ)

```
function F1(x: integer): integer;
begin
  if x > 0 then F1:=x*F2(x-1) else F1:=1;
end;
function F2(x: integer): integer;
begin
  if x < 0 then F2:=1 else F2:=x+F1(x-1);
end;
```

Ответы:

- 1) Обе рекурсивные (Прямая рекурсия)
- 2) Обе рекурсивные (Косвенная рекурсия)
- 3) F1 – рекурсивная, F2 – нет
- 4) F2 – рекурсивная, F1 – нет
- 5) Обе не являются рекурсивными

Верный ответ: 2

4. Компетенция/Индикатор: ИД-2опк-9 Использует программные средства для решения практических задач

Вопросы, задания

1.Язык Си. Управляющая структура языка C – «цикл». Опишите синтаксис циклов и изобразите блок-схемы, сравните с аналогами в языке Паскаль. Напишите программный код циклического алгоритма тремя способами (с помощью трех разных циклов); найти среднее значение элементов целочисленного массива.

2. Язык Си. Описание и вызов функций. Передача параметров по значению и по ссылке. Прототипы функций, файлы заголовков и модули. Напишите функцию поиска значения минимального из элементов двумерного массива и его индексов.

Материалы для проверки остаточных знаний

1. При написании программ можно... (выберите только один ответ)

Ответы:

- 1) использовать только стандартные базовые типы данных
- 2) можно описывать и моделировать собственные типы данных

Верный ответ: 2

II. Описание шкалы оценивания

Оценка: 5 («отлично»)

Описание характеристики выполнения знания: оценки «отлично» заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание материалов изученной дисциплины, умение свободно выполнять задания, предусмотренные программой, усвоивший основную и знакомый с дополнительной литературой, рекомендованной программой. Как правило, оценка «отлично» выставляется студенту, проявившему творческие способности в понимании, изложении и использовании материалов изученной дисциплины, безупречно ответившему не только на вопросы билета, но и на дополнительные вопросы в рамках основной программы дисциплины экзамена, правильно выполнившему практическое задание

Оценка: 4 («хорошо»)

Описание характеристики выполнения знания: оценки «хорошо» заслуживает студент, обнаруживший полное знание материала изученной дисциплины, успешно выполняющий предусмотренные в программе задания, усвоивший основную литературу, рекомендованную в программе. Как правило, оценка «хорошо» выставляется студенту, показавшему систематический характер знаний по дисциплине, ответившему на все вопросы билета, правильно выполнившему практическое задание, но допустившему при этом принципиальные ошибки

Оценка: 3 («удовлетворительно»)

Описание характеристики выполнения знания: оценки «удовлетворительно» заслуживает студент, обнаруживший знание материала изученной дисциплины в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справляющийся с выполнением заданий, предусмотренных программой, знакомый с основной литературой, рекомендованной программой. Как правило, оценка «удовлетворительно» выставляется студентам, допустившим погрешность в ответе на теоретические вопросы и/или при выполнении практических заданий, но обладающим необходимыми знаниями для их устранения под руководством преподавателя, либо неправильно выполнившему практическое задание, но по указанию экзаменатора выполнившим другие практические задания из того же раздела дисциплины, знающий базовые типы и управляющие структуры обоих языков (Паскаль и Си) и базовые алгоритмы обработки разных структур данных

Оценка: 2 («неудовлетворительно»)

Описание характеристики выполнения знания: оценка «неудовлетворительно» выставляется студенту, обнаружившему серьезные пробелы в знаниях основного материала изученной дисциплины, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение или приступить к профессиональной деятельности по окончании вуза без дополнительных занятий по соответствующей дисциплине, не ответившим на вопросы билета и дополнительные вопросы, и неправильно выполнившему

практическое задание, не знающий базовые типы и управляющие структуры обоих языков (Паскаль и Си) и базовые алгоритмы обработки разных структур данных. Неправильное или правильное выполнение только практического задания не является однозначной причиной для выставления оценки «неудовлетворительно», обязательно требуется устный ответ по написанному на листах во время подготовки к устному ответу. Оценка «неудовлетворительно» выставляется также, если студент - после начала экзамена отказался его сдавать; - нарушил правила сдачи экзамена (списывал, подсказывал, обманом пытался получить более высокую оценку и т.д.).

III. Правила выставления итоговой оценки по курсу

Оценка определяется в соответствии с Положением о балльно-рейтинговой системе для студентов НИУ «МЭИ» на основании семестровой и экзаменационной составляющих. В приложение к диплому выносится оценка за 2 семестр.