

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКАЯ ГОСУДАРСТВЕННАЯ АКАДЕМИЯ»

«Утверждаю»
Проректор по учебной работе
«30» 03 2022


Г.Ю. Нагорная



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Высокоуровневые методы информатики и программирования

Уровень образовательной программы бакалавриат

Направление подготовки 09.03.03 Прикладная информатика

Направленность (профиль) Прикладная информатика в экономике

Форма обучения очная

Срок освоения ОП 4 года

Институт Прикладной математики и информационных технологий

Кафедра разработчик РПД Прикладная информатика

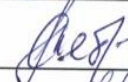
Выпускающая кафедра Прикладная информатика

Начальник
учебно-методического управления




Семенова Л. У.

Директор института ПМиИТ



Тебуев Д. Б.

Заведующий выпускающей кафедрой



Хапаева Л. Х.

г. Черкесск, 2022 г.

СОДЕРЖАНИЕ

1. Цели освоения дисциплины.....	4
2. Место дисциплины в структуре образовательной программы.....	4
3. Планируемые результаты обучения по дисциплине	5
4. Структура и содержание дисциплины.....	6
4.1. Объем дисциплины и виды учебной работы.....	6
4.2. Содержание дисциплины	7
4.2.1. Разделы (темы) дисциплины, виды учебной деятельности и формы контроля.....	7
4.2.2. Лекционный курс	8
4.2.3. Лабораторный практикум	14
4.2.4. Практические занятия	15
4.3. Самостоятельная работа обучающегося	20
5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине	23
6. Образовательные технологии.....	27
7. Учебно - методическое и информационное обеспечение дисциплины	28
7.1. Перечень основной и дополнительной учебной литературы.....	29
7.2. Перечень ресурсов информационно - телекоммуникационной сети «Интернет».....	29
7.3. Информационные технологии	29
8. Материально-техническое обеспечение дисциплины	29
8.1. Требования к аудиториям (помещениям, местам) для проведения занятий...	29
8.2. Требования к оборудованию рабочих мест преподавателя и обучающихся...	29
8.3. Требования к специализированному оборудованию.....	29
9. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья.....	29
Приложение 1. Фонд оценочных средств.....	30
Приложение 2. Аннотация рабочей программы.....	74
Рецензия на рабочую программу.....	75
Лист переутверждения рабочей программы дисциплины.....	76

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целями освоения дисциплины «Высокоуровневые методы информатики и программирования» являются:

- ознакомление обучающихся с современными технологиями программирования;
- изучение методов структурного, модульного и объектно-ориентированного программирования;
- изучение и применение высокоуровневых методов программирования в практической деятельности.

Задачами изучения дисциплины являются:

- выработка оценки современного состояния и перспективных направлений развития технологий программирования;
- формирование навыков создания программных продуктов с использованием современных сред разработки;
- приобретение практических навыков разработки, отладки и тестирования программ решения прикладных задач по обработке информации с использованием современного интерфейсного оформления;
- изучение способов доступа к базам данных.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

2.1. Дисциплина «Высокоуровневые методы информатики и программирования» относится к дисциплинам по выбору вариативной части, формируемой участниками образовательных отношений, имеет тесную связь с другими дисциплинами.

2.2. В таблице приведены предшествующие и последующие дисциплины, направленные на формирование компетенций дисциплины в соответствии с матрицей компетенций ОП.

Предшествующие и последующие дисциплины, направленные на формирование компетенций

№ п/п	Предшествующие дисциплины	Последующие дисциплины
1	Информатика и программирование Встроенные языки программирования для офисных приложений	Разработка приложений для мобильных платформ Технологии сетевого программирования

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Планируемые результаты освоения образовательной программы (ОП) – компетенции обучающихся определяются требованиями стандарта по направлению подготовки 09.03.03 Прикладная информатика и формируются в соответствии с матрицей компетенций ОП

№ п/п	Номер/ индекс компетенции	Наименование компетенции (или ее части)	В результате изучения дисциплины обучающиеся должны:
1	2	3	4
1.	ПК-3	Способен программировать приложения и создавать программные прототипы решения прикладных задач	<p>ПК-3.1. Проводит оценку предложенных решений по реализации интерфейсов форматов обмена данными</p> <p>ПК-3.2. Демонстрирует знания о современных программных средствах и платформах инфраструктуры информационных технологий организации</p> <p>ПК-3.4. Формирует знания в области компьютерной графики и использует знания о современных языках программирования в профессиональной деятельности</p>
2.	ПК-4	Способен проектировать информационные системы в соответствии с профилем подготовки по видам обеспечения	<p>ПК-4.1. Демонстрирует знания основ современных систем управления базами данных и умения разрабатывать структуру базы данных информационных систем в соответствии с архитектурной спецификацией и требованиями заказчика к информационной системе</p> <p>ПК-4.2. Выявляет первоначальные требования заказчика к информационной системе, определяет возможности достижения соответствия информационной системы первоначальным требованиям заказчика</p> <p>ПК-4.4. Применяет методы решения практических задач на основе готовых пакетов прикладных программ, используя объектно-ориентированное программирование в предметной области.</p>

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Вид учебной работы	Всего часов	Семестр	
		№ 5	
		часов	
1	2	3	
Аудиторная контактная работа (всего)	72	72	
В том числе:			
Лекции (Л)	36	36	
Лабораторные работы (ЛР)	36	36	
Контактная внеаудиторная работа, в том числе:	2	2	
Групповые и индивидуальные консультации	2	2	
Самостоятельная работа обучающегося (СРО) (всего)	106	106	
Работа с лекционным материалом, поиск и обзор литературы и электронных источников информации по индивидуально заданной проблеме курса	42	42	
Выполнение и подготовка к защите лабораторной и контрольной работам	20	20	
Работа с электронным портфолио	18	18	
Подготовка к тестированию	18	18	
Подготовка к промежуточному контролю (ППК)	8	8	
Промежуточная аттестация	Экзамен(Э)	Э (36)	Э (36)
	Прием экз., час.	0,5	0,5
	Консультация, час.	2	2
	СРО, час.	33,5	33,5
ИТОГО: Общая трудоемкость	часов	216	216
	зач. ед.	6	6

4.2. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.2.1. Разделы (темы) дисциплины, виды учебной деятельности и формы контроля

№ п/п	№ семестра	Наименование раздела дисциплины	Виды учебной деятельности, включая самостоятельную работу обучающихся (в часах)					Формы текущей и промежуточной аттестации
			Л	ЛР	ПЗ	СРО	всего	
1	2	3	4	5	6	7	8	9
1.	5	Раздел 1. Технологии программирования	6	6		20	32	Отчет по лабораторной работе, контрольные вопросы, тестовый контроль, контрольная работа
2.		Раздел 2. Объектно-ориентированный подход к проектированию и разработке программ	8	4		28	42	Отчет по лабораторной работе, контрольные вопросы, тестовый контроль, контрольная работа
3.		Раздел 3. Основы визуального программирования	10	14		28	52	Отчет по лабораторной работе, контрольные вопросы, тестовый контроль, контрольная работа
4.		Раздел 4. Основы работы с базами данных	12	12		30	52	Отчет по лабораторной работе, контрольные вопросы, тестовый контроль, контрольная работа

5.		Контактная внеаудиторная работа					2	Индивидуальн ые и групповые консультации
6.		Промежуточная аттестация					36	Экзамен
Итого часов в 3 семестре:			36	36		106	216	
Всего:			36	36		106	216	

4.2.2. Лекционный курс

№ п/п	Наименование раздела дисциплины	Наименование темы лекции	Содержание лекции	Всего часов
1	2	3	4	5
Семестр 5				
1.	Раздел 1. Технологии программирования	Тема 1.1 Основные понятия программирования	Понятие программы, программирования, программного продукта. Понятие языка программирования. Классификация языков программирования. Этапы создания программных продуктов. Основные направления в области создания технологий программирования Этапы решения задач на ЭВМ. Критерии качества программ, жизненный цикл программы. Разработка алгоритмов, основные алгоритмические структуры. Программирование на языке высокого уровня	2
2.		Тема 1.2 Программирование в средах современных информационных систем.	Структурное программирование Элементы теории модульного программирования. Структура модуля на языке Паскаль Организация внешних подпрограмм на языке Паскаль. Объектно-ориентированное программирование	4
3.	Раздел 2. Объектно-ориентированный подход к проектированию и разработке программ	Тема 2.1. Объектно-ориентированный подход к проектированию и разработке программ	Основные понятия объектно-ориентированного программирования. Объектный тип данных; переменные объектного типа; классы и объекты. Понятие объекта.	4

			Принципы ООП–инкапсуляция; наследование; полиморфизм.	
4.		Тема 2.1. Объекта на языке Турбо Паскаль.	Структура объекта на языке Турбо Паскаль. Конструкторы, деструкторы.	4
5.	Раздел 3. Основы визуального программирования	Тема 3.1 Интегрированная среда разработки LAZARUS. Основные компоненты.	Интегрированная среда разработки LAZARUS. Элементы среды разработки. Окно проекта. Окно свойств. Окно редактирования кода. Окно редактирования форм. Окна отладочной информации. Типы данных LAZARUS. Функции преобразования данных в Lazarus.	4
6		Тема 3.2 Основные компоненты LAZARUS.	Компоненты для ввода, отображения, редактирования и вывода информации. Работа с компонентами выбора из списка ListBox, CheckBox, ComboBox. Работа с массивами в Lazarus, компоненты Memo, StringGrid.	4
7.		Тема 3.3 Работа с файлами в LAZARUS	Понятие файла. Описание данных файлового типа Основные функции работы с файлами Текстовые файлы. Функции EOF, EOLN	2
8.	Раздел 4. Основы работы с базами данных	Тема 4.1 Разработка клиент-серверных приложений с использованием СУБД Firebird	Программа администрирования баз данных IB EXPERT IB EXPERT – создание и регистрация базы данных IB EXPERT – разработка структуры таблиц базы данных IB EXPERT – проектирование связей между таблицами базы данных	4

			Типы данных в Firebird	
9.		Тема 4.2. Создание пользовательского интерфейса клиент-серверных приложений в LAZARUS	Визуальное проектирование клиентских приложений в Lazarus.. Lazarus – подключение к базе данных Firebird Lazarus – компоненты доступа и отображения данных Lazarus – свойства компонента TIBConnection Lazarus – свойства компонентов TSQLQuery и TSQLTransac tion Компоненты отображения данных DBGrid, DBEdit, DBNavigatorи их свойства	4
10		Тема 4.3 Основы языка SQL-	Создание и удаление базы данных. Создание и удаление таблиц. Извлечение данных из таблиц. Выбор данных из нескольких таблиц. Вложенные запросы. Ввод, удаление и изменение значений полей.	4
ИТОГО часов в семестре:				36

4.2.3. Лабораторный практикум

№ п/п	Наименование раздела дисциплины	Наименование лабораторных работ	Всего часов
1.	2	3	4
1.	Раздел 1. Технологии программирования	Структурное программирование	2
2.		Внешние подпрограммы и модули на языке Турбо Паскаль	4
3.	Раздел 2. Объектно- ориентированный подход к проектированию и разработке	Объекты в Турбо Паскале	4

	программ		
4.	Раздел 3. Основы визуального программирования	Компоненты для ввода, отображения, редактирования и вывода информации.	2
5.		Работа с компонентами выбора из списка ListBox, CheckBox, ComboBox.	4
6.		Работа с массивами в Lazarus, компоненты Memo, StringGrid.	4
7.		Работа с файлами в LAZARUS	4
8	Раздел 4. Основы работы с базами данных	Разработка клиент-серверных приложений с использованием СУБД Firebird	6
9		Создание пользовательского интерфейса клиент-серверных приложений в LAZARUS	6
	ИТОГО:		36

4.2.4. Практические занятия не предусмотрены

4.3. САМОСТОЯТЕЛЬНАЯ РАБОТА ОБУЧАЮЩЕГОСЯ

№ п/п	Наименование раздела (темы) дисциплины	№ п/п	Виды СРО	Всего часов
1	2	3	4	5
Семестр 5				
1. _____	Раздел 1. Технологии программирования	1.1	Проработка лекций - включает чтение конспекта лекций, профессиональной литературы, периодических изданий.	10
.		1.2	Проработка лекций - включает чтение конспекта лекций, профессиональной литературы, периодических изданий. Подготовка к лабораторному занятию Подготовка к тестированию	10
2. _____	Раздел 2. Объектно-ориентированный подход к проектированию и разработке программ	2.1	Проработка лекций - включает чтение конспекта лекций, профессиональной литературы, периодических изданий. Подготовка к лабораторному занятию Подготовка к тестированию	18

			Составление портфолио	
		2.2	Проработка лекций - включает чтение конспекта лекций, профессиональной литературы, периодических изданий. Подготовка к лабораторному занятию Подготовка к тестированию	10
3. _____	Раздел 3. Основы визуального программирования	3.1	Проработка лекций - включает чтение конспекта лекций, профессиональной литературы, периодических изданий. Подготовка к лабораторному и практическому занятию.	10
		3.2	Проработка лекций - включает чтение конспекта лекций, профессиональной литературы, периодических изданий. Подготовка к лабораторному и практическому занятию. Составление портфолио	10
		3.3	Проработка лекций - включает чтение конспекта лекций, профессиональной литературы, периодических изданий. Подготовка к лабораторному занятию Подготовка к тестированию	8
4. _____	Раздел 4. Основы работы с базами данных	4.1	Проработка лекций - включает чтение конспекта лекций, профессиональной литературы, периодических изданий. Подготовка к лабораторному и практическому занятию.	10
		4.2	Проработка лекций - включает чтение конспекта лекций, профессиональной литературы, периодических изданий. Подготовка к лабораторному и практическому занятию. Составление портфолио	10

		4.3	Проработка лекций - включает чтение конспекта лекций, профессиональной литературы, периодических изданий. Подготовка к лабораторному занятию Подготовка к тестированию Составление портфолио	10
ИТОГО часов в семестре:				106

5. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

5.1. Методические указания для подготовки обучающихся к лекционным занятиям

На лекциях рекомендуется деятельность обучающегося в форме активного слушания, т.е. предполагается возможность задавать вопросы на уточнение понимания темы и рекомендуется конспектирование основных положений лекции. Основная дидактическая цель лекции — обеспечение ориентировочной основы для дальнейшего усвоения учебного материала.

В ходе лекционных занятий необходимо вести конспектирование учебного материала. Обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации. После лекции необходимо доработать свой конспект лекции, делая в нем соответствующие записи из литературы, рекомендованной преподавателем и предусмотренной рабочей программой.

Специфической чертой изучения данного курса является то, что приобретение умений и навыков работы невозможно без систематической тренировки, которая осуществляется на практических занятиях. Консультации проводятся с целью оказания помощи обучающимся в изучении учебного материала, подготовки их к практическим занятиям.

5.2. Методические указания для подготовки обучающихся к лабораторным занятиям

Особое место в структуре дисциплины занимает лабораторный практикум, выполняемый как во время лабораторных занятий в компьютерном классе, так и в свободное от аудиторных занятий время.

Лабораторные работы имеют своей целью научить обучающихся основам визуального программирования в интерактивной среде LAZARUS, разрабатывать базы данных и приложения доступа к данным.

В процессе подготовки и проведения лабораторных занятий обучающиеся закрепляют полученные ранее теоретические знания, приобретают навыки их практического применения, опыт рациональной организации учебной работы, готовятся к сдаче экзамена.

При выполнении лабораторных работ используются соответствующие учебно-методические пособия (в них приводятся задания по лабораторным работам, методические указания по их выполнению, справочный материал с примерами решений). По каждой лабораторной работе оформляется отчет, на основании которого проводится защита работы (цель – оценка уровня освоения учебного материала). Результаты

лабораторных работ учитываются при промежуточной и итоговой аттестации по дисциплине.

Для успешного освоения материала необходимо понимание задачи, которая должна решаться компьютером в соответствии с предложенным методом – следует четко представлять, какие данные являются исходными и какие результаты должны получаться при решении задачи. Систематическое освоение необходимого учебного материала позволяет быть готовым для тестирования и выполнения индивидуальных заданий.

В начале семестра обучающиеся получают сводную информацию о формах проведения занятий и формах контроля знаний. Тогда же обучающимся предоставляется список тем лекционных и лабораторных занятий. Каждое лабораторное занятие по соответствующей тематике теоретического курса состоит из вопросов для подготовки, на основе которых проводится устный опрос каждого обучающегося. Также после изучения каждого раздела обучающиеся для закрепления проеденного материала решают тесты.

Поскольку активность обучающегося на практических занятиях является предметом внутрисеместрового контроля его продвижения в освоении курса, подготовка к таким занятиям требует от обучающегося ответственного отношения.

При подготовке к занятию обучающиеся в первую очередь должны использовать материал лекций и соответствующих литературных источников. Самоконтроль качества подготовки к каждому занятию обучающиеся осуществляют, проверяя свои знания и отвечая на вопросы для самопроверки по соответствующей теме.

Входной контроль осуществляется преподавателем в виде проверки и актуализации знаний обучающихся по соответствующей теме.

Выходной контроль осуществляется преподавателем проверкой качества и полноты выполнения задания.

Типовой план лабораторных занятий:

1. Изложение преподавателем темы занятия, его целей и задач.
2. Выдача преподавателем задания обучающимся, необходимые пояснения.
3. Выполнение задания обучающимися на компьютере. Оформление отчета по лабораторной работе.
4. Защита обучающимися лабораторной работе.

Дидактические цели лабораторного занятия: углубление, систематизация и закрепление знаний, превращение их в убеждения; проверка знаний; привитие умений и навыков самостоятельной работы с книгой; развитие культуры речи, формирование умения аргументировано отстаивать свою точку зрения, отвечать на вопросы слушателей; умение слушать других, задавать вопросы.

Задачи: стимулировать регулярное изучение программного материала, первоисточников; закреплять знания, полученные на уроке и во время самостоятельной работы; обогащать знаниями благодаря выступлениям товарищей и учителя на занятии, корректировать ранее полученные знания.

Функции лабораторного занятия:

- учебная (углубление, конкретизация, систематизацию знаний, усвоенных во время занятий и в процессе самостоятельной подготовки);
- развивающая (развитие логического мышления учащихся, приобретение ими умений работать с различными литературными источниками, формирование умений и навыков анализа фактов, явлений, проблем и т.д.);
- воспитательная (воспитание ответственности, работоспособности, воспитание культуры общения и мышления, привитие интереса к изучению предмета, формирование потребности рационализации и учебно-познавательной деятельности и организации досуга);

- диагностическая -коррекционную и контролирующую (контроль за качеством усвоения обучающимися учебного материала, выявление пробелов в его усвоении и их преодоления).

Организация подготовки лабораторного занятия

1. Сообщить тему и план.
2. Предложить для самостоятельного изучения основную и дополнительную литературу.

5.3 Методические указания для подготовки обучающихся к практическим занятиям

Одной из важных форм самостоятельной работы является подготовка к практическому занятию.

Основная цель проведения практического занятия заключается в закреплении знаний полученных в ходе прослушивания лекционного материала.

Практическое занятие проводится в форме устного опроса обучающихся по теме занятия а также в виде решения практических задач или моделирования практической ситуации.

В ходе подготовки к практическому занятию обучающемуся следует просмотреть материалы лекции, а затем начать изучение учебной литературы. Следует знать, что освещение того или иного вопроса в литературе часто является личным мнением автора, построенного на анализе различных источников, поэтому следует не ограничиваться одним учебником, а рассмотреть как можно больше материала по интересующей теме.

При подготовке к практическим занятиям обучающийся должен придерживаться следующей технологии:

1. внимательно изучить основные вопросы темы и план практического занятия, определить место темы занятия в общем содержании, ее связь с другими темами;
2. найти и проработать соответствующие разделы в рекомендованных нормативных документах, учебниках и дополнительной литературе;
3. после ознакомления с теоретическим материалом ответить на вопросы для самопроверки;
4. продумать свое понимание сложившейся ситуации в изучаемой сфере, пути и способы решения проблемных вопросов;
5. продумать развернутые ответы на предложенные вопросы темы, опираясь на лекционные материалы, расширяя и дополняя их данными из учебников, дополнительной литературы.

5.4 Методические указания по самостоятельной работе обучающихся

Самостоятельная работа обучающегося предполагает различные формы индивидуальной учебной деятельности: конспектирование научной литературы, сбор и анализ практического материала в СМИ, проектирование, выполнение тематических и творческих заданий и пр. Выбор форм и видов самостоятельной работы определяется индивидуально-личностным подходом к обучению совместно преподавателем и обучающимся. Формы текущего контроля успеваемости и промежуточной аттестации обучающихся.

Содержание внеаудиторной самостоятельной работы обучающихся по дисциплине «Высокоуровневые методы информатики и программирования» включает в себя различные виды деятельности:

- чтение текста (учебника, первоисточника, дополнительной литературы);
- составление плана текста;
- конспектирование текста;
- исследовательская работа;

- работа с электронными информационными ресурсами;
- выполнение тестовых заданий;
- ответы на контрольные вопросы;
- аннотирование, реферирование, рецензирование текста;
- составление глоссария или библиографии по конкретной теме;
- выполнение лабораторных заданий.

Работа с литературными источниками и интернет ресурсами

В процессе подготовки к практическим занятиям, обучающимся необходимо обратить особое внимание на самостоятельное изучение рекомендованной учебно-методической (а также научной и популярной) литературы.

Самостоятельная работа с учебниками, учебными пособиями, научной, справочной и популярной литературой, материалами периодических изданий и Интернета, статистическими данными является наиболее эффективным методом получения знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у обучающихся свое отношение к конкретной проблеме.

Более глубокому раскрытию вопросов способствует знакомство с дополнительной литературой, рекомендованной преподавателем по каждой теме семинарского или практического занятия, что позволяет обучающимся проявить свою индивидуальность в рамках выступления на данных занятиях, выявить широкий спектр мнений по изучаемой проблеме.

Электронное портфолио – совокупность студенческих работ, собранных с применением электронных средств и носителей.

Портфолио составляется в таком ключе, чтобы обеспечить эффективное взаимодействие с научными руководителями, преподавателями и кураторами в вузе в период обучения.

Все виды работ обучающегося по данному курсу после оценивания преподавателем должны выкладываться в электронное портфолио.

Методические указания по подготовке к устному опросу

Целью устного собеседования являются обобщение и закрепление изученного курса. Обучающимся предлагаются для освещения сквозные концептуальные проблемы. При подготовке следует использовать лекционный материал и учебную литературу. Для более глубокого постижения курса и более основательной подготовки рекомендуется познакомиться с указанной дополнительной литературой. Готовясь к семинару, обучающийся должен, прежде всего, ознакомиться с общим планом семинарского занятия. Следует внимательно прочесть свой конспект лекции по изучаемой теме и рекомендуемую к теме семинара литературу. При этом важно научиться выделять в рассматриваемой проблеме самое главное и сосредотачивать на нем основное внимание при подготовке. С незнакомыми терминами и понятиями следует ознакомиться в предлагаемом глоссарии, словаре или энциклопедии.

Ответ на каждый вопрос из плана семинарского занятия должен быть доказательным и аргументированным, обучающемуся нужно уметь отстаивать свою точку зрения. Для этого следует использовать документы, монографическую, учебную и справочную литературу. Активно участвуя в обсуждении проблем на семинарах обучающиеся учатся последовательно мыслить, логически рассуждать, внимательно слушать своих товарищей, принимать участие в спорах и дискуссиях. Для

успешной подготовки к устному опросу, обучающийся должен законспектировать рекомендуемую литературу, внимательно осмыслить фактический материал и сделать выводы. Обучающемуся надлежит хорошо подготовиться, чтобы иметь возможность грамотно и полно ответить на заданные ему вопросы, суметь сделать выводы и показать значимость данной проблемы для изучаемого курса. Обучающемуся необходимо также дать анализ той литературы, которой он воспользовался при подготовке к устному опросу на семинарском занятии.

При подготовке, обучающийся должен правильно оценить вопрос, который он взял для выступления к семинарскому занятию. Но для того чтобы правильно и четко ответить на поставленный вопрос, необходимо правильно уметь пользоваться учебной и дополнительной литературой.

Перечень требований к любому выступлению обучающегося примерно таков:
связь выступления с предшествующей темой или вопросом.

раскрытие сущности проблемы.

методологическое значение для научной, профессиональной и практической деятельности.

Разумеется, обучающийся не обязан строго придерживаться такого порядка изложения, но все аспекты вопроса должны быть освещены, что обеспечит выступлению необходимую полноту и завершенность.

Приводимые участником семинара примеры и факты должны быть существенными, по возможности перекликаться с профилем обучения.

Выступление обучающегося должно соответствовать требованиям логики. Четкое вычленение излагаемой проблемы, ее точная формулировка, неукоснительная последовательность аргументации именно данной проблемы, без неоправданных отступлений от нее в процессе обоснования, безусловная доказательность, непротиворечивость и полнота аргументации, правильное и содержательное использование понятий и терминов.

Методические рекомендации прохождения тестирования

Подготовку к итоговому тестированию необходимо осуществлять поэтапно.

На первом этапе необходимо повторить основные положения всех тем, детально разбирая наиболее сложные моменты. Непонятные вопросы необходимо выписывать, чтобы по ним можно было проконсультироваться с преподавателем перед прохождением итогового тестирования. Подготовку по темам каждой дидактической единицы целесообразно производить отдельно. На этом этапе необходимо использовать материалы лекционного курса, материалы семинарских занятий, тестовые задания для текущего контроля знаний, а также презентации лекционного курса.

На втором этапе подготовки предлагается без повторения теоретического материала дать ответы тестовые задания для рубежного контроля знаний. Если ответы на какие-то вопросы вызвали затруднение, необходимо еще раз повторить соответствующий теоретический материал.

Наконец, третий этап подготовки необходимо осуществить непосредственно накануне теста. На данном этапе необходимо аккуратно просмотреть весь лекционный курс.

В случае, если результаты выполнения тестового задания оказались неудовлетворительными, необходимо зафиксировать темы, на вопросы по которым были даны неверные ответы, и еще раз углубленно повторить соответствующие темы в соответствии с указанными выше тремя этапами подготовки к тестированию.

Методические указания к выполнению контрольной работы

Контрольной работе как одной из форм самостоятельной учебно-исследовательской работы отводится особая роль при формировании компетенции будущего специалиста и бакалавра. Здесь обучающийся демонстрирует применение полученных знаний для создания приложений, решающих конкретные поставленные перед ним задачи. Обучающийся предъявляет преподавателю несколько версий программ, как правило, в электронном виде и получает от преподавателя положительное заключение о результате, либо замечания и предложения по корректировке программы. Программа должна предъявляться в виде, допускающем быстрый переход к ее компиляции, т.е. не допускается передача в виде изображения. Принимаются любые варианты программы, решающие исходную задачу. Преподаватель при приеме приводящей к верному результату программы зачитывает ее как исполненную, но может дать рекомендации по ее улучшению.

Методические рекомендации по подготовке к экзамену

Экзамен – это форма итоговой отчетности студента по изученной дисциплине. По решению кафедры экзамен может проводиться в нескольких формах – устной по билетам. Главная задача проведения экзамена – проверка знаний, навыков и умений студента, по прослушанной дисциплине.

Огромную роль в успешной подготовке к экзамену играет правильная организация подготовки к нему. Рекомендуется при подготовке к экзамену опираться на следующий план:

просмотреть программу курса, с целью выявления наиболее проблемных тем, вопросов, которые могут вызвать трудности при подготовке к экзамену.

прорешать задачу, предложенные в учебно-методическом комплексе. При этом для эффективного закрепления информации первый раз без использования учебных материалов и нормативно-правовых актов, второй раз с их использованием.

При выполнении первых двух пунктов плана студент получит возможность оценить свои знания и навыки по прослушанной дисциплине и сориентироваться при планировании объема подготовки.

темы необходимо изучать последовательно, внимательно обращая внимание на описание вопросов, которые раскрывают ее содержание. Начинать необходимо с первой темы.

после работы над первой темой необходимо ответить на контрольные вопросы к теме и решить тестовые задания к ней.

после изучения всех тем студенту рекомендуется ответить на контрольные вопросы по всему курсу.

6. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

№ п/п	№ семестра	Виды учебной работы	Образовательные технологии	Всего часов
1	2	3	4	
1	5	Лекция «Технологии программирования»	Лекция- презентация	2
2	5	Лекция «Разработка клиент-серверных приложений с использованием СУБД Firebird»	Лекция- визуализация	2
3	5	Лекция «Создание пользовательского интерфейса клиент-серверных приложений в LAZARUS»	Диалоговые технологии	2
4	5	Лабораторная работа Работа с файлами в LAZARUS	Учебно-исследовательская работа Компьютерная симуляция	2
5	5	Лабораторная работа Разработка клиент-серверных приложений с использованием СУБД Firebird	Учебно-исследовательская работа. Компьютерная симуляция	2

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

7.1. Перечень основной и дополнительной учебной литературы

Основная литература

1. Давыдова, Н. А. Программирование: учебное пособие / Н. А. Давыдова, Е. В. Боровская. — 4-е изд. — Москва: Лаборатория знаний, 2020. — 239 с. — ISBN 978-5-00101-788-2. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/6485.html>
2. Згуральская, Е. Н. Технологии программирования : учебное пособие / Е. Н. Згуральская. — Ульяновск : Ульяновский государственный технический университет, 2020. — 72 с. — ISBN 978-5-9795-1995-1. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/106123.html>
3. Логанов С.В. Объектно-ориентированные принципы разработки информационных систем : учебное пособие / Логанов С.В., Моругин С.Л.. — Москва : Ай Пи Ар Медиа, 2022. — 217 с. — ISBN 978-5-4497-1576-0. — Текст : электронный // IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/118880.html>

Список дополнительной литературы

1. Комлев Н.Ю. Объектно Ориентированное Программирование. Хорошая книга для

Хороших Людей / Комлев Н.Ю.. — Москва : СОЛОН-ПРЕСС, 2014. — 298 с. — ISBN 978-5-91359-138-8. — Текст : электронный // IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/26923.html>

2. Санников Е.В. Курс практического программирования в Delphi. Объектно-ориентированное программирование / Санников Е.В.. — Москва : СОЛОН-ПРЕСС, 2016. — 188 с. — ISBN 978-5-91359-122-7. — Текст : электронный // IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/90323.html>

7.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

<http://window.edu.ru>- Единое окно доступа к образовательным ресурсам;
<http://fcior.edu.ru> - Федеральный центр информационно-образовательных ресурсов;
<http://elibrary.ru> - Научная электронная библиотека.

7.3. Информационные технологии, лицензионное программное обеспечение,

В компьютерном классе должны быть установлены средства:

Лицензионное программное обеспечение	Реквизиты лицензий/ договоров
Microsoft Azure Dev Tools for Teaching 1. Windows 7, 8, 8.1, 10 2. Visual Studio 2008, 2010, 2013, 2019 5. Visio 2007, 2010, 2013 6. Project 2008, 2010, 2013 7. Access 2007, 2010, 2013 и т. д.	Идентификатор подписчика: 1203743421 Срок действия: 30.06.2022 (продление подписки)
MS Office 2003, 2007, 2010, 2013	Сведения об Open Office: 63143487, 63321452, 64026734, 6416302, 64344172, 64394739, 64468661, 64489816, 64537893, 64563149, 64990070, 65615073 Лицензия бессрочная
Антивирус Dr.Web Desktop Security Suite	Лицензионный сертификат Серийный № 8DVG-V96F-H8S7-NRBC Срок действия: с 20.10.2022 до 22.10.2023
Консультант Плюс	Договор № 272-186/С-23-01 от 20.12.2022 г.
Цифровой образовательный ресурс IPRsmart	Лицензионный договор № 10423/23П от 30.06.2023 г. Срок действия: с 01.07.2023 г. до 01.07.2024г.
Свободное программное обеспечение:	
UML – унифицированный язык моделирования	
CASE-средство BPwin.	

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

8.1. Требования к аудиториям (помещениям, местам) для проведения занятий

Учебная аудитория для проведения занятий лекционного типа

Специализированная мебель:

Кафедра настольная - 1шт., парты - 15шт., стулья - 40шт., доска - 2шт., стол преподавательский - 1шт., шкаф книжный с полками - 1шт., шкаф двухдверный - 1шт., лаб. стол - 1шт.

Набор демонстрационного оборудования и учебно-наглядных пособий, обеспечивающих тематические иллюстрации:

Настенный экран – 1 шт.

Проектор – 1 шт.

Ноутбук – 1 шт.

Лаборатория современных экономических информационных систем

Специализированная мебель:

Парты - 6шт., доска меловая - 1шт., компьютерные столы - 7шт., стол преподавательский - 3шт., стулья - 28 шт., стол лабораторный -3 шт.

Лабораторное оборудование, технические средства обучения, служащие для предоставления учебной информации большой аудитории:

Компьютер в сборе - 7 шт.

Учебная аудитория для проведения занятий семинарского типа, курсового проектирования (выполнение курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации

Специализированная мебель:

Парты - 6шт., доска меловая - 1шт., компьютерные столы - 7шт., стол преподавательский - 3шт., стулья - 28 шт., стол лабораторный -3 шт.

Технические средства обучения, служащие для предоставления учебной информации большой аудитории:

Компьютер в сборе - 7 шт.

8.2. Требования к оборудованию рабочих мест преподавателя и обучающихся

1. рабочее место преподавателя, оснащенное компьютером с доступом в Интернет,
2. рабочие места обучающихся, оснащенные компьютерами с доступом в Интернет, предназначенные для работы в электронной образовательной среде, *и т.п.*

8.3. Требования к специализированному оборудованию

9. ОСОБЕННОСТИ РЕАЛИЗАЦИИ ДИСЦИПЛИНЫ ДЛЯ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Для обеспечения образования инвалидов и обучающихся с ограниченными возможностями здоровья разрабатывается (в случае необходимости) адаптированная образовательная программа, индивидуальный учебный план с учетом особенностей их психофизического развития и состояния здоровья, в частности применяется индивидуальный подход к освоению дисциплины, индивидуальные задания: рефераты, письменные работы и, наоборот, только устные ответы и диалоги, индивидуальные консультации, использование диктофона и других записывающих средств для воспроизведения лекционного и семинарского материала.

В целях обеспечения обучающихся инвалидов и лиц с ограниченными возможностями здоровья комплектуется фонд основной учебной литературой, адаптированной к ограничению электронных образовательных ресурсов, доступ к которым организован в БИЦ Академии. В библиотеке проводятся индивидуальные консультации для данной категории пользователей, оказывается помощь в регистрации и использовании сетевых и локальных электронных образовательных ресурсов, предоставляются места в читальном зале.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ПО ДИСЦИПЛИНЕ Высокоуровневые методы информатики и программирования

ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ
Разработка программных приложений в юридических информационных системах
1. Компетенции, формируемые в процессе изучения дисциплины

Индекс	Формулировка компетенции
ПК-3	Способен программировать приложения и создавать программные прототипы решения прикладных задач
ПК-4	Способен проектировать информационные системы в соответствии с профилем подготовки по видам обеспечения

2. Этапы формирования компетенции в процессе освоения дисциплины

Основными этапами формирования указанных компетенций при изучении с дисциплины являются последовательное изучение содержательно связанных между собой разделов (тем) учебных занятий. Изучение каждого раздела (темы) предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций обучающимися.

Этапность формирования компетенций прямо связана с местом дисциплины в образовательной программе.

Разделы (темы) дисциплины	Формируемые компетенции (коды)	
	ПК-3	ПК-4
Раздел 1. Технологии программирования	+	+
Раздел 2. Объектно-ориентированный подход к проектированию и разработке программ	+	+
Раздел 3. Основы визуального программирования		+
Раздел 4. Основы работы с базами данных	+	

3. Показатели, критерии и средства оценивания компетенций, формируемых в процессе изучения дисциплины
ПК-3 Способен программировать приложения и создавать программные прототипы решения прикладных задач

Индикаторы достижения компетенций	Критерии оценивания результатов обучения				Средства оценивания результатов обучения	
	неудовлетв	удовлетв	хорошо	отлично	Текущий контроль	Промежуточная аттестация
ПК-3.1. Проводит оценку предложенных решений по реализации интерфейсов форматов обмена данными	Не может проводить оценку предложенных решений по реализации интерфейсов форматов обмена данными	Частично знает оценку предложенных решений по реализации интерфейсов форматов обмена данными	Демонстрирует знания в проведении оценки предложенных решений по реализации интерфейсов форматов обмена данными	Раскрывает полное содержание оценки предложенных решений по реализации интерфейсов форматов обмена данными	Отчет по лабораторной работе, контрольные вопросы, тестовый контроль, контрольная работа	Экзамен
ПК-3.2. Демонстрирует знания о современных программных средствах и платформах инфраструктуры информационных технологий организации	Не умеет разрабатывать современные программные средства, не знает о платформах инфраструктуры информационных технологий организации	Частично демонстрирует знания о современных программных средствах и платформах инфраструктуры информационных технологий организации	Демонстрирует знания о современных программных средствах и платформах инфраструктуры информационных технологий организации	Свободно разрабатывает современные программные средства, знает платформы инфраструктуры информационных технологий организации	Отчет по лабораторной работе, контрольные вопросы, тестовый контроль, контрольная работа	Экзамен
ПК-3.4. Формирует знания в области компьютерной графики и использует знания о современных языках программирования в профессиональной деятельности	Не владеет знаниями в области компьютерной графики и использует знания о современных языках программирования в профессиональной деятельности	Частично владеет знаниями в области компьютерной графики и использует знания о современных языках программирования в профессиональной деятельности	Владеет современными технологиями в области компьютерной графики и использует знания о современных языках программирования в профессиональной деятельности	Свободно владеет современными технологиями в области компьютерной графики и использует знания о современных языках программирования в профессиональной деятельности	Отчет по лабораторной работе, контрольные вопросы, тестовый контроль, контрольная работа	Экзамен

ПК-4 Способен проектировать информационные системы в соответствии с профилем подготовки по видам обеспечения

Индикаторы достижения компетенций	Критерии оценивания результатов обучения				Средства оценивания результатов обучения	
	неудовлетворительно	удовлетворительно	хорошо	отлично	Текущий контроль	Промежуточная аттестация
ПК-4.1. Демонстрирует знания основ современных систем управления базами данных и умения разрабатывать структуру базы данных информационных систем в соответствии с архитектурной спецификацией и требованиями заказчика к информационной системе	Отсутствуют знания основ современных систем управления базами данных и умения разрабатывать структуру базы данных информационных систем в соответствии с архитектурной спецификацией и требованиями заказчика к информационной системе	Имеются знания основ современных систем управления базами данных и умения разрабатывать структуру базы данных информационных систем в соответствии с архитектурной спецификацией и требованиями заказчика к информационной системе	Показывает знания основ современных систем управления базами данных и умения разрабатывать структуру базы данных информационных систем в соответствии с архитектурной спецификацией и требованиями заказчика к информационной системе	Демонстрирует знания основ современных систем управления базами данных и умения разрабатывать структуру базы данных информационных систем в соответствии с архитектурной спецификацией и требованиями заказчика к информационной системе	Отчет по лабораторной работе, контрольные вопросы, тестовый контроль, контрольная работа	Экзамен
ПК-4.2. Выявляет первоначальные требования заказчика к информационной системе, определяет возможности достижения соответствия информационной системы первоначальным требованиям заказчика	Не выявляет первоначальные требования заказчика к информационной системе, определяет возможности достижения соответствия информационной системы первоначальным требованиям заказчика	Выявляет первоначальные требования заказчика к информационной системе, определяет возможности достижения соответствия информационной системы первоначальным требованиям заказчика с некоторыми неточностями	Выявляет первоначальные требования заказчика к информационной системе, определяет возможности достижения соответствия информационной системы первоначальным требованиям заказчика	Успешно выявляет первоначальные требования заказчика к информационной системе, определяет возможности достижения соответствия информационной системы первоначальным требованиям заказчика	Отчет по лабораторной работе, контрольные вопросы, тестовый контроль, контрольная работа	Экзамен
ПК-4.4. Применяет методы решения практических задач на основе готовых пакетов прикладных программ, используя объектно-ориентированное программирование в предметной области	Не применяет методы решения практических задач на основе готовых пакетов прикладных программ, используя объектно-ориентированное программирование в предметной области	Применяет некоторые методы решения практических задач на основе готовых пакетов прикладных программ, используя объектно-ориентированное программирование в предметной области	Применяет методы решения практических задач на основе готовых пакетов прикладных программ, используя объектно-ориентированное программирование в предметной области	Демонстрирует владение методами решения практических задач на основе готовых пакетов прикладных программ, используя объектно-ориентированное программирование в предметной области	Отчет по лабораторной работе, контрольные вопросы, тестовый контроль, контрольная работа	Экзамен

4. Комплект контрольно-оценочных средств по дисциплине

Вопросы к экзамену

по дисциплине Высокоуровневые методы информатики и программирования

1. Понятие программы, программирования, программного продукта.
2. Понятие языка программирования. Классификация языков программирования.
3. Этапы создания программных продуктов.
4. Основные направления в области создания технологий программирования
5. Этапы решения задач на ЭВМ.
6. Критерии качества программ, жизненный цикл программы.
7. Разработка алгоритмов, основные алгоритмические структуры.
8. Программирование на языке высокого уровня
9. Структурное программирование
10. Элементы теории модульного программирования
11. Структура модуля на языке Паскаль
12. Организация внешних подпрограмм на языке Паскаль
13. Объектно-ориентированное проектирование и программирование
14. Основные понятия объектно-ориентированного программирования.
15. Сущность объектно-ориентированного подхода; объектный тип данных; переменные объектного типа; классы и объекты
16. Принципы ООП– инкапсуляция; наследование; полиморфизм.
17. Структура объекта на языке Турбо Паскаль
18. Конструкторы и деструкторы.
19. Основы визуального программирования.
20. Этапы создания приложения в средах быстрой разработки
21. Среда разработки Lazarus, система окон
22. Lazarus –основные управляющие элементы
23. Lazarus – ввод, вывод данных
24. Lazarus – функции преобразования данных
25. Lazarus – работа с компонентами ListBox, CheckBox, ComboBox
26. Lazarus – работа с массивами
27. Lazarus – компоненты Memo, StringGrid
28. Lazarus – работа с файлами
29. Описание данных файлового типа
30. Ввод и вывод текстовых файлов
31. Основные функции работы с файлами
32. Текстовые файлы. Функции EOF, EOLN
33. Программа администрирования баз данных IB EXPERT
34. IB EXPERT – создание и регистрация базы данных
35. IB EXPERT – разработка структуры таблиц
36. IB EXPERT – проектирование связей между таблицами базы данных
37. Типы данных в Firebird
38. Lazarus – подключение к базе данных Firebird
39. Lazarus – компоненты доступа и отображения данных
40. Lazarus – свойства компонента TIBConnection
41. Lazarus – свойства компонентов TSQLQuery и TSQLTransaction
42. Компоненты отображения данных DBGrid, DBEdit, DBNavigatorи их свойства
43. Firebird – основные SQL-команды
44. SQL-команда формирования запросов на выборку
45. SQL-команды добавления, удаления, изменения записей.

46. Тестирование и отладка программ. Типы ошибок в программе.
47. Ошибки времени компиляции, времени выполнения, алгоритмические.
48. Методы тестирования программ.

СЕВЕРО-КАВКАЗСКАЯ ГОСУДАРСТВЕННАЯ АКАДЕМИЯ

Кафедра Прикладной информатики

20 – 20 уч. год

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

По дисциплине «Высокоуровневые методы информатики и программирования»

Для обучающихся направления 09.03.03 «Прикладная информатика»

Вопросы:

1. Понятие программы, программирования, программного продукта
2. Lazarus – подключение к базе данных Firebird
3. Разработать проект. Найти корни квадратного уравнения $ax^2+bx+c=0$. (На Lazarus)

Зав. кафедрой

Л.Х. Хапаева

Контрольные вопросы

по дисциплине Высокоуровневые методы информатики и программирования

Вопросы к разделу 1.

1. Понятие программы, программирования, программного продукта.
2. Понятие языка программирования. Классификация языков программирования.
3. Этапы создания программных продуктов.
4. Основные направления в области создания технологий программирования
5. Этапы решения задач на ЭВМ.
6. Критерии качества программ, жизненный цикл программы.
7. Разработка алгоритмов, основные алгоритмические структуры.
8. Программирование на языке высокого уровня

Вопросы к разделу 2.

1. Структурное программирование
2. Элементы теории модульного программирования
3. Структура модуля на языке Паскаль
4. Организация внешних подпрограмм на языке Паскаль
5. Объектно-ориентированное проектирование и программирование
6. Основные понятия объектно-ориентированного программирования.
7. Сущность объектно-ориентированного подхода; объектный тип данных; переменные объектного типа; классы и объекты
8. Принципы ООП– инкапсуляция; наследование; полиморфизм.
9. Структура объекта на языке Турбо Паскаль
10. Конструкторы и деструкторы.

Вопросы к разделу 3.

1. Визуальное программирование
2. Интегрированная среда разработки LAZARUS.
3. Элементы среды разработки.
4. Окно проекта. Окно свойств.
5. Окно редактирования кода. Окно редактирования форм.
6. Типы данных LAZARUS
7. Ввод, вывод данных в Lazarus
8. Функции преобразования данных в Lazarus
9. Lazarus – работа с компонентами ListBox, CheckBox, ComboBox
10. Работа с массивами в Lazarus
11. Lazarus – компоненты Memo, StringGrid
12. Работа с файлами в Lazarus
13. Описание данных файлового типа
14. Ввод и вывод текстовых файлов
15. Основные функции работы с файлами
16. Текстовые файлы. Функции EOF, EOLN

Вопросы к разделу 4.

1. Программа администрирования баз данных IB EXPERT
2. IB EXPERT – создание и регистрация базы данных
3. IB EXPERT – разработка структуры таблиц базы данных
4. IB EXPERT – проектирование связей между таблицами базы данных
5. Типы данных в Firebird

6. Lazarus – подключение к базе данных Firebird
7. Lazarus – компоненты доступа и отображения данных
8. Lazarus – свойства компонента TIBConnection
9. Lazarus – свойства компонентов TSQLQuery и TSQLTransaction
10. Компоненты отображения данных DBGrid, DBEdit, DBNavigator и их свойства
11. Firebird – основные SQL-команды
12. SQL-команда формирования запросов на выборку
13. SQL-команды добавления, удаления, изменения записей.
14. Тестирование и отладка программ. Типы ошибок в программе.
15. Ошибки времени компиляции, времени выполнения, алгоритмические.
16. Методы тестирования программ.

Тестовые вопросы

по дисциплине Высокоуровневые методы информатики и программирования

ПК-3

1. Запись алгоритма на языке программирования, приводящая к конечному результату за конечное число шагов называется _____
2. Комплекс взаимосвязанных программ, предназначенный для реализации определенной задачи массового спроса –
 1. Программа
 2. Программный продукт
 3. Подпрограмма
 4. Программное обеспечение
3. Сфера действий, направленная на создание программ–_____
4. Язык программирования, разработанный для быстроты и удобства использования программистом. Основная черта таких языков – абстракция, то есть введение смысловых инструкций, кратко описывающих такие структуры данных и операции над ними, описания которых на машинном коде очень длинны и сложны для понимания.
 1. Язык Ассемблера
 2. Язык низкого уровня
 3. Алгоритмический язык
 4. Язык высокого уровня
5. Система правил определения поведения отдельных языковых конструкций, которая определяет смысловое значение предложений алгоритмического языка.
 1. Синтаксис
 2. Полиморфизм
 3. Орфография
 4. Семантика
6. Часть языка программирования, которая описывает структуру программ как наборов символов - _____
7. Формальные языки, специально созданные для общения человека с вычислительной машиной
 1. Языки Ассемблера
 2. Языки программирования
 3. Языки машинные
 4. Языки естественные
8. Методология разработки программного обеспечения, в основе которой лежит представление программы в виде иерархической структуры блоков - _____
9. Парадигма программирования, в которой основными концепциями являются понятия объектов и классов.
 1. Модульное программирование
 2. Объектно-ориентированное программирование
 3. Структурное программирование
 4. Процедурное программирование
10. Парадигма программирования, основанная на автоматическом доказательстве теорем. Такое программирование основано на теории и аппарате математической логики с использованием математических принципов резолюций - _____
11. Парадигма программирования, которая описывает процесс вычисления в виде инструкций, изменяющих состояние программы. Такая программа очень похожа на приказы, то есть это последовательность команд, которые должен выполнить компьютер.

1. Модульное программирование
 2. Императивная
 3. Структурное программирование
 4. Логическое программирование
12. Модульное программирование – это _____
13. Достоинство модульной программы:
1. создание программы по частям в произвольном порядке
 2. снижает количество ошибок
 3. дает всегда эффективную программу
 4. не требует компоновки
14. Укажите, в каком программировании нельзя использовать оператор GOTO
1. модульном
 2. объектно-ориентированном
 3. структурном
 4. декларативном
15. Оператор, который нельзя использовать в структурном программировании:
1. оператор цикла
 2. условный оператор
 3. оператор безусловного перехода
 4. оператор присваивания
16. Программирование, основанное на модульной структуре программного продукта и типовых управляющих структурах алгоритмов обработки называется _____
17. Декомпозиция решаемой задачи на подзадачи – более простые с точки зрения программирования части исходной задачи является важнейшим технологическим приемом
1. информационного моделирования
 2. объектно – ориентированного проектирования
 3. структурного проектирования
 4. модульного программирования
18. Последовательное разложение общей функции обработки данных на простые функциональные элементы – это _____
19. Метод построения алгоритма в котором сначала строятся вспомогательные алгоритмы а затем основной алгоритм называется ...
1. сборочный метод
 2. метод последовательной детализации
 3. метод «сверху вниз»
 4. нисходящий метод
20. При структурном программировании задача выполняется:
1. без участия программиста
 2. поэтапным разбиением на более легкие задачи
 3. объединением отдельных модулей
 4. без тестирования
21. Укажите, что не является базовой алгоритмической структурой:
1. следование
 2. ветвление
 3. безусловный переход
22. Функционально законченный фрагмент программы, оформленный в виде отдельного файла с исходным кодом или поименованной непрерывной его части, предназначенный для использования в других программах называется _____
23. Сущность модульного программирования заключается:

1. в разбиении программы на отдельные функционально независимые части
 2. в разбиении программы на функции
 3. в разбиении программы на процедуры
 4. в разбиении программы на равные части
24. Рекомендуемые размеры модулей:
1. большие
 2. **небольшие**
 3. равные
 4. фиксированной длины
25. Независимость модуля заключается в.....
1. В написании, тестировании и отладке независимо от остальных модулей
 2. В независимости от работы основной программы
 3. В разработке независимо от остальных модулей
 4. В независимости процедур в модуле друг от друга
26. Достоинствами модульного программирования является : _____
27. Недостаток модульного программирования:
1. увеличивает трудоемкость программирования
 2. усложняет процедуру комплексного тестирования
 3. снижает быстродействие программы
 4. не позволяет выполнить оптимизацию программы
1. Объектный класс в технологии объектно-ориентированного программирования- это _____
 2. Инкапсуляция в технологии объектно-ориентированного программирования _____
 1. вставку объекта в форму
 2. объединение нескольких объектов
 3. вставку одного объекта в другой
 4. объединение данных и программного кода в единое целое
 3. **Наследование** в технологии объектно-ориентированного программирования:
 1. создание объекта из объектного класса
 2. перенос свойств и методов существующего объектного класса на вновь создаваемый на его основе класс
 3. создание новых методов у существующего класса
 4. добавление нового объекта в коллекцию объектов
 4. Понятие **Полиморфизм** в технологии объектно-ориентированного программирования:
 1. различная программная реализация одного и того же свойства у объектов на разных ступенях иерархии наследования
 2. различное количество свойств у объектов на разных ступенях иерархии наследования
 3. различное поведение объектов на разных ступенях иерархии наследования
 4. различное количество методов у объектов на разных ступенях иерархии наследования
 5. **Метод класса** в технологии объектно-ориентированного программирования:
 1. Блок-схема алгоритма обработки данных, реализованного в программном коде объекта
 2. Текстовое описание метода обработки данных, реализованного в программном коде объекта
 3. **Программный код, выполняющий определенные действия с объектом класса**

4. Описание алгоритма обработки данных, реализованного в программном коде объекта
6. **Интерфейс объекта** в технологии объектно-ориентированного программирования – это _____
7. **Событие** в технологии объектно-ориентированного программирования:
 1. Щелчок мышью по изображению объекта на экране
 2. Воздействие среды на объект, вызывающее его реакцию
 3. Любое воздействие пользователя на объект
 4. Воздействие объекта на среду, вызываемое его программным кодом
8. К основным принципам объектно-ориентированного программирования не относится...
 1. инкапсуляция
 2. наследование
 3. полиморфизм
 4. компиляция
9. Идеология программирования в оконных операционных системах основана на:
 1. функциях
 2. событиях
 3. процедурах
 4. объектах
10. Понятия объекта в ООП – это _____
11. Объектно-ориентированная программа - ...
 1. Программирование, основанное на объектах
 2. Совокупность объектов и способов их взаимодействия
 3. Структура среды программирования
 4. Программа с подпрограммами
12. Дополнить предложение. Основным окном разрабатываемого приложения является ...
 1. Код
 2. **Форма**
 3. Объект
 4. Инспектор объектов
13. Lazarus – это _____
14. Объектом называется любая именованная сущность, имеющая:
 1. Свойства, классы, методы
 2. **Свойства, методы, события**
 3. Свойства, типы данных, методы, события
 4. Компонент панели инструментов
15. Методы – это ...
 1. Действия, которые объект может выполнить по заданному коду
 2. **Действия, которые объект может выполнить по команде**
 3. Действия, которые объект может выполнять по запросу
 4. Описание алгоритма обработки данных, реализованного в программном коде объекта

ПК-4

1. Способ создания программ путем манипулирования графическими объектами вместо написания кода в текстовом виде называется _____
2. К визуальным средам не относится:
 1. Visual Basic
 2. **Turbo Pascal**
 3. Lazarus

4. C++
3. Процесс графического отображения сложных процессов или понятий на экране компьютера в виде графических примитивов:
 1. графическое редактирование
 2. оптимизация
 3. **визуализация**
 4. компиляция
4. Проект Lazarus представляет собой свободную среду быстрой разработки программного обеспечения для компилятора:
 1. Java;
 2. **Free Pascal;**
 3. Си++;
 4. Visual Basic.
5. В главное окно проекта Lazarus не входит:
 1. **строка формул;**
 2. палитра компонентов;
6. Представленная структура


```

Unit <имя >;
Interface
    {интерфейсная часть}
Implementation
    {раздел реализации}
Begin
    {раздел инициализации}
End.
      
```

 является структурой ...
 1. процедуры
 2. функции
 3. **модуля**
 4. программы
7. В среде разработки **Lazarus** окно **Инспектор Объектов** предназначен для:
 1. редактирования свойств формы или компонентов и определять события, на которые будет реагировать форма или компоненты
 2. разработки интерфейсной части приложения
 3. ввода программного кода
 4. просмотра состава и свойств проекта
8. Инспектор объектов **Lazarus** содержит страницу:
 1. **"Свойства" ("Properties");**
 2. "Избранное" ("Favorites");
 3. "Процедуры" ("Procedures");
 4. "Функции" ("Functions");
9. Функция StrToInt(s) является функцией преобразования:
 1. целого числа в строку символов
 2. строки символов в действительное число
 3. **строки символов в целое число**
 4. целого числа в дату
10. Функция FloatToStr(x) является функцией преобразования:
 1. **действительного числа в строку символов**
 2. строки символов в действительное число
 3. строки символов в целое число
 4. целого числа в строку символов
11. Тип String предназначен для хранения:

1. целых чисел
 - 2. строка символов**
 3. действительных чисел
 4. массивов
12. С помощью одного оператора IF...THEN можно создать:
1. три пути работы программы
 - 2. два пути работы программы**
 3. 4 пути работы программы
 4. неограниченное количество путей работы программы
13. Компонентом, используемым для размещения на форме однострочных надписей является _____
14. Компонентом, предназначенным для отображения рисунков является _____
15. Свойство, которое не является свойством компонента BUTTON:
1. Caption
 2. Top
 - 3. Picture**
 4. Font
16. Компонент, который представляет собой текстовое окно, в которое можно вводить текст или числа во время выполнения программы размещения текста на форме:
1. Label;
 2. Button;
 - 3. Edit;**
 4. Image;
17. Компонент, который используется для организации выбора из нескольких взаимоисключающих возможностей (при этом допускается выбор только одного варианта из нескольких):
1. GroupBox;
 - 2. RadioButton;**
 3. ListBox;
 4. ComboBox;
18. Компонент-флажок, который используется для обозначения включения или отключения какой либо опции, называется...
1. RadioGroup
 2. Функциональная кнопка
 - 3. CheckBox**
 4. ListBox
19. Компонент, который может выполнять некоторый код через регулярные промежутки времени. Размещенный на форме, он остается невидимым во время работы приложения:
1. GroupBox;
 2. RadioButton;
 3. ListBox;
 - 4. Timer.**
20. Текстовая часть программы пишется непосредственно в окне:
- 1. Редактора кода;**
 2. Инспектора объектов;
 3. Палитры компонентов
 4. Проектировщика форм.
1. Операторы CREATE TABLE и DROP TABLE относятся к группе операторов
1. управления транзакциями

2. манипулирования данными
3. **определения данных**
4. определения доступа к данным
2. Операторы INSERT, DELETE, UPDATE относятся к группе операторов
 1. управления курсором
 2. управления транзакциями
 3. **манипулирования данными**
 4. определения доступа к данным
3. Операции COMMIT и ROLLBACK относятся к операциям
 4. определения доступа к данным
 1. запросов
 2. **управления транзакциями**
 3. манипулирования данными
5. Ключевое слово ALL в операторе SELECT означает, что
 1. все символы в результирующем наборе заменяются символом *
 2. выражению можно присвоить псевдоним
 3. в результирующий набор включаются все строки, удовлетворяющие условиям запроса
 4. **в результирующий набор включаются все столбцы из исходных таблиц запроса**
6. Для объединения таблиц используют оператор _____
7. Константы 'Иванов И.И.' , '123456789' относятся к типу
 1. BIT
 2. CHAR
 3. **VARCHAR**
 4. NUMBER
8. Какой оператор следует использовать при выборе данных с учетом диапазона значений:
 1. In
 2. **Between**
 3. Like
 4. Where
9. Укажите, с какой обязательной командой всегда используется Select?
 1. Where
 2. Like
 3. **From**
 4. In
10. Операторы, которые являются основными при модификации данных с помощью SQL
 1. Create Table и Drop Table
 2. **Insert, Delete и Update**
 3. COMMIT, ROLLBACK
 4. Select, From и Where
11. Оператор Union позволяет:
 1. Объединить несколько столбцов одной таблицы.
 2. Объединить несколько строк одной таблицы.
 3. Объединить несколько **таблиц по их строкам.**
 4. **Обеспечить доступ к данным**
12. Отладка –это:
 1. определение списка параметров
 2. правило вызова процедур (функций)
 3. **процедура поиска ошибок, когда известно, что ошибка есть**

4. составление блок-схемы алгоритма
13. Ошибки при составлении программы бывают _____
14. Синтаксические ошибки программы обнаруживаются при _____
15. Трассировка программы – это ...
 1. тестирование исходного кода
 2. отладка модуля
 3. составление блок-схемы алгоритма
 - 4. пошаговая проверка выполнения программы**
16. Тестирование, предполагающее детальное изучение логики программы, и построение таких входных данных, которые позволили бы при многократном выполнении программы обеспечить выполнение максимально возможного количества маршрутов, логических ветвлений, циклов и т.д. является ...
 1. функциональным тестированием
 2. тестированием «черного ящика»
 3. тестирование «белого ящика»
 4. динамическим тестированием
17. Тестирование, которое полностью абстрагируется от логики программы, а тестовые наборы подбираются на основании анализа функциональных входных спецификаций, является ...
 1. структурным тестированием
 2. тестированием «черного ящика»
 3. тестирование «белого ящика»
 4. динамическим тестированием

Задания для контрольных работ

по дисциплине Высокоуровневые методы информатики и программирования

Варианты заданий

Вариант 0	1	Этапы решения задач на ЭВМ
	2	Lazarus- ввод и вывод текстовых файлов
	3	Составить проект на Lazarus для решения задачи Вычислить объем и площадь прямоугольного параллелепипеда по заданным длинам ребер a, b и c .
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Для целочисленных массивов A(5) и B(6) найти максимальный и минимальный элементы.
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих действий над элементами одномерного целочисленного массива: 1) Нахождение суммы четных элементов; 2) Нахождение произведения отрицательных элементов. Найти сумму четных и произведение отрицательных элементов для заданного массива A(20) .
	6	Задан текстовый файл. Загрузить его в компонент многострочный редактор. Переписать в другой текстовый файл нечетные строки.

Вариант 1	1	Технологии программирования, классификация.
	2	Lazarus-компоненты Edit, Button, их свойства
	3	Составить проект на Lazarus для решения задачи Вычислить периметр и площадь прямоугольного треугольника по длинам двух катетов a и b .
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Для целочисленных массивов A(5,5) и B(6,6) найти суммы элементов, расположенных ниже главной диагонали.
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих действий над элементами одномерного массива: 1) Нахождение суммы положительных элементов; 2) Нахождение произведения отрицательных элементов. Найти сумму положительных и произведение отрицательных элементов для заданного массива.
	6	Задан текстовый файл. Загрузить его в компонент Мемо. Определить число строк в файле. Ответ вывести в компонент Label

Вариант 2	1	Отладка и тестирование программ.
	2	Функции преобразования в числовой формат и обратно в Lazarus.
	3	Составить проект на Lazarus для решения задачи Дана длина ребра куба. Найти площадь грани, площадь полной поверхности и объем этого куба.
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Найти сумму положительных элементов целочисленных массивов

		A(6,6) и B(5,5).
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих действий над элементами одномерного массива: 1) Нахождение максимального элемента; 2) Нахождение минимального элемента. Найти максимальный и минимальный элементы для заданного массива.
	6	Задан текстовый файл. Загрузить его в компонент многострочный редактор. Переписать в другой текстовый файл строки, в которых нет цифр.

Вариант 3	1	Технологии программирования
	2	Lazarus- компоненты ListBox, CheckBox, ComboBox
	3	Составить проект на Lazarus для решения задачи Известна длина окружности. Найти площадь круга, ограниченного этой окружностью.
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Вычислить сумму и произведение элементов целочисленных массивов X(3,5) и Y(3,4).
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих действий над элементами одномерного массива: 1) Нахождение суммы элементов; 2) Нахождение произведения элементов. Найти сумму и произведение элементов для заданного массива.
	6	Задан текстовый файл. Загрузить его в компонент многострочный редактор. Переписать в другой текстовый файл строки, в которых нет пробелов.

Вариант 4	1	Структурное программирование
	2	Lazarus- компоненты Label, CheckBox и их свойства
	3	Составить проект на Lazarus для решения задачи Даны два действительных числа x и y . Вычислить их сумму, разность, произведение и частное.
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Найти минимальные элементы и их порядковые номера действительных массивов A(10) и B(5).
	5	Реализовать в виде модуля набор подпрограмм для вычисления суммы, произведения, количества положительных элементов квадратной матрицы. Найти сумму и произведение элементов для матрицы X(5,5).
		Задан текстовый файл. Загрузить его в компонент Мемо. Определить количество символов в файле. Ответ вывести в компонент Label.

Вариант 5	1	Модульное программирование
	2	Lazarus- компоненты Мемо, RadioButton и их свойства
	3	Составить проект на Lazarus для решения задачи Найти площадь кольца, внутренний радиус которого равен r , а внешний – $R(R>r)$.
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Найти среднее арифметическое отрицательных элементов массивов X(3,5) и Y(2,3).
	5	Реализовать в виде модуля набор подпрограмм для вычисления объема геометрических тел(шар, куб, цилиндр, прямоугольный параллелепипед,

		тетраэдр). Используя этот модуль, решить задачу. Вычислить объем шара, куба, цилиндра.
	6	Задан текстовый файл. Загрузить его в компонент Мемо. Определить количество символов в первой строке. Ответ вывести в компонент Label

Вариант 6	1	Объектно-ориентированное программирование
	2	Lazarus- компоненты StringGrid, Image и их свойства
	3	Составить проект на Lazarus для решения задачи Вычислить периметр и площадь прямоугольного треугольника по длинам двух катетов a и b .
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Вычислить сумму элементов массивов A(20) и B(15) с номерами кратными 3.
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих операций над векторами: 1) сложения; 2) вычитания; 3) скалярного умножения; 4) умножения вектора на число; 5) нахождения длины вектора. Вектор представить следующим типом: Const NMax=10; Type Vector=Array[1..NMax] of Real; Используя этот модуль, решить задачу. Дан массив A – массив векторов. Найти длины векторов для элементов этого массива.
	6	Задан текстовый файл. Загрузить его в компонент Мемо. Определить количество символов в последней строке. Ответ вывести в компонент Label

Вариант 7	1	Принципы ООП
	2	Lazarus – работа с файлами
	3	Составить проект на Lazarus для решения задачи Написать программу – модель анализа пожарного датчика в помещении, которая выводит сообщение "Пожарная ситуация", если температура в комнате превысила 60° С.
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Найти сумму положительных элементов целочисленных массивов A(6) и B(10).
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих операций с квадратными матрицами: 1) сложения двух матриц; 2) умножения одной матрицы на другую; 3) нахождения транспонированной матрицы. Матрицу описать следующим образом: Const NMax = 10; Type Matrica=Array[1..NMax,1..NMax] of Real; Используя этот модуль, решить задачу.

		Вычислить сумму трех заданных матриц.
	6	Задан текстовый файл. Загрузить его в компонент Мемо. Переписать в другой файл первые три строки файла.

Вариант 8	1	Структура модуля на языке Паскаль
	2	Этапы создания приложения в средах быстрой разработки
	3	Составить проект на Lazarus для решения задачи Найти произведение цифр заданного четырехзначного числа.
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Найти максимальные элементы и их порядковые номера действительных массивов A(10) и B(5).
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих операций с квадратными матрицами: 1) сложения двух матриц; 2) умножения одной матрицы на другую; 3) нахождения транспонированной матрицы. Матрицу описать следующим образом: Const NMax = 10; Type Matrica=Array[1..NMax,1..NMax] of Real; Используя этот модуль, решить задачу. Задан массив величин типа Matrica. Вычислить произведение элементов массива.
	6	Задан текстовый файл. Загрузить его в компонент Мемо. Переписать в другой файл последние три строки файла.

Вариант 9	1	Конструкторы и деструкторы
	2	Lazarus – работа с массивами
	3	Составить проект на Lazarus для решения задачи Известна длина окружности. Найти площадь круга, ограниченного этой окружностью
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Найти сумму элементов для действительных массивов A(5,5) и B(4,6).
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих операций над обыкновенными дробями вида P/Q (P – целое, Q – натуральное): 1) сложение; 2) умножение; 3) вычитание; 4) деление. . Дробь представить следующим типом: Type Frac=Record P: Integer; Q: 1..32767 End; Используя этот модуль, решить задачу. Дан массив A(5) – массив обыкновенных дробей. Вычислить среднее арифметическое всех дробей, результат представить в виде несократимой дроби.
	6	Создайте проект, содержащий компоненты ListBox и обеспечивающий решение следующей задачи:

		кнопка позволяет выводить в ListBox содержимое текстового файла.
Вариант 10	1	Особенности программирования в оконных операционных средах.
	2	Lazarus – ввод, вывод данных
	3	Составить проект на Lazarus для решения задачи Написать программу, определяющую результат гадания на ромашке – "любит – не любит", взяв за исходные данные количество лепестков п.
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Найти сумму и количество отрицательных элементов массивов X(5,6) и Y(3,4).
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих операций над обыкновенными дробями вида P/Q (P – целое, Q – натуральное): 1) сложение; 2) умножение; 3) вычитание; 4) деление. . Дробь представить следующим типом: <pre> Type Frac=Record P: Integer; Q: 1..32767 End;</pre> Используя этот модуль, решить задачу. Дан массив A (10)– массив обыкновенных дробей. Вычислить произведение всех элементов массива.
	6	Создайте проект, содержащий компоненты ListBox и Button, обеспечивающий решение следующей задачи: кнопка позволяет выводить в ListBox содержимое текстового файла.
Вариант 11	1	Основные события Lazarus.
	2	Структура модуля на языке Паскаль
	3	Составить проект на Lazarus для решения задачи Написать программу, которая проверяет, является ли год високосным. Исходные данные – год.
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Для действительных массивов A(5) и B(16) найти минимальные элементы.
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих операций над комплексными числами: 1) сложения; 2) вычитания; 3) умножения; 4) модуля комплексного числа. Комплексное число представить следующим типом: <pre> Type Complex=Record R: Real; M: Real End;</pre> Используя этот модуль, решить следующую задачу. Дан массив A – массив комплексных чисел. Получить массив C, элементами которого будут модули сумм рядом стоящих комплексных чисел.
	6	Создайте проект, содержащий компоненты Memo, Label и Button, обеспечивающий решение следующей задачи: кнопка позволяет определить минимальный из положительных элементов

		целочисленного массива A(10) и вывести результат в компонент Label. Для ввода массива использовать компонент Memo.
Вариант 12	1	Среда разработки Lazarus.
	2	Case-технологии
	3	Составить проект на Lazarus для решения задачи Заданы координаты трех вершин треугольника $(x_1, y_1), (x_2, y_2), (x_3, y_3)$. Найти его периметр и площадь.
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Вычислить сумму и произведение элементов целочисленных массивов X(3,2) и Y(3,4).
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих операций над комплексными числами: 1) сложения; 2) вычитания; 3) умножения; 4) модуля комплексного числа. Комплексное число представить следующим типом: Type Complex=Record R: Real; M: Real End: Используя этот модуль, решить следующую задачу. Дан массив A(M) – массив комплексных чисел. Получить матрицу B(N,M), каждая строка которой получается возведением в степень, равную номеру этой строки, данного массива A.
	6	Создайте проект, содержащий компоненты Memo, Label и Button, обеспечивающий решение следующей задачи: кнопка позволяет определить сумму элементов целочисленного массива X(10) и вывести результат в компонент Label. Для ввода массива использовать компонент Memo.

Вариант 13	1	Законы эволюции ПО
	2	Основные термины ООП. Класс. Экземпляр класса. Методы.
	3	Составить проект на Lazarus для решения задачи Даны два действительных числа x и y. Вычислить их сумму, разность, произведение и частное.
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Найти среднее арифметическое положительных элементов массивов X(3,5) и Y(2,3).
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих операций с квадратными матрицами: 1) сложения двух матриц; 2) умножения одной матрицы на другую; 3) нахождения транспонированной матрицы. Матрицу описать следующим образом: Const NMax = 10; Type Matrica=Array[1..NMax,1..NMax] of Real; Используя этот модуль, решить задачу. Вычислить произведение 3 заданных матриц.
	6	Создайте проект, содержащий компоненты StringGrid, Label и Button, обеспечивающий решение следующей задачи:

		кнопка позволяет определить минимальный элемент массива X(10) и вывести результат в компонент Label. Для ввода массива использовать компонент StringGrid.
--	--	---

Вариант 14	1	Среда быстрой разработки программ. Система окон.
	2	ИВ EXPERT – создание и регистрация базы данных
	3	Составить проект на Lazarus для решения задачи Написать программу, которая проверяет, является ли год високосным. Исходные данные – год.
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Вычислить количество элементов равных заданному числу x для массивов A(2,3) и B(3,4).
	5	Реализовать в виде модуля набор подпрограмм для вычисления объема геометрических тел(шар, куб, цилиндр, прямоугольный параллелепипед, тетраэдр). Используя этот модуль, решить задачу. Вычислить объем тетраэдра, куба, цилиндра.
	6	Создайте проект, содержащий компоненты StringGrid, Label и Button, обеспечивающий решение следующей задачи: кнопка позволяет определить сумму и количество положительных элементов массива X(10) и вывести результат в компонент Label. Для ввода массива использовать компонент StringGrid.

Вариант 15	1	Типы ошибок в программе
	2	Среда разработки Lazarus.. Главное окно
	3	Составить проект на Lazarus для решения задачи Написать программу проверки знания даты начала второй мировой войны. В случае неправильного ответа пользователя, программа должна выводить сообщение об ошибке и правильный ответ. Исходные данные – дата.
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Вычислить сумму и количество элементов больших 5 для действительных массивов B(20) и C(10).
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих операций над комплексными числами: 1) сложения; 2) вычитания; 3) умножения; 4) модуля комплексного числа. Комплексное число представить следующим типом: Type Complex=Record R: Real; M: Real End: Используя этот модуль, решить следующую задачу. Даны два массива комплексных чисел. Получить массив C, элементами которого будут суммы элементов заданных массивов.
	6	Создайте проект, содержащий компоненты StringGrid, Label и Button, обеспечивающий решение следующей задачи: кнопка позволяет определить количество положительных элементов массива Y(10) и вывести результат в компонент Label. Для ввода массива использовать компонент StringGrid.

Вариант	1	Объекты в Lazarus. Описание класса.
---------	---	-------------------------------------

16	2	Визуальное программирование.
	3	Составить проект на Lazarus для решения задачи Написать программу, которая сравнивает два введенных с клавиатуры числа. Программа должна указать, какое число больше, или, если числа равны, вывести соответствующее сообщение.
		Составить программу на языке Паскаль с использованием внешней подпрограммы. Вычислить сумму элементов массивов X(15) и Y(10).
		Реализовать в виде модуля набор подпрограмм для выполнения следующих действий над элементами одномерного массива: 1) Нахождение максимального элемента; 2) Нахождение минимального элемента. Используя этот модуль, найти сумму максимального и минимального элементов для массива A(20). Найти максимальный и минимальный элементы для заданного массива.
	6	Создайте проект, содержащий компоненты StringGrid, Label и Button, обеспечивающий решение следующей задачи: кнопка позволяет определить сумму положительных элементов массива X(10) и вывести результат в компонент Label. Для ввода массива использовать компонент StringGrid.

Вариант 17	1	Основные понятия объектно-ориентированного программирования.
	2	Исключительные ситуации. Обработка исключительных ситуаций
	3	Составить проект на Lazarus для решения задачи Написать программу. Даны действительные числа a, b, c. Удвоить эти числа, если $a > b > c$, и заменить их абсолютными значениями, если это не так.
	4	Составить программу на языке Паскаль с использованием внешней подпрограммы. Вычислить сумму элементов массивов A(20) и B(15) с нечетными номерами.
	5	Реализовать в виде модуля набор подпрограмм для выполнения следующих операций над комплексными числами: 1) сложения; 2) вычитания; 3) умножения; 4) модуля комплексного числа. Комплексное число представить следующим типом: Type Complex=Record R: Real; M: Real End: Используя этот модуль, решить следующую задачу. Дан массив X – массив комплексных чисел. Найти произведение элементов массива.
	6	Создайте проект, содержащий компоненты StringGrid, Label и Button, обеспечивающий решение следующей задачи: кнопка позволяет определить максимальный элемент массива A(5,5) и вывести результат в компонент Label. Для ввода массива использовать компонент StringGrid.

Вариант 18	1	Основные понятия объектно-ориентированного программирования.
	2	IB EXPERT – разработка структуры таблиц
	3	Составить проект на Lazarus для решения задачи Даны действительные числа a, b, c. Удвоить эти числа, если $a > b > c$, и заменить их абсолютными значениями, если это не так.

4	<p>Составить программу на языке Паскаль с использованием внешней подпрограммы. Вычислить произведение и количество элементов массивов X(10) и Y(12), модуль которых больше 9.</p>
5	<p>Реализовать в виде модуля набор подпрограмм для выполнения следующих операций над комплексными числами: 1) сложения; 2) вычитания; 3) умножения; 4) модуля комплексного числа. Комплексное число представить следующим типом: Type Complex=Record R: Real; M: Real End: Используя этот модуль, решить следующую задачу. Дан массив A – массив комплексных чисел. Найти сумму элементов массива.</p>
6	<p>Создайте проект, содержащий компоненты StringGrid, Label и Button, обеспечивающий решение следующей задачи: кнопка позволяет определить среднее арифметическое положительных элементов массива A(5,5) и вывести результат в компонент Label. Для ввода массива использовать компонент StringGrid.</p>

5. Методические материалы, определяющие процедуры оценивания компетенции

5.1 Критерии оценивания качества устного ответа

Оценка «отлично» выставляется за глубокое знание предусмотренного программой материала, за умение четко, лаконично и логически последовательно отвечать на поставленные вопросы.

Оценка «хорошо» – за твердое знание основного (программного) материала, за грамотные, без существенных неточностей ответы на поставленные вопросы.

Оценка «удовлетворительно» – за общее знание только основного материала, за ответы, содержащие неточности или слабо аргументированные, с нарушением последовательности изложения материала.

Оценка «неудовлетворительно» – за незнание значительной части программного материала, за существенные ошибки в ответах на вопросы, за неумение ориентироваться в материале, за незнание основных понятий дисциплины.

5.2 Критерии оценивания тестирования

При тестировании все верные ответы берутся за 100%.

90%-100% отлично

75%-90% хорошо

60%-75% удовлетворительно

менее 60% неудовлетворительно

5.3 Критерии оценивания качества выполнения лабораторного практикума

Оценка «зачтено» выставляется обучающемуся, если лабораторная работа выполнена правильно и обучающийся ответил на все вопросы, поставленные преподавателем на защите.

Оценка «не зачтено» выставляется обучающемуся, если лабораторная работа выполнена не правильно или обучающийся не проявил глубоких теоретических знаний при защите работы

5.4 Критерии оценивания контрольной работы

При проверке контрольной работы:

выполнено 6 заданий – отлично

выполнено 4-5 заданий – хорошо

выполнено 3 задания – удовлетворительно

выполнено менее 3 заданий – неудовлетворительно

5.5 Критерии оценивания результатов освоения дисциплины

Оценка «отлично» выставляется за глубокое знание предусмотренного программой материала, содержащегося в основных и дополнительных рекомендованных литературных источниках, за умение четко, лаконично и логически последовательно отвечать на поставленные вопросы, за умение анализировать изучаемые явления в их взаимосвязи и диалектическом развитии, применять теоретические положения при решении практических задач.

Оценка «хорошо» – за твердое знание основного (программного) материала, включая расчеты (при необходимости), за грамотные, без существенных неточностей ответы на поставленные вопросы, за умение применять теоретические положения для решения практических задач.

Оценка «удовлетворительно» – за общее знание только основного материала, за ответы, содержащие неточности или слабо аргументированные, с нарушением последовательности изложения материала, за слабое применение теоретических положений при решении практических задач.

Оценка **«неудовлетворительно»** – за незнание значительной части программного материала, за существенные ошибки в ответах на вопросы, за неумение ориентироваться в расчетах, за незнание основных понятий дисциплины.