

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКАЯ ГОСУДАРСТВЕННАЯ АКАДЕМИЯ»

«УТВЕРЖДАЮ»

Проректор по учебной работе  Г.Ю. Нагорная
« 21 » 03 2026 г.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Языки и методы программирования

Уровень образовательной программы бакалавриат

Направление подготовки 01.03.02 Прикладная математика и информатика

Направленность(профиль) «Математические и информационные системы и технологии в астрономии»

Форма обучения: очная

Срок освоения ОП 4 года

Институт Цифровых технологий

Кафедра разработчик РПД Математика

Выпускающая кафедра Астрофизика

Начальник
учебно-методического управления

 Семенова Л. У.

Директор института ЦТ

 Кумратова А. М.

И. О. заведующего выпускающей кафедрой

 Валявин Г. Г.

г. Черкесск, 2026 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	7
	4.2.3. Лабораторный практикум	8
	4.2.4. Практические занятия	10
	4.3. Самостоятельная работа обучающегося	12
5	Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине	12
6	Образовательные технологии	18
7	Учебно-методическое и информационное обеспечение дисциплины	19
	7.1. Перечень основной и дополнительной учебной литературы	19
	7.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»	19
	7.3. Информационные технологии, лицензионное программное обеспечение	20
8	Материально-техническое обеспечение дисциплины	20
	8.1. Требования к аудиториям (помещениям, местам) для проведения занятий	20
	8.2. Требования к оборудованию рабочих мест преподавателя и обучающихся:	22
	8.3. Требования к специализированному оборудованию	22
9	Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья	23
	Приложение 1. Фонд оценочных средств	24

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины «Языки и методы программирования» является ознакомление обучающихся с историей развития языков программирования, с парадигмами программирования, структурами данных, знакомство с методами, применяемыми в программировании, известными алгоритмами, возможностью работы с динамической памятью.

Задачи дисциплины:

В результате освоения дисциплины обучающийся должен знать синтаксис и семантику выполнения операторов языков программирования, уметь применять языки программирования для решения широкого круга задач.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

2.1. Дисциплина «Языки и методы программирования» относится к части, формируемой участниками образовательных отношений Блока 1. Дисциплины (модули), 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	Основы информатики	Технологии программирования в Internet
2	Процедурное программирование	Проект 2 курса
3	Основы программирования	Основы программирования в 1С
4		Системы программирования

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Планируемые результаты освоения образовательной программы (ОП) – компетенции обучающихся определяются требованиями стандарта по направлению подготовки 01.03.02 Прикладная математика и 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 осуществлять программную реализацию алгоритмов при разработке прикладных информационных систем.	<p>ПК-3.1 Обладает знаниями в области программирования и современных компьютерных технологий при разработке прикладных информационных систем. Способен осуществлять оптимизацию алгоритмов при создании прикладных программ или прикладных комплексов</p> <p>ПК-3.2 Анализирует стандартные технические требования к проектированию программного обеспечения.</p> <p>ПК-3.3 Способен осуществлять разработку технических требований к программному обеспечению, при его создании</p>

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Вид учебной работы	Всего часов	Семестры	
		№ 3	
		Часов	
1	2	3	
Аудиторная контактная работа (всего)	54	54	
В том числе:			
Лекции (Л)	18	18	
Лабораторные работы (ЛР)	36	36	
Практические занятия (ПЗ)	-	-	
Контактная внеаудиторная работа в том числе:	2	2	
Индивидуальные групповые консультации	2	2	
Самостоятельная работа обучающегося (СР) (всего)	16	16	
<i>Подготовка к лабораторным занятиям</i>	4	4	
<i>Подготовка к текущему контролю</i>	4	4	
<i>Контрольная работа</i>	4	4	
<i>Подготовка к промежуточному контролю</i>	4	4	
Промежуточная аттестация	Зачет, экзамен (ЗаО, Э) В том числе	Э	
	Прием зачета, час	-	
	Прием экз., час	0,5	
	Консультации	2	
	СРО, час	33,5	
ИТОГО: Общая трудоемкость	Часов	108	
	зач. ед.	3	

4.2. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

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

№ п/п	Наименование раздела (темы) 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
Семестр 3							
1.	Раздел 1. Основы программирования	4	4	-	4	12	контрольные вопросы, индивидуальные задания к лабораторным работам, тестирование, контрольная работа
2.	Раздел 2. Интегрированная среда разработки Lazarus	6	6	-	4	16	контрольные вопросы, индивидуальные задания к лабораторным работам, тестирование, контрольная работа
3.	Раздел 3. Введение в язык программирования Python	8	26	-	8	42	контрольные вопросы, индивидуальные задания к лабораторным работам, тестирование, контрольная работа
	Контактная внеаудиторная работа, в том числе: индивидуальные и групповые консультации					2	
4.	Промежуточная аттестация	-				36	Экзамен
Итого в 3 семестре:		18	36	-	16	108	
Всего:		18	36	-	16	108	

4.2.2. Лекционный курс

№ п/п	Наименование раздела дисциплины	Наименование темы лекции	Содержание лекции	Всего часов
1	2	3	4	5
Семестр 3				
1.	Раздел 1. Основы программирования	Программирование алгоритмов линейной , разветвляющейся и циклической структуры	Изучение объявления и использования констант, переменных, стандартных функций; запись арифметических выражений; организации ввода-вывода данных. Изучить возможности языка программирования для реализации: условной и безусловной передаче управления; вычислительного процесса разветвляющейся структуры; вычислительного процесса циклической структуры; приема программирования-табулирования функции от одного аргумента.	2

		<p>Обработка матриц</p> <p>Программирование с использованием подпрограмм пользователя</p>	<p>Изучение правила организации вложенного цикла с учетом порядка перебора элементов матрицы; правила использования приемов программирования в структурах с вложенными циклами и способа ввода-вывода матриц. Разработка алгоритм решения в соответствии с заданием и составить программу решения задачи.</p> <p>Изучение разновидности подпрограмм и правила их записи; способы передачи параметров в подпрограммы; правила записи программ, использующих подпрограммы различных видов; порядок выполнения программ, использующих подпрограммы (процедуры и функции)</p>	2
2.	<p>Раздел 2. Интегрированная среда разработки Lazarus.</p>	<p>Текстовые компоненты и списки в Lazarus</p> <p>Кнопки, индикаторы и управляющие элементы в Lazarus</p>	<p>Изучение свойств заданных компонентов, используя Help и ИСП Lazarus,</p> <p>Изучение свойств заданных компонентов, используя Help и ИСП Lazarus,</p>	4
		<p>Работа с меню, панелями и компонентами внешнего оформления</p> <p>Обработка событий клавиатуры и мыши</p>	<p>Изучение свойств заданных компонентов, используя Help и ИСП Lazarus,</p>	2

3.	Раздел 3. Введение в язык программирования Python	Введение в язык программирования Python Математические операции в Python	Изучение основных типов данных, команды ввода и вывода данных Изучение основных математических операций в Python	2
		Структура ветвление в Python Работа с циклами в Python	Изучение структуры ветвления (if, if-else, if-elif-else), работа с числами и строками используя данную структуру Изучение циклических конструкций	2
		Работа со строками списками в Python. Операции над списками в Python	Изучение методов работы со строками Изучение списков в Python	2
		Функции в Python Работа с двумерными массивами	Изучение функции в Python Изучение вложенных списков в Python	2
Итого часов в 3 семестре:				18
Всего:				18

4.2.3. Лабораторный практикум

№ п/п	Наименование раздела дисциплины	Наименование лабораторной работы	Содержание лабораторной работы	Всего часов
1	2	3	4	5
Семестр 3				
1. 2.	Раздел 1. Основы программирования	Лабораторная работа № 1 «Программирование алгоритмов линейной структуры»	Изучение объявления и использования констант, переменных, стандартных функций; запись арифметических выражений; организации ввода-вывода данных. Изучить возможности языка программирования для реализации: условной и безусловной передаче управления;	2
		Лабораторная работа № 2 «Программирование алгоритмов разветвляющейся и циклической структуры»		

			вычислительного процесса разветвляющейся структуры; вычислительного процесса циклической структуры; приема программирования-табулирования функции от одного аргумента.	
		Лабораторная работа №3 «Обработка матриц» Лабораторная работа №4 «Программирование с использованием подпрограмм пользователя»	Изучение правила организации вложенного цикла с учетом порядка перебора элементов матрицы; правила использования приемов программирования в структурах с вложенными циклами и способа ввода-вывода матриц. Разработка алгоритм решения в соответствии с заданием и составить программу решения задачи. Изучение разновидности подпрограмм и правила их записи; способы передачи параметров в подпрограммы; правила записи программ, использующих подпрограммы различных видов; порядок выполнения программ, использующих подпрограммы (процедуры и функции).	2
3.	Раздел 2. Интегрированная среда разработки Lazarus.	Лабораторная работа №5 Текстовые компоненты и списки в Lazarus	Изучение свойств заданных компонентов, используя Help и ИСП Lazarus	2
		Лабораторная работа №6 Кнопки, индикаторы и управляющие элементы в Lazarus	Изучение свойств заданных компонентов, используя Help и ИСП Lazarus	2

		Лабораторная работа №7 Работа с меню, панелями и компонентами внешнего оформления Лабораторная работа №8 Обработка событий клавиатуры и мыши	Изучение свойств заданных компонентов, используя Help и ИСП Lazarus	2
4.	Раздел 3. Введение в язык программирования Python	Лабораторная работа №9 Введение в язык программирования Python	Изучение основных типов данных, команды ввода и вывода данных	2
5.		Лабораторная работа №10 Математические операции в Python	Изучение основных математических операций в Python	2
6.		Лабораторная работа №11 Структура ветвление в Python	Изучение структуры ветвления (if, if-else, if-elif-else), работа с числами и строками используя данную структуру	4
7.		Лабораторная работа №12 Работа с циклами в Python	Изучение циклических конструкций	2
8.		Лабораторная работа №13 Работа со строками в Python	Изучение методов работы со строками	2
9.		Лабораторная работа №14 Работа со списками. Операции над списками в Python	Изучение списков в Python	2
		Лабораторная работа №15 Функции в Python	Изучение функции в Python	6
		Лабораторная работа №16 Работа с двумерными массивами	Изучение вложенных списков в Python	6
Итого часов в 3 семестре				36
Всего				36

4.2.4. Практические занятия (не предусмотрено)

4.3. САМОСТОЯТЕЛЬНАЯ РАБОТА ОБУЧАЮЩЕГОСЯ

№ п/п	Наименование раздела (темы) дисциплины	№ п/п	Виды СРО	Всего часов
1	2	3	4	5
Семестр 3				
1.	Раздел 1 Основы программирования	1.1.	Подготовка к лабораторным занятиям	4
		1.2.	Подготовка к текущему контролю	
		1.3.	Контрольная работа	
		1.4.	Подготовка к промежуточному контролю	
2.	Раздел 2. Интегрированная среда разработки Lazarus.	2.1.	Подготовка к лабораторным занятиям	4
		2.2.	Подготовка к текущему контролю	
		2.3.	Контрольная работа	
		2.4.	Подготовка к промежуточному контролю	
3.	Раздел 3. Введение в язык программирования Python	3.1	Подготовка к лабораторным занятиям	8
		3.2	Подготовка к текущему контролю	
		3.3	Контрольная работа	
		3.2	Подготовка к промежуточному контролю	
ИТОГО часов в 3 семестре:				16
Всего				16

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

5.1. Методические указания для обучающихся к лекционным занятиям

Какими бы замечательными качествами в области методики ни обладал лектор, какое бы большое значение на занятиях ни уделял лекции слушатель, глубокое понимание материала достигается только путем самостоятельной работы над ним.

Работа над конспектом лекции осуществляется по этапам:

- повторить изученный материал по конспекту;
- непонятные положения отметить на полях и уточнить;
- неоконченные фразы, пропущенные слова и другие недочеты в записях устранить, пользуясь материалами из учебника и других источников;
- завершить техническое оформление конспекта (подчеркивания, выделение главного, выделение разделов, подразделов и т.п.).

Самостоятельную работу следует начинать с доработки конспекта, желательно в тот же день, пока время не стерло содержание лекции из памяти (через 10 ч после лекции

в памяти остается не более 30-40 % материала). Работа над конспектом не должна заканчиваться с прослушивания лекции. После лекции, в процессе самостоятельной работы, перед тем, как открыть тетрадь с конспектом, полезно мысленно восстановить в памяти содержание лекции, вспомнив ее структуру, основные положения и выводы.

С целью доработки необходимо прочитать записи, восстановить текст в памяти, а также исправить описки, расшифровать не принятые ранее сокращения, заполнить пропущенные места, понять текст, вникнуть в его смысл. Далее прочитать материал по рекомендуемой литературе, разрешая в ходе чтения, возникшие ранее затруднения, вопросы, а также дополнения и исправляя свои записи. Записи должны быть наглядными, для чего следует применять различные способы выделений. В ходе доработки конспекта углубляются, расширяются и закрепляются знания, а также дополняется, исправляется и совершенствуется конспект. Еще лучше, если вы переработаете конспект, дадите его в новой систематизации записей. Это, несомненно, займет некоторое время, но материал вами будет хорошо проработан, а конспективная запись его приведена в удобный для запоминания вид. Введение заголовков, скобок, обобщающих знаков может значительно повысить качество записи. Этому может служить также подчеркивание отдельных мест конспекта красным карандашом, приведение на полях или на обратной стороне листа краткой схемы конспекта и др.

Подготовленный конспект и рекомендуемая литература используется при подготовке к практическому (семинарскому) занятию. Подготовка сводится к внимательному прочтению учебного материала, к выводу с карандашом в руках всех утверждений и формул, к решению примеров, задач, к ответам на вопросы, предложенные в конце лекции преподавателем или помещенные в рекомендуемой литературе. Примеры, задачи, вопросы по теме являются средством самоконтроля.

Непременным условием глубокого усвоения учебного материала является знание основ, на которых строится изложение материала. Обычно преподаватель напоминает, какой ранее изученный материал и в какой степени требуется подготовить к очередному занятию. Эта рекомендация, как и требование систематической и серьезной работы над всем лекционным курсом, подлежит безусловному выполнению. Потери логической связи как внутри темы, так и между ними приводит к негативным последствиям: материал учебной дисциплины перестает основательно восприниматься, а творческий труд подменяется утомленным переписыванием. Обращение к ранее изученному материалу не только помогает восстановить в памяти известные положения, выводы, но и приводит разрозненные знания в систему, углубляет и расширяет их. Каждый возврат к старому материалу позволяет найти в нем что-то новое, переосмыслить его с иных позиций, определить для него наиболее подходящее место в уже имеющейся системе знаний. Неоднократное обращение к пройденному материалу является наиболее рациональной формой приобретения и закрепления знаний. Очень полезным, но, к сожалению, еще мало используемым в практике самостоятельной работы, является предварительное ознакомление с учебным материалом. Даже краткое, беглое знакомство с материалом очередной лекции дает многое. Обучающиеся получают общее представление о ее содержании и структуре, о главных и второстепенных вопросах, о терминах и определениях. Все это облегчает работу на лекции и делает ее целеустремленной.

5.2. Методические указания для подготовки обучающихся к лабораторным занятиям

Ведущей дидактической целью лабораторных занятий является систематизация и обобщение знаний по изучаемой теме, приобретение практических навыков по тому или другому разделу курса, закрепление практически полученных теоретических знаний.

В начале каждого лабораторного занятия кратко приводится теоретический материал, необходимый для решения задач по данной теме. После него предлагается

решение этих задач и список заданий для самостоятельного выполнения.

Практическая работа включает в себя самоконтроль по предложенным вопросам, выполнение творческих и проверочных заданий, тестирование по теме.

Лабораторные работы сопровождают и поддерживают лекционный курс.

Количество лабораторных работ в строгом соответствии с содержанием курса. Каждая лабораторная предусматривает получение практических навыков по лекционным темам дисциплины «Языки и методы программирования». Для обучающегося подготовлен набор индивидуальных заданий по каждой лабораторной работе. В каждой лабораторной работе обучающийся оформляет полученные результаты. Также в текущей аттестации к лабораторным занятиям предусмотрена форма контроля в виде устной защиты каждого практического индивидуального задания по всем темам лабораторных занятий.

При проведении промежуточной и итоговой аттестации обучающегося важно всегда помнить, что систематичность, объективность, аргументированность — главные принципы, на которых основаны контроль и оценка знаний. Проверка, контроль и оценка знаний.

По окончании курса обучающимися сдается зачет, в ходе которого они должны показать свои теоретические знания и практические навыки по дисциплине «Языки и методы программирования».

5.3. Методические указания для подготовки обучающихся к практическим занятиям

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

В начале семестра обучающиеся получают сводную информацию о формах проведения занятий и формах контроля знаний. Тогда же обучающимся предоставляется список тем лекционных и практических заданий, а также тематика рефератов. Каждое практическое 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, сдают коллоквиумы и делают реферативные работы по дополнительным материалам курса.

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 практических занятий:

1. Изложение преподавателем темы занятия, его целей и задач.
2. Выдача преподавателем задания обучающимся, необходимые пояснения.
3. Выполнение задания обучающимися под наблюдением преподавателя. Обсуждение результатов. Резюме преподавателя.
4. Общее подведение итогов занятия преподавателем и выдача домашнего задания.

Коллоквиум

Форма проверки и оценивание знаний обучающихся в системе образования, представляет собой проводимый по инициативе преподавателя промежуточный контроль знаний по определенным разделам для оценки текущего уровня знаний обучающихся, а также для

повышения знаний. После окончания семинарского занятия обучающемуся выставляются оценки за работу.

Обучающийся при подготовке к практическому занятию может консультироваться с преподавателем и получать от него наводящие разъяснения.

Задания для самостоятельной работы и формы контроля за их выполнением.

Формы самостоятельной работы обучающегося по освоению дисциплины:

1. Усвоение текущего учебного материала;
2. Конспектирование первоисточников;
3. Подготовка по темам для самостоятельного изучения;
4. Написание докладов и реферативных работ по заданным темам;
5. Изучение специальной, методической литературы;
6. Подготовка к экзамену.

Дидактические цели практического 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 обучающихся и преподавателя на занятии, корректировать ранее полученные знания.

Функции практического занятия:

- учебная (углубление, конкретизация, систематизацию знаний, усвоенных во время занятий и в процессе самостоятельной подготовки к семинару);

- 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 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 практического занятия

1. Сообщить тему и план.
2. Предложить для самостоятельного изучения основную и дополнительную литературу.
3. Предоставить устные или письменные советы по подготовке к практическим занятиям.
4. Предоставить обучающимся индивидуальные задания и при необходимости провести консультацию по теме.

Этапы практического занятия. Содержание и характеристика этапа.

Организационная часть

Цель - мобилизовать обучающихся к обучению; активизировать их внимание, создать рабочую атмосферу для проведения занятия.

Мотивация и стимулирование учебной деятельности

Предусматривает формирование потребности изучения конкретного учебного материала, сообщение темы, цели и задач.

Обсуждение проблем, вынесенных на практическое занятие.

Диагностика правильности усвоения обучающимися знаний

Состоит в выяснении причин непонимания определенного элемента содержания учебной информации, неумение или ложности выполнения интеллектуальной или практической деятельности. Осуществляется с помощью серии оперативных и кратковременных контрольных работ, устных фронтальных опросов.

Подведение итогов.

Организация самостоятельной работы обучающихся.

Содержит объяснение содержания задачи, методики его выполнения, краткую аннотацию рекомендованных источников информации, предложения по выполнению индивидуальных заданий.

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 «Языки и методы программирования» включает в себя различные виды деятельности:

- чтение текста (учебника, первоисточника, дополнительной литературы);
- составление плана текста;
- конспектирование текста;
- работа со словарями и справочниками;
- ознакомление с нормативными документами;
- исследовательская работа;
- использование аудио- и видеозаписи;
- работа с электронными информационными ресурсами;
- выполнение тестовых заданий;
- ответы на контрольные вопросы;
- аннотирование, реферирование, рецензирование текста;
- составление глоссария, кроссворда или библиографии по конкретной теме;
- решение вариативных задач и упражнений.

5.5 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

5.5 Методические указания по выполнению контрольной работы

Контрольная работа оформляется в распечатанном или рукописном варианте. Номер варианта выбирается по порядковому номеру списка обучающихся. Контрольная работа с другим номером варианта не зачитываются. Работа выполняется аккуратно, в случае рукописного оформления чтение ее не должно вызывать затруднений.

Контрольная работа должна состоять из титульного листа и основной части. Допускается включение в работу приложений, содержащих таблицы, рисунки, полученные на компьютере. На титульном листе обязательно указывается наименование дисциплины, ФИО обучающегося, группа, вариант задания, ФИО преподавателя. Выполненная и оформленная работа должна быть представлена преподавателю не позднее, чем за 10 дней до начала сессии.

В основной части контрольной работы до решения каждой задачи должны быть представлены собственные данные: вариант задания, формулировка задания, численные значения, соответствующие своему варианту. Далее должно быть представлено решение с расшифровкой формул и последовательности действий. Все вычисления сначала представляются в виде расчетных формул, затем в формулы подставляются численные значения и записывается ответ с указанием единиц измерений (без промежуточных расчетов). Все вычислительные процедуры следует производить с точностью до 0,01.

6. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

№ п/п	Виды учебной работы	Образовательные технологии	Всего часов
1	2	3	4
Семестр 3			
1	<i>Лабораторная работа №1</i>	Самостоятельная работа обучающихся по индивидуальным заданиям лабораторного практикума с ПЭВМ.	2
2	<i>Лабораторная работа №2</i>	Самостоятельная работа обучающихся по индивидуальным заданиям лабораторного практикума с ПЭВМ.	2
3	<i>Лабораторная работа №3</i>	Самостоятельная работа обучающихся по индивидуальным заданиям лабораторного практикума с ПЭВМ.	2
4	<i>Лабораторная работа №14</i>	Самостоятельная работа обучающихся по индивидуальным заданиям лабораторного практикума с ПЭВМ.	2
5	<i>Лабораторная работа №15</i>	Самостоятельная работа обучающихся по индивидуальным заданиям лабораторного практикума с ПЭВМ.	2
6	<i>Лабораторная работа № 16</i>	Самостоятельная работа обучающихся по индивидуальным заданиям лабораторного практикума с ПЭВМ. Оформление лабораторной и защита.	2
Итого часов в 3 семестре:			12
Всего			12

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

7.1. Перечень основной и дополнительной учебной литературы

Основная литература

1. Гумерова, Л. З. Программирование в Lazarus 7 : учебное пособие / Л. З. Гумерова, Г. Н. Аглямзянова. — Красноярск : Научно-инновационный центр, 2019. — 246 с. — ISBN 978-5-6042232-5-3. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/97107.html> . — Режим доступа: для авторизир. пользователей
2. Кауфман, В. Ш. Языки программирования. Концепции и принципы / В. Ш. Кауфман. — 3-е изд. — Саратов : Профобразование, 2024. — 464 с. — ISBN 978-5-4488-0137-2. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/145939.html>
3. Дубровин, В. В. Программирование на C#. Часть 1 : учебное пособие / В. В. Дубровин. — Тамбов : Тамбовский государственный технический университет, ЭБС АСВ, 2017. — 80 с. — ISBN 978-5-8265-1829-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/85942.html> . — Режим доступа: для авторизир. пользователей
4. Тюльпинова, Н. В. Технология алгоритмизации и программирования на языке Pascal : учебное пособие / Н. В. Тюльпинова. — Саратов : Вузовское образование, 2019. — 244 с. — ISBN 978-5-4487-0471-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/80540.html> . — Режим доступа: для авторизир. пользователей
5. Туральчук, К. А. Параллельное программирование с помощью языка C# : учебное пособие / К. А. Туральчук. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2024. — 189 с. — ISBN 978-5-4497-2476-2. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/133969.html>
6. Ефимова, Е. А. Основы программирования на языке Visual Prolog : учебное пособие / Е. А. Ефимова. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2024. — 265 с. — ISBN 978-5-4497-2472-4. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/133962.html>
7. Вязовик, Н. А. Программирование на Java : учебное пособие / Н. А. Вязовик. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 601 с. — ISBN 978-5-4497-0852-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/146383.html>
8. Кисленко, Н. П. Интернет-программирование на PHP : учебное пособие / Н. П. Кисленко. — Новосибирск : Новосибирский государственный архитектурно-строительный университет (Сибстрин), ЭБС АСВ, 2015. — 177 с. — ISBN 978-5-7795-0745-5. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/68769.html>
9. Лубашева, Т. В. Основы алгоритмизации и программирования : учебное пособие / Т. В. Лубашева, Б. А. Железко. — Минск : Республиканский институт профессионального образования (РИПО), 2016. — 379 с. — ISBN 978-985-503-625-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/67689.html>
10. Сузи, Р. А. Язык программирования Python : учебное пособие / Р. А. Сузи. — 4-е изд.

— Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2024. — 350 с. — ISBN 978-5-4497-3351-1. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/142310.html>

Дополнительная литература

1. Лубашева, Т. В. Основы алгоритмизации и программирования : учебное пособие / Т. В. Лубашева, Б. А. Железко. — Минск : Республиканский институт профессионального образования (РИПО), 2016. — 379 с. — ISBN 978-985-503-625-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/67689.html> . — Режим доступа: для авторизир. пользователей
2. Костюкова, Н. И. Комбинаторные алгоритмы для программистов : учебное пособие / Н. И. Костюкова. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2024. — 216 с. — ISBN 978-5-4497-2406-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/133945.html>
3. Зыков, С. В. Введение в теорию программирования. Объектно-ориентированный подход : учебное пособие / С. В. Зыков. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 187 с. — ISBN 978-5-4497-0926-4. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/146342.html>
4. Малиновская, Е. А. Языки программирования. Часть 1 : лабораторный практикум / Е. А. Малиновская, Р. А. Рыскаленко. — Ставрополь : Северо-Кавказский федеральный университет, 2016. — 103 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/69449.html> . — Режим доступа: для авторизир. пользователей
5. Кауфман, В. Ш. Языки программирования. Концепции и принципы / В. Ш. Кауфман. — 3-е изд. — Саратов : Профобразование, 2024. — 464 с. — ISBN 978-5-4488-0137-2. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/145939.html>

7.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

<http://elibrary.ru> - Научная электронная библиотека.

7.3. Информационные технологии, лицензионное программное обеспечение

Лицензионное программное обеспечение	Реквизиты лицензий/ договоров
Антивирус Dr.Web Desktop Security Suite	Лицензионный договор № 621 Срок действия: с 25.09.2025 до 24.09.2026
Цифровой образовательный ресурс IPR SMART	Лицензионный договор № 12873/25П от 02.07.2025 г. Срок действия: с 01.07.2025 г. до 30.06.2026 г.
Бесплатное ПО	
LibreOffice, OpenOffice, МойОфис, Visual Studio Community, Sumatra PDF, 7-Zip, Adobe Acrobat Reader, Lazarus, Virtual box, Visual Studio Code, Blender 3D	

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

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

1. Учебная аудитория для проведения занятий лекционного типа:

Специализированная мебель: Кафедра настольная - 1шт., доска меловая - 1шт., стулья – 65 шт., парты - 34шт. Технические средства обучения, служащие для предоставления учебной информации большой аудитории: Экран на штативе – 1 шт. Проектор – 1 шт. Ноутбук – 1 шт.

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

Специализированная мебель: Кафедра настольная - 1шт., доска меловая - 1шт., стулья – 65 шт., парты - 34шт. Технические средства обучения, служащие для предоставления учебной информации большой аудитории: Экран на штативе – 1 шт. Проектор – 1 шт. Ноутбук – 1 шт.

3. Лаборатория компьютерной графики

Специализированная мебель: Стол преподавательский - 1шт., компьютерные столы - 10шт., парты -7шт., стулья - 24шт., доска меловая - 1шт. Лабораторное оборудование, технические средства обучения, служащие для предоставления учебной информации большой аудитории: Персональный компьютер – 10 шт. Экран настенный рулонный – 1 шт.

4. Лаборатория синергетики и фракталов

Специализированная мебель: Стол преподавательский - 1шт., стул мягкий - 1шт., доска меловая - 1шт., парты - 10шт., компьютерные столы - 11шт., стулья - 21 шт., Лабораторное оборудование, технические средства обучения, служащие для предоставления учебной информации большой аудитории: Персональный компьютер – 11шт. Экран рулонный настенный – 1 шт. Проектор – 1 шт.

5. Помещение для самостоятельной работы

Библиотечно-издательский центр.

Отдел обслуживания печатными изданиями

Специализированная мебель:Рабочие столы на 1 место – 21 шт.Стулья – 55 шт. Набор демонстрационного оборудования и учебно-наглядных пособий, обеспечивающих

тематические иллюстрации: экран настенный – 1 шт.

Проектор – 1 шт. Ноутбук – 1 шт.

Информационно-библиографический отдел.

Специализированная мебель:

Рабочие столы на 1 место - 6 шт. Стулья - 6 шт.

Компьютерная техника с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду ФГБОУ ВО «СевКавГА»:

Персональный компьютер – 1 шт. Сканер – 1 шт. МФУ – 1 шт. Отдел обслуживания электронными изданиями Специализированная мебель:

Рабочие столы на 1 место – 24 шт. Стулья – 24 шт.

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

Интерактивная система - 1 шт. Монитор – 21 шт. Сетевой терминал -18 шт. Персональный компьютер -3 шт. МФУ – 2 шт. Принтер –1шт.

6. Помещение для хранения и профилактического обслуживания учебного оборудования

Специализированная мебель: Шкаф – 1 шт., стул -2 шт., кресло компьютерное – 2 шт., стол угловой компьютерный – 2 шт., тумбочки с ключом – 2 шт. Учебное пособие (персональный компьютер в комплекте) – 2 шт.

8.2. Требования к оборудованию рабочих мест преподавателя и обучающихся

Рабочее место преподавателя, оснащенное компьютером с доступом в Интернет.

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 нет

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 Языки и методы программирования

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

Языки и методы программирования

Компетенции, формируемые в процессе изучения дисциплины

Индекс	Формулировка компетенции
ПК – 3	Способен осуществлять программную реализацию алгоритмов при разработке прикладных информационных систем.

2. Этапы формирования компетенции в процессе освоения дисциплины

Основными этапами формирования указанных компетенций при изучении обучающимися 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	Формируемы компетенции и коды
	ПК-3
Раздел 1. Основы программирования	+
Раздел 2. Интегрированная среда разработки Lazarus	+
Раздел 3. Введение в язык программирования Python	+

3. Показатели, критерии и средства оценивания компетенций, формируемых в процессе изучения дисциплины

ПК-3 Способен осуществлять программную реализацию алгоритмов при разработке прикладных информационных систем.

Индикаторы достижения компетенций	Критерии оценивания результатов обучения				Средства оценивания результатов обучения	
	неудовлетв	удовлетв	хорошо	отлично	Текущий контроль	Промежуточная аттестация
ПК-3.1 Обладает знаниями в области программирования и современных компьютерных технологий при разработке прикладных информационных систем. Способен осуществлять оптимизацию алгоритмов при создании прикладных программ или прикладных комплексов	Не обладает знаниями в области программирования и современных компьютерных технологий при разработке прикладных информационных систем. Не способен осуществлять оптимизацию алгоритмов при создании прикладных программ или прикладных комплексов	Частично обладает знаниями в области программирования и современных компьютерных технологий при разработке прикладных информационных систем. Частично способен осуществлять оптимизацию алгоритмов при создании прикладных программ или прикладных комплексов	Обладает с неточностями знаниями в области программирования и современных компьютерных технологий при разработке прикладных информационных систем. Способен осуществлять оптимизацию алгоритмов при создании прикладных программ или прикладных комплексов	Обладает знаниями в области программирования и современных компьютерных технологий при разработке прикладных информационных систем. Способен осуществлять оптимизацию алгоритмов при создании прикладных программ или прикладных комплексов	контрольные вопросы, индивидуальные задания к лабораторным работам., тестирование, контрольная работа	Зачет с оценкой, экзамен
ПК-3.2 Анализирует стандартные технические требования к проектированию программного обеспечения.	Не анализирует стандартные технические требования к проектированию программного обеспечения.	Частично анализирует стандартные технические требования к проектированию программного обеспечения.	Умеет с неточностями анализировать стандартные технические требования к проектированию программного обеспечения.	Анализирует стандартные технические требования к проектированию программного обеспечения.	контрольные вопросы, индивидуальные задания к лабораторным работам., тестирование, контрольная работа	Зачет с оценкой, экзамен
ПК-3.3 Способен осуществлять разработку технических требований к	Не способен осуществлять разработку технических требований	Частично способен осуществлять разработку технических требований	Способен с неточностями осуществлять	Способен осуществлять разработку технических требований к	контрольные вопросы, индивидуальные	Зачет с оценкой, экзамен

программному обеспечению, при его создании	к программному обеспечению, при его создании	к программному обеспечению, при его создании	разработку технических требований к программному обеспечению, при его создании	программному обеспечению, при его создании	задания к лабораторным работам., тестирование, контрольная работа	
--	--	--	--	--	---	--

4. Комплект контрольно-оценочных средств по дисциплине

Вопросы экзамену

по дисциплине «Языки и методы программирования»

1. Этапы решения задач с использованием ЭВМ
2. Понятие алгоритма. Подходы к определению алгоритма. Свойства алгоритма. Способы записи алгоритма
3. Понятие исполнителя. Система команд исполнителя
4. Основные алгоритмические конструкции
5. Понятие о синтаксисе и семантике формального языка
6. Классификация языков программирования
7. Выражение. Приоритеты в выражении
8. Стандартные функции
9. Структуры данных. Стандартные типы данных
10. Типы данных, определяемые программистом
11. Операторы ввода и вывода в Object Pascal
12. Условный оператор в Object Pascal
13. Оператор множественного выбора в Object Pascal
14. Массивы в Object Pascal
15. Записи в Object Pascal
16. Бинарный поиск в массиве
17. Сортировка данных методом обмена в Object Pascal
18. Сортировка данных методом выбора в Object Pascal
19. Процедуры в Object Pascal
20. Общая структура модуля в Object Pascal
21. Типизированные и нетипизированные указатели
22. Односвязные линейные и циклические списки
23. Инкапсуляция и свойства объекта
24. Виртуальные методы
25. Понятие о визуальном проектировании приложений
26. Экранная форма в Lazarus и его свойства
27. Методы вычерчивания графических примитивов в Object Pascal
28. Битовые образы в Object Pascal

- 29.Создание модуля компонента в ObjectPascal14
- 30.Модель базы данных в ObjectPascal
- 31.История создания языка. Python2 и Python3.
- 32.Среды разработки
- 33.Области применимости
- 34.Ввод-вывод в Python
- 35.Арифметические операции в Python
- 36.Типы данных Python
- 37.Нерегулярные структуры Python
- 38.Модульный аспект Python
- 39.Специфика разработки программных средств.
- 40.Жизненный цикл программного средства. Понятие качества программного средства.
- 41.Обеспечение надежности - основной мотив разработки программного средства.
- 42.Методы борьбы со сложностью.
- 43.Обеспечение точности перевода.
- 44.Преодоление барьера между пользователем и разработчиком.
- 45.Обеспечение контроля правильности принимаемых решений.
- 46.Порядок разработки программного модуля.
- 47.Структурное программирование и пошаговая детализация.
- 48.Понятие о псевдокоде.
- 49.Контроль программного модуля.
- 50.Стратегия проектирования тестов. Заповеди отладки.
- 51.Автономная отладка и тестирование программного модуля.
- 52.Комплексная отладка и тестирование программного средства.
- 53.Документация, создаваемая в процессе разработки программных средств
- 54.Пользовательская документация программных средств.
- 55.Документация по сопровождению программных средств.
- 56.Назначение аттестации программного средства.
- 57.Испытания и оценка качества программного средства.
- 58.Виды испытаний и методы оценки качества программного средства.

Задачи к экзамену

1. Даны 3 различных массива целых чисел (размер каждого не превышает 15). В каждом

- массиве найти сумму элементов и среднеарифметическое значение.
2. Вычислить площадь правильного шестиугольника со стороной, a , используя подпрограмму вычисления площади треугольника.
 3. Пользователь вводит две стороны трех прямоугольников. Вывести их площади.
 4. Даны катеты двух прямоугольных треугольников. Написать функцию вычисления длины гипотенузы этих треугольников. Сравнить и вывести какая из гипотенуз больше, а какая меньше.
 5. Преобразовать строку так, чтобы буквы каждого слова в ней были отсортированы по алфавиту.
 6. Даны две дроби A/B и C/D (A, B, C, D — натуральные числа). Составить программу деления дроби на дробь. Ответ должен быть несократимой дробью. Использовать подпрограмму алгоритма Евклида для определения НОД.
 7. Задана окружность $(x-a)^2 + (y-b)^2 = R^2$ и точки $P(p_1, p_2)$, $F(f_1, f_1)$, $L(l_1, l_2)$. Выяснить и вывести на экран, сколько точек лежит внутри окружности. Проверку, лежит ли точка внутри окружности, оформить в виде процедуры.
 8. Даны две дроби A/B и C/D (A, B, C, D — натуральные числа). Составить программу вычитания из первой дроби второй. Ответ должен быть несократимой дробью. Использовать подпрограмму алгоритма Евклида для определения НОД.
 9. Напишите программу, которая выводит в одну строчку все делители переданного ей числа, разделяя их пробелами.
 10. Составить программу нахождения наибольшего общего делителя (НОД) и наименьшего общего кратного (НОК) двух натуральных чисел $\text{НОК}(A, B) = (A*B)/\text{НОД}(A, B)$. Использовать подпрограмму алгоритма Евклида для определения НОД.
 11. Даны числа X, Y, Z, T — длины сторон четырехугольника. Вычислить его площадь, если угол между сторонами длиной X и Y — прямой. Использовать две подпрограммы для вычисления площадей: прямоугольного треугольника и прямоугольника.
 12. Напишите программу, которая переводит переданное ей неотрицательное целое число в 10-значный восьмеричный код, сохранив лидирующие нули.
 13. Найти все натуральные числа, не превосходящие заданного n , которые делятся на каждую из своих цифр.
 14. Ввести одномерный массив A длиной m . Поменять в нём местами первый и последний элементы. Длину массива и его элементы ввести с клавиатуры. В программе описать процедуру для замены элементов массива. Вывести исходные и полученные массивы.
 15. Из заданного числа вычли сумму его цифр. Из результата вновь вычли сумму его цифр и т. д. Через сколько таких действий получится ноль?
 16. Даны 3 различных массива целых чисел. В каждом массиве найти произведение элементов и среднеарифметическое значение.
 17. На отрезке $[100, N]$ ($210 < N < 231$) найти количество чисел, составленных из цифр a, b, c .
 18. Составить программу, которая изменяет последовательность слов в строке на обратную.
 19. Дана матрица $B[N, M]$. Найти в каждой строке матрицы максимальный и минимальный элементы и поменять их с первым и последним элементами строки соответственно.
 20. Дана целая квадратная матрица n -го порядка. Определить, является ли она магическим квадратом, т. е. такой матрицей, в которой суммы элементов во всех строках и столбцах одинаковы.
 21. Дана прямоугольная матрица $A[N, N]$. Переставить первый и последний столбцы

местами и вывести на экран.

22. Дана квадратная матрица $A[N, N]$, Записать на место отрицательных элементов матрицы нули, а на место положительных — единицы. Вывести на печать нижнюю треугольную матрицу в общепринятом виде.
23. Упорядочить по возрастанию элементы каждой строки матрицы размером $n \times m$.
24. Дана целочисленная квадратная матрица. Найти в каждой строке наибольший элемент и в каждом столбце наименьший. Вывести на экран.
25. Дана действительная квадратная матрица порядка N (N — нечетное), все элементы которой различны. Найти наибольший элемент среди стоящих на главной и побочной диагоналях и поменять его местами с элементом, стоящим на пересечении этих диагоналей.
26. Задана квадратная матрица. Получить транспонированную матрицу (перевернутую относительно главной диагонали) и вывести на экран.
27. В данной действительной квадратной матрице порядка n найти наибольший по модулю элемент. Получить квадратную матрицу порядка $n - 1$ путем отбрасывания из исходной матрицы строки и столбца, на пересечении которых расположен элемент с найденным значением.
28. Расположить столбцы матрицы $D[M, N]$ в порядке возрастания элементов k -й строки ($1 \leq k \leq M$).

СЕВЕРО-КАВКАЗСКАЯ ГОСУДАРСТВЕННАЯ АКАДЕМИЯ

Кафедра «Математика»

20__ - 20__ учебный год

Экзаменационный билет № 1

по дисциплине Языки и методы программирования

для обучающихся по направлению подготовки 01.03.02 Прикладная математика и информатика

1. Ввод-вывод в Python

2. Назначение аттестации программного средства.

3. Написать программу, которая бы вычисляла заданное арифметическое выражение при заданных переменных. Ввод переменных осуществляется с клавиатуры. Вывести результат с 2-мя знаками после запятой.

$$a = \frac{\sqrt{|x-1|} - \sqrt[3]{|y|}}{1 + \frac{x^2}{2} + \frac{z^2}{4}}$$

Вычислить.

$$x = 2; y = 6; z = 7.$$

и $b = x(\operatorname{arctg}Z + e)^{-(y+3)}$; если

Зав. кафедрой

Кочкаров А.М.

Контрольные вопросы

по дисциплине «Языки и методы программирования»

Вопросы к разделу 1,2

1. Этапы решения задач с использованием ЭВМ
2. Понятие алгоритма. Подходы к определению алгоритма. Свойства алгоритма. Способы записи алгоритма
3. Понятие исполнителя. Система команд исполнителя
4. Основные алгоритмические конструкции
5. Понятие о синтаксисе и семантике формального языка
6. Классификация языков программирования
7. Выражение. Приоритеты в выражении
8. Стандартные функции
9. Структуры данных. Стандартные типы данных
10. Типы данных, определяемые программистом
11. Операторы ввода и вывода в Object Pascal
12. Условный оператор в Object Pascal
13. Оператор множественного выбора в Object Pascal
14. Массивы в Object Pascal
15. Записи в Object Pascal
16. Бинарный поиск в массиве
17. Сортировка данных методом обмена в Object Pascal
18. Сортировка данных методом выбора в Object Pascal
19. Процедуры в Object Pascal
20. Общая структура модуля в Object Pascal
21. Типизированные и нетипизированные указатели
22. Односвязные линейные и циклические списки
23. Инкапсуляция и свойства объекта
24. Виртуальные методы
25. Понятие о визуальном проектировании приложений
26. Экранная форма в Lazarus и ее свойства
27. Методы вычерчивания графических примитивов в Object Pascal
28. Битовые образы в Object Pascal

29.Создание модуля компонента в ObjectPascal14

30.Модель базы данных в ObjectPascal

Вопросы к разделу 3.

1. История создания языка. Python2 и Python3.
2. Среды разработки
3. Области применимости
4. Ввод-вывод в Python
5. Арифметические операции в Python
6. Типы данных Python
7. Нерегулярные структуры Python
8. Модульный аспект Python
9. Специфика разработки программных средств.
10. Жизненный цикл программного средства. Понятие качества программного средства.
11. Обеспечение надежности - основной мотив разработки программного средства.
Методы борьбы со сложностью.
12. Обеспечение точности перевода.
13. Преодоление барьера между пользователем и разработчиком.
14. Обеспечение контроля правильности принимаемых решений.
15. Порядок разработки программного модуля.
16. Структурное программирование и пошаговая детализация.
17. Понятие о псевдокоде.
18. Контроль программного модуля.
19. Стратегия проектирования тестов. Заповеди отладки.
20. Автономная отладка и тестирование программного модуля.
21. Комплексная отладка и тестирование программного средства.
22. Документация, создаваемая в процессе разработки программных средств
23. Пользовательская документация программных средств.
24. Документация по сопровождению программных средств.
25. Назначение аттестации программного средства.
26. Испытания и оценка качества программного средства.
27. Виды испытаний и методы оценки качества программного средства.

Комплект тестовых заданий

по дисциплине «Языки и методы программирования»

1. Точно определённая система понятных исполнителю предписаний, формальное выполнение которых позволяет получить решение задачи для любого допустимого набора исходных данных за конечное число шагов называют _____.

2. Компилятор это...

-язык программирования

-программа, которая переводит исходный код на машинный язык и создает выполняемую программу

-программа, которая пошагово переводит исходный код на машинный язык и сразу же выполняет его.

3. Язык низкого уровня это...

-язык программирования, чей синтаксис наиболее приближен к машинному языку

-программа, которая переводит исходный код на машинный язык и создает выполняемую программу

-программа, которая пошагово переводит исходный код на машинный язык и сразу же выполняет его

4. Наиболее быстро выполняется программа, написанная на языке _____.

5. Исполняемый файл (программа) будет иметь наибольший размер, если программа создавалась на...

-языке низкого уровня

-языке высокого уровня

-объектно-ориентированном языке

-визуальных средах разработки программ

6. Исполняемый файл (программа) будет иметь наименьший размер, если программа создавалась на...

-машинном языке

-языке высокого уровня

-объектно-ориентированном языке

-визуальных средах разработки программ

7. Проектом является:

-полученная в результате компиляции программа

- совокупность файлов разных форматов, из которых строится программа
- название полученного в результате компиляции выполняемого файла

8. Что произойдет в результате выполнения кода?

ShowMessage('Hello!');

- слово Hello! появится в заголовке окна
- произойдет ошибка в программе
- слово Hello! выйдет в сообщении в отдельном окне
- слово Hello! выйдет в сообщении в окне формы

9. В результате выполнения кода ShowMessage('Hello, all!'); в заголовке окна появится _____.

10. За что отвечает свойство Caption формы?

- за надпись на форме
- за надпись в заголовке окна формы
- за надпись на всех кнопках формы

11. Свойство Caption кнопки отвечает за _____.

12. Укажите, из каких окон состоит среда Lazarus:

- дерево объектов
- конструктор формы
- конструктор компонентов
- редактор кода
- объектный инспектор

13. Кнопку мы можем установить в окне_____.

14. Где мы можем найти кнопку, которую хотим установить на форму?

- в главном меню Lazarus
- на палитре компонентов
- в объектном инспекторе

15. Язык Python:

Выберите верные варианты

- является компилируемым языком
- является интерпретируемым языком
- является низкоуровневым языком программирования
- является высокоуровневым языком программирования
- является универсальным языком программирования

16. Язык Python поддерживает следующие парадигмы программирования:

Выберите верные варианты

- автоматное программирование
- структурное программирование
- модульное программирование
- процедурное программирование
- логическое программирование

17. Язык Python поддерживает следующие парадигмы программирования:

Выберите верные варианты

- программирование в ограничениях
- аспектно-ориентированное программирование
- объектно-ориентированное программирование
- структурное программирование
- модульное программирование

18. Язык Python поддерживает следующие стили программирования:

Выберите верные варианты

- процедурное программирование
- декларативное программирование
- объектно-ориентированное программирование
- функциональное программирование
- аспектно-ориентированное программирование

19. Язык Python поддерживает следующие парадигмы программирования:
Выберите верные варианты

- объектно-ориентированное программирование
- модульное программирование
- императивное программирование
- функциональное программирование
- логическое программирование

20. Язык Python поддерживает следующие парадигмы программирования:

Выберите верные варианты

- модульное программирование
- структурное программирование

- аспектно-ориентированное программирование
- программирование в ограничениях
- процедурное программирование

21. Python поддерживает следующие простые скалярные типы:

Выберите верные варианты

- целое
- беззнаковое целое
- булево (логическое)
- число с плавающей точкой
- строка

22. Привести переменную x к типу числа с плавающей точкой можно следующим способом:

- (double)x
- x.double()
- x.__double__()
- double(x)
- никаким из вышеперечисленных

23. Тип переменной во время выполнения скрипта определяется по следующим правилам:

- тип переменной явно указывается при определении переменной и не изменяется в процессе выполнения скрипта
- тип переменной явно указывается при определении переменной и изменяется только при приведении этой переменной к другому типу
- тип переменной определяется типом первого значения, которое было ей присвоено, и далее не изменяется
- тип переменной изменяется при приведении этой переменной к другому типу, а также может изменяться в зависимости от контекста использования этой переменной
- тип переменной изменяется при присваивании, но не может изменяться в зависимости от контекста использования этой переменной

24. Тип переменной x можно узнать следующим способом:

- type(x)
- x.type()
- x.__class__ == имя_типа
- x.__class__ is имя_типа
- x.__class__ == "имя типа"

25. Из приведенных ниже высказываний укажите истинное:

- перед использованием переменной она должна быть определена с указанием типа и инициализирована каким-либо значением
- перед использованием переменной она должна быть определена с указанием типа
- перед использованием переменной она должна быть инициализирована каким-либо значением
- переменные не обязаны быть инициализированы каким-либо значением перед использованием, а тип переменной определяется в зависимости от контекста
- перед использованием переменной она будет автоматически проинициализирована значением по умолчанию

26. Укажите результат выполнения скрипта:

```
foo = (1,)
```

```
bar = foo
```

```
bar += (1,)
```

```
print (foo)
```

ответ _____

27. Укажите результат выполнения скрипта:

```
foo = 0x01
```

```
bar = foo
```

```
bar += 0x01
```

```
print (foo)
```

ответ _____

28. Укажите результат выполнения скрипта:

```
foo = {'1':1}
```

```
bar = foo
```

```
bar['2']=1
```

```
print (foo)
```

ответ _____

29. Укажите результат выполнения скрипта:

```
foo = [1]
```

```
bar = foo
```

```
bar += [1]
```

```
print (foo)
```

ответ _____

30. Укажите результат выполнения скрипта:

```
foo = [1]
bar = foo
bar = bar + [1]
print (foo)
```

ответ _____

Комплект заданий для лабораторной работы

по дисциплине «Языки и методы программирования»

Лабораторная работа № 1

Тема: Программирование алгоритмов линейной структуры

Цель работы – овладение практическими навыками разработки и программирования вычислительного процесса линейной структуры и навыками по отладке и тестированию программ

Задание к работе

1. Вычислить значения переменных, указанных в таблице 1 (вариант соответствует порядковому номеру студента в подгруппе), по заданным расчетным формулам и наборам исходных данных. На экран вывести значения вводимых исходных данных и результаты вычислений, сопровождая вывод наименованиями переменных

Таблица 1.

Вариант задания	Расчетные формулы	Значения исходных данных
1.	$a = 2\cos(x-\pi/6)/0,5+\sin^2y;$ $b = 1+(Z^2/(3+Z^2/5))$	$x = 1,426; y = -1,220$ $Z = 3,5$
2.	$\gamma = e^{y/x} - \sqrt{y/x} ;$ $\varphi = (y-x)(y-z)/(y-x)/(1+(y-x)^2)$	$x = 1,825; y = 18,225$ $z = -3,298$
3.	$s = 1+x+x^2/2!+x^3/3!+x^4/4!;$ $k = x(\sin x^3+\cos^2y)$	$x = 0,335$ $y = 0,025$
4.	$y = e^{-bt} \sin(at + b) - \sqrt{ bt + a }$ $s = b \sin(at^2\cos 2t)-1$	$a = -0,5; b = 1,7$ $t = 0,44$
5.	$w = \sqrt{x^2 + b} - b^2 \sin^3(x + a)/x;$ $v = \cos^2 x^3 - x/\sqrt{a^2 + b^2}$	$a = 1,5; b = 15,5$ $x = -2,9$
6.	$r = x^3 \operatorname{tg}^2(x + b)^2 + a\sqrt{x + b};$ $g = (bx^2 - a)/(e^{ax} - 1)$	$a = 16,5; b = 3,4$ $x = 0,61$
7.	$R = x^2(x+1)/b - \sin^2(x+a);$ $S = \sqrt{xb/a + \cos^2(x + b)^3}$	$a = 0,7; b = 0,05$ $x = 0,5$
8.	$y = \sin^3(x^2 + a)^2 - \sqrt{x/b}$ $Z = x^2/a + \cos(x+b)^3$	$a = 1,1$ $b = 0,004; x = 0,2$
9.	$f = \sqrt{mtgt + c \sin t };$ $Z = m\cos(btsint)+c$	$m = 2; c = -1$ $t = 1,2; b = 0,7$
10.	$y = btg^2x - a/\sin^2(x/a)$ $d = ae^{\sqrt{a}} \cos(bx/a)$	$a = 3,2; b = 17,5$ $x = -4,8$
11.	$f = \ln(a+x^2)+\sin^2(x/b);$ $Z = e^{-cx} (x + \sqrt{x + a})/(x - \sqrt{ x - b })$	$a = 10,2; b = 17,5$ $x = -4,8$

12.	$y = (a^{2x} + b^{-x} \cos(a+b)x)/(x+1);$	$a = 0,3; b = 0,9$
	$R = \sqrt{x^2 + b} - b^2 \sin^3(x+a)/x$	$x = 0,61$
13.	$Z = \sqrt{ax \sin 2x + e^{-2x}(x+b)};$	$a = 0,5; b = 3,1$
	$w = \cos^2 x^3 - x/\sqrt{a^2 + b^2}$	$x = 1,4$
14.	$u = (a^{2x} + e^{-x} \cos bx)/(bx - e^{-x} \sin bx + 1);$	$a = 0,5; b = 2,9$
	$f = e^{2x} \ln(a+x) - b^4 \ln(b-x)$	$x = 0,3$
15.	$Z = (\sin x / \sqrt{1 + m^2 \sin^2 x}) - cm \ln mx$	$m = 0,7; c = 2,1; x = 1,7$
	$C = e^{-BX} \sqrt{x+1} + e^{-bx} \sqrt{x+1,5}$	$a = 0,5; b = 1,08$

2. Модифицировать программу, а затем выполнить ее на ЭВМ таким образом, чтобы вывод вычисленных значений переменных осуществлялся в соответствии со следующей разметкой строк:

```
*****
* РЕЗУЛЬТАТЫ ВЫЧИСЛЕНИЙ*
*****
<пропуск 1-ой строки>
<идентификатор>
****
<идентификатор>=...
****
```

Лабораторная работа №2

Тема: Программирование алгоритмов разветвляющейся и циклической структуры

Цель работы – овладение практическими навыками разработки и программирования вычислительного процесса разветвляющейся и циклической структур; получение дальнейших навыков по отладке и тестированию программ.

Задание к работе

1. Вычислить значение функции, заданной в таблице 2 (в соответствии с вариантом задания). Осуществить выход значений исходных данных и результатов вычислений значений функции, сопровождая выход наименованиями переменных.

Таблица 2

Вариант	Функция	Условие	Исходные данные	Диапазон и шаг изменения аргумента
1	$Y = \begin{cases} at^2 \text{ int} \\ 1 \\ e^{at} \cos bt \end{cases}$	$1 \leq t \leq 2$ $t < 1$ $t > 2$	$a = -0.5$ $b = 2$	$t \in [0;3]$ $\Delta t = 0.15$

2	$Y = \begin{cases} \pi x^2 - 7/x^2 \\ ax^3 + 7\sqrt{x} \\ \lg(x + 7\sqrt{x}) \end{cases}$	$\begin{cases} x < 1.3 \\ x = 1.3 \\ x > 1.3 \end{cases}$	$a = 1.5$	$\begin{aligned} x &\in [0.8; 2] \\ \Delta x &= 0.1 \end{aligned}$
3	$\omega = \begin{cases} ax^2 + bx + c \\ a/x + \sqrt{x^2 + 1} \\ (a + bx)/\sqrt{x + a} \end{cases}$	$\begin{cases} x < 1.2 \\ x = 1.2 \\ x > 1.2 \end{cases}$	$\begin{aligned} a &= 2.8 \\ b &= -0.3 \\ c &= 4 \end{aligned}$	$\begin{aligned} x &\in [1; 2] \\ \Delta x &= 0.05 \end{aligned}$
4	$Q = \begin{cases} \pi x^x - 7/x^2 \\ ax^3 + 7\sqrt{x} \\ \ln(x + 7\sqrt{x+a}) \end{cases}$	$\begin{cases} x < 1.4 \\ x = 1.4 \\ x > 1.4 \end{cases}$	$a = 1.65$	$\begin{aligned} x &\in [0.7; 2] \\ \Delta x &= 0.2 \end{aligned}$
5	$Y = \begin{cases} 1.5 \cos^2 x \\ 1.8ax \\ (x-2)^2 + 6 \\ 3 \operatorname{tg} x \end{cases}$	$\begin{cases} x < 1 \\ x = 1 \\ 1 < x < 2 \\ x > 2 \end{cases}$	$a = 2.3$	$\begin{aligned} x &\in [0.2; 2] \\ \Delta x &= 0.2 \end{aligned}$
6	$\omega = \begin{cases} x\sqrt{x-a} \\ x \sin ax \\ e^{-ax} \cos ax \end{cases}$	$\begin{cases} x > a \\ x = a \\ x < a \end{cases}$	$a = 2.5$	$\begin{aligned} x &\in [1; 5] \\ \Delta x &= 0.5 \end{aligned}$
7	$Q = \begin{cases} bx - \lg bx \\ 1 \\ bx + \lg bx \end{cases}$	$\begin{cases} bx < 1 \\ bx = 1 \\ bx > 1 \end{cases}$	$b = 1.5$	$\begin{aligned} x &\in [0.1; 1] \\ \Delta x &= 0.1 \end{aligned}$
8	$Y = \begin{cases} \sin \lg x \\ \cos^2 x \end{cases}$	$\begin{cases} x > 3.5 \\ x \leq 3.5 \end{cases}$		$\begin{aligned} x &\in [0.1; 1] \\ \Delta x &= 0.1 \end{aligned}$
9	$F = \begin{cases} \lg(x+1) \\ \sin^2 ax \end{cases}$	$\begin{cases} x > 1 \\ x \leq 1 \end{cases}$	$a = 20.3$	$\begin{aligned} x &\in [0.5; 2] \\ \Delta x &= 0.1 \end{aligned}$
10	$Z = \begin{cases} \ln^3 x + x^2 / \sqrt{x+t} \\ \sqrt{x+t} + 1/x \\ \cos x + t \sin^2 x \end{cases}$	$\begin{cases} x < 0.5 \\ x = 0.5 \\ x > 0.5 \end{cases}$	$t = 2.2$	$\begin{aligned} x &\in [0.2; 2] \\ \Delta x &= 0.2 \end{aligned}$
11	$S = \begin{cases} (a+b)/(e^x + \cos x) \\ (a+b) * (x+1) \\ e^x + \sin x \end{cases}$	$\begin{cases} x < 2.8 \\ 2.8 \leq x < 6 \\ x \geq 6 \end{cases}$	$\begin{aligned} a &= 2.6 \\ b &= -0.39 \end{aligned}$	$\begin{aligned} x &\in [0; 7] \\ \Delta x &= 0.5 \end{aligned}$

12	$Y = \begin{cases} a \lg x + \sqrt{ x } \\ 2a \cos x + 3x^2 \end{cases}$	$\begin{cases} x > 1 \\ x \leq 1 \end{cases}$	$a = 0.9$	$\begin{aligned} x &\in [0.8; 2] \\ \Delta x &= 0.1 \end{aligned}$
13	$\omega = \begin{cases} a/i + bi^2 + c \\ i \\ ai + bi^3 \end{cases}$	$\begin{cases} i < 4 \\ 4 \leq i \leq 6 \\ i > 6 \end{cases}$	$\begin{aligned} a &= 2.1 \\ b &= 1.8 \\ c &= -20.5 \end{aligned}$	$\begin{aligned} i &\in [0; 12] \\ \Delta i &= 1 \end{aligned}$
14	$z = \begin{cases} a \sin(i^2 + 1/n) \\ \cos(i + 1/n) \end{cases}$	$\begin{cases} \sin(i^2 + 1/n) > 0 \\ \sin(i^2 + 1/n) < 0 \end{cases}$	$\begin{aligned} a &= 0.3 \\ n &= 10 \end{aligned}$	$\begin{aligned} i &\in [1; 10] \\ \Delta i &= 1 \end{aligned}$
15	$\omega = \begin{cases} \sqrt{at^2 + bsint + 1} \\ at + b \\ \sqrt{at^2 + bcost + 1} \end{cases}$	$\begin{cases} t < 0.1 \\ t = 0.1 \\ t > 0.1 \end{cases}$	$\begin{aligned} a &= 2.5 \\ b &= 0.4 \end{aligned}$	$\begin{aligned} t &\in [-1; 1] \\ \Delta t &= 0.2 \end{aligned}$

- Запустить программу на ЭВМ и протестировать все ветви ее алгоритма.
- Модифицировать программу таким образом, чтобы многократно вычислялось значение функции при изменении аргумента в указанном диапазоне и с заданным шагом (табл.1).
- Организовать вывод значений аргументов и вычисленных значений функции в виде следующей таблицы:

ТАБЛИЦА ФУНКЦИИ Y(X)

X	Y
...	...
...	...

- Выполнить модифицированную программу на компьютере

Лабораторная работа №3

Тема: Обработка матриц

Цель работы – овладение навыками алгоритмизации и программирования структур с вложенными циклами, навыками использования приемов программирования во вложенных циклах, способами ввода и вывода матриц.

Задание к работе

- Обработать на компьютере матрицу в соответствии с вариантом задания, указанного в таблице 3. Вывести на экран результаты и исходную матрицу в общепринятом виде.
- Проверить правильность выполнения программы с помощью тестового варианта

Таблица 3

Вариант	Имя матрицы и	Действия	Условия и ограничения

	размеры		
1	$A(10,15)$	Вычислить и запомнить сумму и число положительных элементов каждого столбца матрицы. Результаты отпечатать в виде двух строк.	$a_{ij} > 0$
2	$AI(N,M)$	Вычислить и запомнить сумму и числа элементов каждой строки матрицы. Результаты отпечатать в виде двух столбцов.	$N \leq 20$ $M \leq 15$
3	$B(N,N)$	Вычислить и запомнить сумму и число элементов матрицы, находящихся под главной диагональю и на ней.	$N \leq 12$
4	$C(N,N)$	Вычислить сумму и число положительных элементов матрицы, находящихся над главной диагональю.	$C_{ij} > 0$ $N \leq 12$
5	$D(K,K)$	Записать на место отрицательных элементов матрицы нули и вывести ее в общепринятом виде	$K \leq 10$
6	$D(10,10)$	Записать на место отрицательных элементов матрицы нули, а на место положительных единицы. Вывести нижнюю треугольную матрицу в общепринятом виде.	
7	$F(N,M)$	Найти в каждой строке матрицы максимальный и минимальный элементы и поместить их на место первого и последнего элемента строки соответственно. Матрицу вывести в общественном виде.	$N \leq 20$ $M \leq 10$
8	$FI(10,8)$	Транспонировать матрицу и вывести элементы главной диагонали и диагонали, расположенной под главной. Результаты разместить в двух строках.	
9	$N(10,10)$	Для целочисленной матрицы найти для каждой строки число элементов, кратных пяти, и наибольший из полученных результатов.	$n_{ij} / 5 * 5 - n_{ij}$
10	$M(10,10)$	Из положительных элементов М сформировать матрицу К(10,КМАХ), располагая их в строках матрицы подряд, где КМАХ-максимальное число положительных элементов строки матрицы М. Записать нули на место отсутствующих элементов. Отпечатать обе матрицы в общепринятом виде.	
11	$P(N,N)$	Найти в каждой строке наибольший элемент и поменять его местами с элементом главной диагонали. Отпечатать полученную матрицу в общепринятом виде.	$N \leq 15$
12	$R(K,N)$	Найти наибольший и наименьший элементы матрицы и поменять их местами.	$K \leq 20$ $N \leq 10$

13	$S(25,8)$	Ввести исходные данные в первые 24 строки и первые 7 столбцов. Вычислить среднее арифметическое значение элементов каждой строки и записать его в 8-й столбец, а также среднее арифметическое каждого столбца и записать его в 25-ю строку. Отпечатать полученную матрицу в общепринятом виде.	
14	$T(N,M)$	Найти строку с наибольшей и наименьшей суммой элементов. Вывести найденные строки и суммы элементов.	$N \leq 20$ $M \leq 15$
15	$V(15,10)$	Упорядочить по возрастанию элементы каждой строки матрицы. Отпечатать полученную матрицу в общепринятом виде.	

Лабораторная работа № 4

Тема: Программирование с использованием подпрограмм пользователя

Цель работы – овладение навыками алгоритмизации и программирования задач с использованием подпрограмм пользователя различных разновидностей, владение навыками написания подпрограмм и обращения к ним, выбора параметров и т. д.

Задание к работе

1. Выполнить на компьютере программу, использующую подпрограмму-процедуру в соответствии с номером варианта, указанным в таблице 4.

Таблица 4

Вариант	Условие задачи	Примечания
1	Вычислить $z = (s1 + s2)/(k1k2)$, где $s1$ и $k1$ - сумма и количество положительных элементов массива $X(N)$; $s2$ и $k2$ - сумма и количество положительных элементов массива $Y(M)$.	$M \leq 100$ $N \leq 100$
2	Вычислить $z = (e^{s1} + e^{s2})/(k1k2)$, где $s1$ и $k1$ - сумма и количество положительных элементов массива $X(100)$; $s2$ и $k2$ - сумма и количество положительных элементов массива $Y(80)$.	Обе суммы вычислять в одной подпрограмме
3	Вычислить и запомнить суммы, положительных элементов каждой строки матриц $A(10,20)$ и $B(15,10)$.	
4	Найти наибольшие элементы и их порядковые номера в массивах $X(N)$ и $Y(M)$.	$M \leq 80$ $N \leq 70$
5	Переписать положительные элементы массива матрицы $X(100)$ и $Y(80)$ в массив Z подряд	Запись в массив Z осуществлять в процедуре
6	Найти наименьшие элементы и номера строк и столбцов, в которых они расположены, для матриц	

	A(10,15) и B(15,12).	
7	$Z = \left(\sum_{i=1}^{40} \sin X_i + \sum_{i=1}^{50} \cos Y_i \right) / \sum_{i=1}^{40} x_i$ Вычислить $Z = (\sum_{i=1}^{40} \sin X_i + \sum_{i=1}^{50} \cos Y_i) / \sum_{i=1}^{40} x_i$, где X_i и Y_i заданы массивами.	Все суммы вычислять в одной подпрограмме
8	Вычислить $Z = (X_{\max} - Y_{\min}) / 2$, где X_{\max} - максимальный элемент массива X(50); Y_{\min} - минимальный элемент массива Y(40).	X_{\max} и Y_{\min} вычислять в одной подпрограмме
9	Вычислить и запомнить количество отрицательных элементов каждого столбца для матриц A(10,10), B(15,20).	
10	Вычислить суммы и количества элементов, находящихся в интервале от а до b для матриц X(10,8) и Y(10,12).	
11	Найти средние значения и стандартные отклонения для элементов массивов X(N), Y(M).	$N \leq 100$ $M \leq 100$
12	Преобразовать массивы X(50) и Y(60), расположив в них подряд только положительные элементы. Вместо остальных элементов записать нули.	
13	Вычислить $Z = (V_1 + V_2 + V_3) / 3$, V_1, V_2, V_3 - объемы шаров с радиусами R_1, R_2, R_3 соответственно	V_i вычислять в подпрограмме
14	Вычислить среднее арифметическое положительных элементов для массивов A(N1), B(N2), C(N3).	$N_1 \leq 100$, $N_2 \leq 100$ $N_3 \leq 100$
15	Вычислить суммы элементов главных диагоналей матриц A(N,M) и B(M,N)	$M \leq 20$ $N \leq 30$

Лабораторная работа № 5

Тема: Текстовые компоненты и списки в Lazarus

(Label, Edit, Memo, ListBox, CheckListBox, ComboBox, MaskEdit)

Цель работы: создать программу с компонентами, согласно варианту.

Таблица вариантов

№ варианта	КОМПОНЕНТ								
	Label	Edit	Memo	ListBox	Check ListBox	Comb- oBox	Mask Edit	Radio Button	But- ton
1	*	*			*			*	
2	*		*			*	*		
3		*		*			*	*	
4		*	*		*	*			
5	*			*			*		*
6	*				*		*	*	
7		*	*			*			*
8		*		*	*			*	
9	*			*		*		*	
10	*		*			*			*

Лабораторная работа № 6

Тема: Кнопки, индикаторы и управляющие элементы в Lazarus (BitBtn, SpeedButton, RadioGroup, CheckBox, GroupBox, UpDown, TrackBar, ScrollBar, Timer)

Цель работы: создать программу и форму, сохраняющие компоненты, согласно варианту.

Таблица вариантов

№ варианта	КОМПОНЕНТ								
	Bit Btn	Speed Button	Up Down	Check- Box	Radio- Group	Group Box	Track Bar	Scrool- Bar	Timer
1	*	*			*			*	
2	*		*			*			
3		*		*			*	*	
4		*	*		*	*			
5	*			*			*		*
6	*				*		*	*	
7		*	*			*			*
8		*		*	*			*	
9	*			*		*		*	
10	*		*			*			*

Лабораторная работа № 7

Тема: Работа с меню, панелями и компонентами внешнего оформления (MainMenu, PopupMenu, Panel, Bevel, Splitter, TabControl, PageControl)

Цель работы: создать программы со стандартными меню, а также с многостраничным компонентом с закладками.

Порядок проведения работы

1. Изучить свойства заданных компонентов.
2. Создать форму и модуль <ЛРЗ-имя>.pas, <ЛРЗ-имя>.dfm.
3. В ИСР Lazarus вызвать конструктор меню и создать главное меню с произвольным набором разделов (не менее четырех) и подменю (одно–два).
4. Аналогично создать всплывающее меню PopupMenu с произвольным набором разделов.
5. Вызвать компонент Panel и создать на нем группу компонентов, например из лабораторной работы № 1.
6. Выделить часть компонентов, используя рамку Bevel.
7. Разделить панель, используя компонент Splitter.
8. Создать многостраничные компоненты, используя PageControl.
9. Создать страницу с переключаемыми закладками, используя TabControl. Панели на странице делать видимыми в зависимости от выбранной закладки в TabControl.

Лабораторная работа № 8

Тема: Обработка событий клавиатуры и мыши

Цель работы: создание программ обработчиков событий клавиатуры и мыши на базе формы, созданной в лабораторной работе № 3.

Таблица вариантов

№ варианта	Компонент							
	On Click	On Dbl Click	On Change	On Enter	OnExit	OnKey Down	On Key Press	OnKey Up
1	*			*		*		
2	*				*		*	
3		*		*		*		
4		*			*		*	
5			*	*		*		
6			*		*		*	
7	*			*				*
8	*				*			*
9		*	*			*		
10		*		*		*		

Лабораторная работа 9

Тема: Введение в язык программирования Python

Цель работы: Познакомиться со средой разработки Python. Изучить основные типы данных, команды ввода и вывода данных

Напишите программу, которая запрашивала бы у пользователя:

Вариант 1

Имя, Фамилия, Возраст, Место жительства

- фамилия, имя ("Ваши фамилия, имя?")

- возраст ("Сколько Вам лет?")

- место жительства ("Где вы живете?")

После этого выводила бы три строки:

"Ваши фамилия, имя"

"Ваш возраст"

"Вы живете в"

Вариант 2

Имя, Дата рождения, Образование

- имя ("Ваше, имя?")

- дата рождения ("Ваша дата рождения?")

- образование ("Где Вы учитесь?")

После этого выводила бы три строки:

"Ваше имя"

"Дата рождения"

"Вы учитесь в "

Вариант 3

Фамилия, Место жительства

- Фамилия("Ваша фамилия?")

- место жительства ("Где Вы живете?")

После этого выводила бы две строки:

"Ваша фамилия"

"Вы живете в"

Вариант 4

Фамилия, Место рождения, любимая музыка

- Фамилия, ("Ваша фамилия?")

- место рождения ("Где Вы родились?")

- музыка("Какая музыка нравится? ")

После этого выводила бы три строки:

"Ваши имя, фамилия"

"Вы родились в"

"Ваша любимая музыка "

Вариант 5

Имя, Фамилия, ФИО мамы, ФИО отца

- ФИО (например, "Ваши фамилия, имя, отчество?")

- возраст ("Сколько Вам лет?")

- место жительства ("Где Вы живете?")

После этого выводила бы три строки:

"Ваши имя, фамилия, отчество"

"Ваш возраст"

"Вы живете в"

Вариант 6

Имя, Любимый предмет в ВУЗе, год обучения

- имя ("Ваше имя?")

- любимый предмет ("Какой Ваш любимый предмет в ВУЗе?")

- номер класса ("На каком курсе Вы учитесь?")

После этого выводила бы три строки:

"Ваше имя"

"Ваш любимый предмет в ВУЗе"

"Вы учитесь на курсе "

Вариант 8

Имя, Фамилия, Отчество, Хобби

- ФИО (например, "Ваши фамилия, имя, отчество?")

- хобби ("Чем Вы увлекаетесь?")

После этого выводила бы две строки:

"Ваши имя, фамилия, отчество"

"Ваше хобби"

Вариант 9

Имя, Фамилия, любимый спорт

- Фамилия, имя ("Ваши фамилия, имя?")

- образование ("В каком ВУЗе Вы учитесь?")

("На каком отделении Вы учитесь?")

После этого выводила бы три строки:

"Ваши имя, фамилия"

"Вы учитесь в "

"Вы учитесь на отделении "

Вариант 10

Имя, Фамилия, Любимый предмет в институте, ФИО куратора

- Фамилия, имя ("Ваши фамилия, имя?")
- любимый предмет в ВУЗе ("Какой Ваш любимый предмет в ВУЗе?")
- ФИО классного руководителя ("ФИО Вашего классного руководителя?")

После этого выводила бы три строки:

"Ваши имя, фамилия"

"Ваш любимый предмет в ВУЗе "

"ФИО Вашего куратора"

Лабораторная работа 10

Тема: Математические операции в Python

Цель работы: познакомиться с основными математическими операциями в Python

Напишите программу, которая бы вычисляла заданное арифметическое выражение при заданных переменных. Ввод переменных осуществляется с клавиатуры. Вывести результат с 2-мя знаками после запятой.

Лабораторные задания

Вариант 1

$$a = \frac{\sqrt{|x-1|} - \sqrt[3]{|y|}}{1 + \frac{x^2}{2} + \frac{z^2}{4}}$$

Вычислить. $b = x(\arctg Z + e)^{-(y+3)}$; если $x = 2; y = 6; z = 7$.

Вариант 2.

$$a = \frac{3 + e^{y-1}}{1 + x(y - tgZ)}$$

Вычислить $b = 1 + |y - x| + \frac{(y - x)^2}{2} + \frac{|y - z|^3}{3}$; если $x = 5; y = 1; z = 4$.

Вариант 3.

$$a = (1 + y) \frac{x + y / (x^2 + 4)}{e^{-x-2} + 1 / (z^2 + 4)}$$

Вычислить $b = \frac{1 + \cos(y - 2)}{x^{4/2} + \sin^2 z}$; если $x = 1; y = 12; z = 6$.

Вариант 4.

$$a = y + \frac{x}{z^2 + \left| \frac{x^2}{y + x^3 / 3} \right|}$$

Вычислить $b = (1 + \arcsin x^2 + tg^2 \frac{z}{2})$; если $x = 11; y = 5; z = 10$.

Вариант 5.

Вычислить $a = \frac{2 \cos(x - \pi/6)}{1/2 + \sin^2 y}$, и $b = 1 + \frac{z^2}{3 + z^2/5}$; если $x = 15; y = 7; z = 3$.

Вариант 6.

Вычислить $a = \frac{1 + \sin^2(x - y)}{2 + |x - 2x/(1 + x^2 y^2)|} + z$, и $b = \cos^2(\operatorname{arctg} \frac{1}{z})$; если $x = 3; y = 4; z = 5$.

Вариант 7.

Вычислить $a = \ln \left| (y - \sqrt{|x|}) \left(x - \frac{y}{z + x^2/4} \right) \right|$, и $b = x - \frac{x^2}{3!} + \frac{x^5}{5}$; если $x = 5; y = 3; z = 17$.

Вариант 8.

Вычислить $a = \left| \frac{\sin^3 |3x^3 + 5y^2 + 15z|}{\sqrt{(12x^3 + 6y^2 - z)^2 + 3.14}} \right|$, и $b = \operatorname{tg}(7x^2 + e^{3y^2 - z})$; если $x = 7; y = 4; z = 8$.

Вариант 9.

Вычислить $a = \sqrt{\frac{|\sin 8y| + 17x}{(1 - \sin 4 \cdot y \cdot \cos(z^2 + 18))^2}}$, и $b = y - \sqrt{\frac{3|z|^2}{3 + |\operatorname{tg} 3x^2 - \sin 5y|}}$; если $x = 17; y = 16; z = 15$.

Вариант 10.

Вычислить $a = l^{(2x^2 - y)} + \frac{3 \arccos y^2}{(z^2 + xy)^2}$, и $b = \operatorname{tg} \left(\frac{|x - y|^2}{3z + \cos x^2} \right)$; если $x = 6; y = 7; z = 8$.

Лабораторная работа 11

Тема: Структура ветвление в Python

Цель работы: познакомиться со структурой ветвление (if, if-else, if-elif-else).
Научиться работать с числами и строками используя данную структуру.

Задания для самостоятельной работы (по вариантам)

Вариант 1

Даны три целых числа. Выбрать из них те, которые принадлежат интервалу [1,3].

Вариант 2

Дан номер года (положительное целое число). Определить количество дней в этом году,

учитывая, что обычный год насчитывает 365 дней, а високосный — 366 дней. Високосным считается год, делящийся на 4, за исключением тех годов, которые делятся на 100 и не делятся на 400 (например, годы 300, 1300 и 1900 не являются високосными, а 1200 и 2000 — являются).

Вариант 3

Написать программу вычисления стоимости покупки с учетом скидки. Скидка в 3% предоставляется в том случае, если сумма покупки больше 500 руб., в 5% - если сумма больше 1000 руб.

Вариант 4

Написать программу, которая бы по введенному номеру единицы измерения (1 — килограмм, 2 — миллиграмм, 3 — грамм, 4 — тонна, 5 — центнер) и массе M выдавала соответствующее значение массы в килограммах.

Вариант 5

Найти косинус минимального из 4 заданных чисел.

Вариант 6

Вывести на экран синус максимального из 3 заданных чисел.

Вариант 7

Даны три стороны одного треугольника и три стороны другого треугольника. Определить, будут ли эти треугольники равновеликими, т. е. имеют ли они равные площади. Если это не так, то вывести «Foul!!!»

Вариант 8

Составьте программу подсчёта площади равнобедренного треугольника. Если площадь треугольника чётная, разделить её на 2, в противном случае вывести сообщение «Не могу делить на 2!»

Вариант 9

Составить программу, которая по данному числу (1-12) выводит название соответствующего ему месяца на английском языке.

Вариант 10

Составить программу, осуществляющую перевод величин из радианной меры в градусную или наоборот. Программа должна запрашивать, какой перевод нужно осуществить, и выполнять указанное действие.

Лабораторная работа 12

Тема: Работа с циклами в Python

Цель работы: познакомиться с циклическими конструкциями

Вариант 1

1. Дано вещественное число – цена 1 кг конфет. Вывести стоимость 1, 2, ... 10 кг конфет. Решить задачу используя циклическую конструкцию `for`.
2. Дана непустая последовательность целых чисел, оканчивающаяся нулем. Найти: а) сумму всех чисел последовательности; б) количество всех чисел последовательности. Решить задачу используя циклическую конструкцию `while`.

Вариант 2

1. Даны два числа A и B ($A < B$). Найти сумму всех целых чисел от A до B включительно. Решить задачу используя циклическую конструкцию `for`.
2. Дана последовательность отрицательных целых чисел, оканчивающаяся положительным числом. Найти среднее арифметическое всех чисел последовательности (без учета положительным числа). Решить задачу используя циклическую конструкцию `while`.

Вариант 3

1. Даны два числа A и B ($A < B$). Найти сумму квадратов всех целых чисел от A до B включительно. Решить задачу используя циклическую конструкцию `for`.
2. Дана последовательность из n целых чисел. Первое число в последовательности чётное. Найти сумму всех идущих подряд в начале последовательности чётных чисел. Условный оператор не использовать. Решить задачу используя циклическую конструкцию `while`.

Вариант 4

1. Найти среднее арифметическое всех целых чисел от a до 200 (значения a и b вводятся с клавиатуры; $a \leq 200$). Решить задачу используя циклическую конструкцию `for`.
2. Дана последовательность из n вещественных чисел, начинающаяся с положительного числа. Определить, какое количество положительных чисел записано в начале последовательности. Условный оператор не использовать. Решить задачу используя циклическую конструкцию `while`.

Вариант 5

1. Найти сумму всех целых чисел от a до b (значения a и b вводятся с клавиатуры; $b \geq a$). Решить задачу используя циклическую конструкцию `for`.
2. Дано целое число N (> 0), являющееся некоторой степенью числа 2: $N = 2^K$. Найти целое число K — показатель этой степени. Решить задачу используя циклическую конструкцию `while`.

Вариант 6

1. Найти сумму квадратов всех целых чисел от a до 50 (значение a вводится с клавиатуры; $0 \leq a \leq 50$). Решить задачу используя циклическую конструкцию `for`.

2. Дано целое число $N (> 1)$. Найти наименьшее целое число K , при котором выполняется неравенство $5^K > N$.

Решить задачу используя циклическую конструкцию `while`.

Вариант 7

1. Дана непустая последовательность целых чисел, оканчивающаяся нулем.

Найти:

а) сумму всех чисел последовательности;

б) количество всех чисел последовательности.

Решить задачу используя циклическую конструкцию `for`.

2. Дано целое число $N (> 1)$. Найти наибольшее целое число K , при котором выполняется неравенство $2^K > N$.

Решить задачу используя циклическую конструкцию `while`.

Вариант 8

1. Дана последовательность из n вещественных чисел. Первое число в последовательности нечетное. Найти сумму всех идущих подряд в начале последовательности нечетных чисел. Условный оператор не использовать. Решить задачу используя циклическую конструкцию `for`.

2. Дано целое число $N (> 0)$. Используя операции деления нацело и взятия остатка от деления, найти количество и сумму его цифр.

Решить задачу используя циклическую конструкцию `while`.

Вариант 9

1. Среди чисел 1, 4, 9, 16, 25, ... найти первое число, большее n . Решить задачу используя циклическую конструкцию `for`.

2. Среди чисел 1, 5, 10, 16, 23, ... найти первое число, большее n . Условный оператор не использовать.

Решить задачу используя циклическую конструкцию `while`.

Вариант 10

1. Известны оценки по физике каждого из 20 учеников класса. Определить среднюю оценку. Решить задачу используя циклическую конструкцию `for`.

2. Дано число $A (> 1)$. Вывести наибольшее из целых чисел K , для которых сумма $1 + 1/2 + \dots + 1/K$ будет меньше A , и саму эту сумму.

Решить задачу используя циклическую конструкцию `while`.

Лабораторная работа 13

Тема: Работа со строками в Python

Цель работы: познакомиться с методами работы со строками.

Задания для самостоятельной работы (по вариантам)

Вариант 1

Дана строка, содержащая русскоязычный текст. Найти количество слов, начинающихся с буквы "е".

Вариант 2

В строке заменить все двоеточия (:) знаком процента (%). Подсчитать количество замен.

Вариант 3

В строке удалить символ точку (.) и подсчитать количество удаленных символов.

Вариант 4

В строке заменить букву(a) буквой (o). Подсчитать количество замен. Подсчитать, сколько символов в строке.

Вариант 5

В строке заменить все заглавные буквы строчными.

Вариант 6

В строке удалить все буквы "a" и подсчитать количество удаленных символов.

Вариант 7

Дана строка. Преобразовать ее, заменив звездочками все буквы "п", встречающиеся среди первых $n/2$ символов. Здесь n - длина строки.

Вариант 8

Дана строка, заканчивающаяся точкой. Подсчитать, сколько слов в строке.

Вариант 9

Определить, сколько раз в тексте встречается заданное слово.

Вариант 10

Дана строка-предложение на английском языке. Преобразовать строку так, чтобы каждое слово начиналось с заглавной буквы.

Лабораторная работа 14

Тема: Работа со списками. Операции над списками в Python

Цель работы: Изучение списков в Python.

Вариант 1

1. Дан одномерный массив, состоящий из N целочисленных элементов. Ввести массив с клавиатуры. Найти максимальный элемент. Вывести массив на экран в обратном порядке.
2. В массиве действительных чисел все нулевые элементы заменить на среднее арифметическое всех элементов массива.

Вариант 2

1. Дан одномерный массив, состоящий из N целочисленных элементов. Ввести массив с клавиатуры. Найти минимальный элемент. Вывести индекс минимального элемента на экран.
2. Дан массив целых чисел. Переписать все положительные элементы во второй массив, а остальные - в третий.

Вариант 3

1. В одномерном числовом массиве D длиной n вычислить сумму элементов с нечетными индексами. Вывести на экран массив D , полученную сумму.

2. Дан одномерный массив из 8 элементов. Заменить все элементы массива меньшие 15 их удвоенными значениями. Вывести на экран монитора преобразованный массив.

Вариант 4

1. Дан массив целых чисел. Найти максимальный элемент массива и его порядковый номер.
2. Дан одномерный массив целого типа. Получить другой массив, состоящий только из нечетных чисел исходного массива или сообщить, что таких чисел нет. Полученный массив вывести в порядке убывания элементов.

Вариант 5

1. Дан одномерный массив из 10 целых чисел. Вывести пары отрицательных чисел, стоящих рядом.
2. Дан целочисленный массив размера 10. Создать новый массив, удалив все одинаковые элементы, оставив их 1 раз.

Вариант 6

1. Дан одномерный массив из 10 целых чисел. Найти максимальный элемент и сравнить с ним остальные элементы. Вывести количество меньших максимального и больших максимального элемента.
2. Одномерный массив из 10-и целых чисел заполнить с клавиатуры, определить сумму тех чисел, которые >5 .

Вариант 7

1. Дан массив целых чисел. Найти сумму элементов с четными номерами и произведение элементов с нечетными номерами. Вывести сумму и произведение.
2. Переставить в одномерном массиве минимальный элемент и максимальный.

Вариант 8

1. Найдите сумму и произведение элементов списка. Результаты вывести на экран.
2. В массиве действительных чисел все нулевые элементы заменить на среднее арифметическое всех элементов массива.

Вариант 9

1. Дан одномерный массив, состоящий из N вещественных элементов. Ввести массив с клавиатуры. Найти и вывести минимальный по модулю элемент. Вывести массив на экран в обратном порядке.
2. Даны массивы A и B одинакового размера 10. Вывести исходные массивы. Поменять местами их содержимое и вывести в начале элементы преобразованного массива A , а затем — элементы преобразованного массива B .

Вариант 10

1. Определите, есть ли в списке повторяющиеся элементы, если да, то вывести на экран это значение, иначе сообщение об их отсутствии.
2. Дан одномерный массив из 15 элементов. Элементам массива меньше 10 присвоить нулевые значения, а элементам больше 20 присвоить 1. Вывести на экран монитора первоначальный и преобразованный массивы в строку.

Лабораторная работа 15
Тема: Функции в Python

Цель работы: изучение функций в Python.

Вариант 1.

1. Составить программу для вычисления площади разных геометрических фигур.
2. Даны 3 различных массива целых чисел (размер каждого не превышает 15). В каждом массиве найти сумму элементов и среднеарифметическое значение.

Вариант 2.

1. Вычислить площадь правильного шестиугольника со стороной, a , используя подпрограмму вычисления площади треугольника.
2. Пользователь вводит две стороны трех прямоугольников. Вывести их площади.

Вариант 3.

1. Даны катеты двух прямоугольных треугольников. Написать функцию вычисления длины гипотенузы этих треугольников. Сравнить и вывести какая из гипотенуз больше, а какая меньше.
2. Преобразовать строку так, чтобы буквы каждого слова в ней были отсортированы по алфавиту.

Вариант 4.

1. Даны две дроби A/B и C/D (A, B, C, D — натуральные числа). Составить программу деления дроби на дробь. Ответ должен быть несократимой дробью. Использовать подпрограмму алгоритма Евклида для определения НОД.
2. Задана окружность $(x-a)^2 + (y-b)^2 = R^2$ и точки $P(p_1, p_2)$, $F(f_1, f_1)$, $L(l_1, l_2)$. Выяснить и вывести на экран, сколько точек лежит внутри окружности. Проверку, лежит ли точка внутри окружности, оформить в виде процедуры.

Вариант 5.

1. Даны две дроби A/B и C/D (A, B, C, D — натуральные числа). Составить программу вычитания из первой дроби второй. Ответ должен быть несократимой дробью. Использовать подпрограмму алгоритма Евклида для определения НОД.
2. Напишите программу, которая выводит в одну строчку все делители переданного ей числа, разделяя их пробелами.

Вариант 6.

1. Составить программу нахождения наибольшего общего делителя (НОД) и наименьшего общего кратного (НОК) двух натуральных чисел $НОК(A, B) = (A * B) / НОД(A, B)$. Использовать подпрограмму алгоритма Евклида для определения НОД.
2. Составить программу вычисления площади выпуклого четырехугольника, заданного длинами четырех сторон и диагонали.

Вариант 7.

1. Даны числа X, Y, Z, T — длины сторон четырехугольника. Вычислить его площадь, если угол между сторонами длиной X и Y — прямой. Использовать две подпрограммы для вычисления площадей: прямоугольного треугольника и прямоугольника.
2. Напишите программу, которая переводит переданное ей неотрицательное целое число в 10-значный восьмеричный код, сохранив лидирующие нули.

Вариант 8.

1. Найти все натуральные числа, не превосходящие заданного n , которые делятся на каждую из своих цифр.

2. Ввести одномерный массив A длиной m . Поменять в нём местами первый и последний элементы. Длину массива и его элементы ввести с клавиатуры. В программе описать процедуру для замены элементов массива. Вывести исходные и полученные массивы.

Вариант 9.

1. Из заданного числа вычли сумму его цифр. Из результата вновь вычли сумму его цифр и т. д. Через сколько таких действий получится нуль?
2. Даны 3 различных массива целых чисел. В каждом массиве найти произведение элементов и среднеарифметическое значение.

Вариант 10.

1. На отрезке $[100, N]$ ($210 < N < 231$) найти количество чисел, составленных из цифр a, b, c .
2. Составить программу, которая изменяет последовательность слов в строке на обратную.

Лабораторная работа 16

Тема: Работа с двумерными массивами (вложенные списки).

Цель работы: изучение вложенных списков в Python.

Вариант 1.

1. Вычислить сумму и число положительных элементов матрицы $A[N, N]$, находящихся над главной диагональю.
2. Дана матрица $B[N, M]$. Найти в каждой строке матрицы максимальный и минимальный элементы и поменять их с первым и последним элементами строки соответственно.

Вариант 2.

1. Дана целая квадратная матрица n -го порядка. Определить, является ли она магическим квадратом, т. е. такой матрицей, в которой суммы элементов во всех строках и столбцах одинаковы.
2. Дана прямоугольная матрица $A[N, N]$. Переставить первый и последний столбцы местами и вывести на экран.

Вариант 3.

1. Определить, является ли заданная целая квадратная матрица n -го порядка симметричной (относительно главной диагонали).
2. Дана вещественная матрица размером $n \times m$. Переставляя ее строки и столбцы, добиться того, чтобы наибольший элемент (или один из них) оказался в верхнем левом углу.

Вариант 4.

1. Дана прямоугольная матрица. Найти строку с наибольшей и строку с наименьшей суммой элементов. Вывести на печать найденные строки и суммы их элементов.
2. Дана квадратная матрица $A[N, N]$, Записать на место отрицательных элементов матрицы нули, а на место положительных — единицы. Вывести на печать нижнюю треугольную матрицу в общепринятом виде.

Вариант 5.

1. Упорядочить по возрастанию элементы каждой строки матрицы размером $n \times m$.

2. Дана действительная матрица размером $n \times m$, все элементы которой различны. В каждой строке выбирается элемент с наименьшим значением. Если число четное, то заменяется нулем, нечетное - единицей. Вывести на экран новую матрицу.

Вариант 6.

1. Дана целочисленная квадратная матрица. Найти в каждой строке наибольший элемент и в каждом столбце наименьший. Вывести на экран.

2. Дана действительная квадратная матрица порядка N (N — нечетное), все элементы которой различны. Найти наибольший элемент среди стоящих на главной и побочной диагоналях и поменять его местами с элементом, стоящим на пересечении этих диагоналей.

Вариант 7.

1. Квадратная матрица, симметричная относительно главной диагонали, задана верхним треугольником в виде одномерного массива. Восстановить исходную матрицу и напечатать по строкам.

2. Для заданной квадратной матрицы сформировать одномерный массив из ее диагональных элементов. Найти след матрицы, просуммировав элементы одномерного массива. Преобразовать исходную матрицу по правилу: четные строки разделить на полученное значение, нечетные оставить без изменения.

Вариант 8.

1. Задана матрица порядка n и число k . Разделить элементы k -й строки на диагональный элемент, расположенный в этой строке.

2. Задана квадратная матрица. Получить транспонированную матрицу (перевернутую относительно главной диагонали) и вывести на экран.

Вариант 9.

1. Для целочисленной квадратной матрицы найти число элементов, кратных k , и наибольший из этих элементов.

2. В данной действительной квадратной матрице порядка n найти наибольший по модулю элемент. Получить квадратную матрицу порядка $n - 1$ путем отбрасывания из исходной матрицы строки и столбца, на пересечении которых расположен элемент с найденным значением.

Вариант 10.

1. Найти максимальный среди всех элементов тех строк заданной матрицы, которые упорядочены (либо по возрастанию, либо по убыванию).

2. Расположить столбцы матрицы $D[M, N]$ в порядке возрастания элементов k -й строки ($1 \leq k \leq M$).

Темы контрольных работ

по дисциплине Языки и методы программирования

Создать приложение в среде разработки Lazarus.

Задание 1

- 01 найти площадь прямоугольного треугольника с катетами a и b и указать, превышает ли она S_1
- 02 найти радиус окружности с площадью S и сравнить его с R_1
- 03 сравнить объемы куба с ребром a и цилиндра с диаметром a и высотой h
- 04 определить какая из трех точек плоскости, заданных координатами x, y находится дальше от начала координат
- 05 определить, находится ли начало координат внутри треугольника, заданного координатами вершин $(-x_1, y_1), (x_1, y_1), (0, y_2)$, $x_1 < 0, y_2 < y_1$
- 06 определить, имеет ли действительные корни уравнение $x^2 + x \sin(R) + Q = 0$.
- 07 сравнить среднее арифметическое и среднее геометрическое чисел A и B
- 08 определить, у какой из трех пар чисел X_i, Y_i наименьшее значение выражения $X_i - Y_i$
- 09 найти гипотенузу прямоугольного треугольника с катетом A и площадью S и сравнить ее с S_1
- 10 определить меньший из углов прямоугольного треугольника с катетами A и B
- 11 определить, могут ли отрезки с длинами a, b, c быть вершинами треугольника
- 12 найти длину первого слова, начинающегося на Z , во введенной строке.
Слова разделяются пробелами
- 13 найти количество символов, отличных от пробела, во введенной строке
- 14 найти максимальное по абсолютной величине из введенных чисел
- 15 найти минимальное по абсолютной величине из введенных чисел
- 16 найти минимальное среди введенных чисел
- 17 массив размером 3×5 заполнить случайными числами от -10 до 10 и найти количество положительных элементов
- 18 массив размером 5×4 заполнить случайными числами от -10 до 10 и найти количество неположительных элементов
- 19 массив размером 3×5 заполнить случайными числами от -10 до 10 и найти наибольший элемент и его индексы
- 20 массив размером 4×3 заполнить случайными числами от 0 до 30 и найти количество неотрицательных элементов
- 21 массив размером 3×5 заполнить случайными числами от -20 до 0 и найти наибольший по абсолютной величине элемент и его индексы
- 22 сравнить количество введенных строк с N_1
- 23 посчитать количество четных цифр во введенном числе
- 24 посчитать количество семерок во введенном числе
- 25 заполнить одномерный массив размерностью 10 случайными числами от 0 до 90 и элементы переставить в обратном порядке

- 26 найти след (сумму элементов главной диагонали) матрицы A размерностью 4x4, заполненной случайными числами от -5 до 35
- 27 сравнить суммы введенных чисел с четными и нечетными номерами
- 28 вывести две введенные строки в алфавитном порядке
- 29 вывести только наиболее короткую из введенных строк
- 30 вывести последнюю в алфавитном порядке из введенных строк
- 31 выяснить, больше ли 3 чем число букв 'd' во введенной строке
- 32 выяснить, сколько раз встречается сочетание 'the' во введенной строке
- 33 определить, сколько из введенных строк заканчиваются '.'
- 34 определить, сколько из введенных строк начинаются на 'Z'
- 35 определить процентное содержание пробелов во введенной строке
- 36 определить процентное содержание буквы Q во введенной строке (без учета пробелов)
- 37 заполнить одномерный массив [1..10] случайными числами от 0 до 90 и определить сумму элементов с четными индексами
- 38 заполнить одномерный массив [1..10] случайными числами от -20 до 20 и определить количество положительных элементов с четными индексами
- 39 найти количество четных чисел среди введенных
- 40 найти количество нечетных чисел среди введенных
- 41 сравнить количество четных и нечетных среди введенных чисел
- 42 массив размером 3 x 5 заполнить случайными числами от -10 до 10 и сравнить количество положительных и отрицательных элементов
- 43 массив размером 3 x 5 заполнить случайными числами от -10 до 10 и сравнить суммы положительных и отрицательных элементов
- 44 массив размером 3 x 5 заполнить случайными числами от -10 до 10 и сравнить количество положительных элементов с N1
- 45 массив размером 4 x 5 заполнить случайными числами от -10 до 10 и найти сумму элементов, у которых второй индекс больше первого

Задание 2

- 01 Определить количество строк файла 'f.txt'.
- 02 Как открыть файл 'qwert.1' для чтения?
- 03 Как открыть файл 'a1.txt' для записи?
- 04 Как открыть файл 'as22.t' для расширения (добавления к текущему содержимому)?
- 05 Написать функцию, определяющую количество строк текстового файла 'abc.txt', в которых не содержится 'abcd'
- 06 Написать функцию, определяющую номер строки с минимальным числом текстового файла 'd.txt', содержащего в каждой строке целое число.
- 07 Написать функцию, определяющую количество строк текстового файла 'abc.txt', в которых содержится 'abcd'
- 08 Как прочитать информацию последовательно из нескольких файлов, используя одну файловую переменную?
- 09 Дописать к файлу его пятую строку,

используя одну файловую переменную.

- 10 Написать подпрограмму, которая выводит на экран имя файла и количество строк в нем по заданному имени файла.
- 11 Написать функцию, которая возвращает третью строку файла с именем, заданным в качестве параметра.
- 12 Написать функцию, которая возвращает последнюю строку файла с именем, заданным в качестве параметра.
- 13 Написать функцию, которая возвращает длину третьей строки файла с именем, заданным в качестве параметра.
- 14 Написать функцию, которая возвращает длину последней строки файла с именем, заданным в качестве параметра.
- 15 Написать функцию, которая возвращает количество строк файла с именем, заданным в качестве параметра.
- 16 Написать функцию, которая возвращает количество длинных (с длиной > 10 символов) строк файла 'abc.txt'.
- 17 Написать функцию, которая возвращает количество коротких (с длиной < 10 символов) строк файла 'abc.txt'.
- 18 Написать функцию, которая возвращает количество содержащих пробел строк файла 'abc.txt'.
- 19 Написать функцию, которая возвращает количество не содержащих пробел строк файла 'abc.txt'.
- 20 Написать функцию, которая возвращает True, если хотя бы одна строка файла 'abc.txt' содержит 'abc'.
- 21 Написать функцию, определяющую номер строки с максимальным числом текстового файла 'd.txt', содержащего в каждой строке целое число.
- 22 Дописать первые 3 строки файла 'c:\1\al.txt' к файлу 'c:\1\d.txt'
- 23 Дописать последние 3 строки файла 'c:\1\al.txt' к файлу 'c:\1\d.txt'
- 24 Записать первые 3 строки файла 'c:\1\al.txt' в файл 'c:\1\d.txt'
- 25 Записать последние 3 строки файла 'c:\1\al.txt' в файл 'c:\1\d.txt'
- 26 Дописать с 4 по 6 строки файла 'c:\1\al.txt' к файлу 'c:\1\d.txt'
- 27 Дописать с 4 по 6 строки файла 'c:\1\al.txt' к файлу 'c:\1\d.txt'
- 28 Записать с 4 по 6 строки файла 'c:\1\al.txt' в файл 'c:\1\d.txt'
- 29 Записать с 4 по 6 строки файла 'c:\1\al.txt' в файл 'c:\1\d.txt'
- 30 Определить, какая из четных строк файла 'qq.txt' имеет минимальную длину.
- 31 Определить, какая из четных строк файла 'qq.txt' имеет максимальную длину.
- 32 Определить, используются ли массивы в программе на языке Pascal, находящейся в файле 'test1.pas'.

Для каждого обучающегося выбирается одна из комбинаций вариантов, например

1A/Z2/5B/6A/7B/8A — Это Приложение с графическим интерфейсом для ввода из входного типизированного файла заданного количества целых чисел, сортировки их методом пузырька и записи в выходной, отличный от входного, текстовый файл.

1A Приложение с графическим интерфейсом

1B Консольное приложение

Z1) для записи в 2A текстовый файл

2B типизированный файл

3A упорядоченных по возрастанию

3B упорядоченных по убыванию

3B случайных

4A целых чисел

4B действительных чисел

Z2) для ввода из входного 5A текстового файла

5B типизированного файла

заданного количества

6A целых чисел

6B действительных чисел

сортировки их

7A методом вставки

7B методом выбора

7B методом пузырька

и вывода в выходной, отличный от входного

8A текстовый файл

8B типизированный файл

Z3) для ввода из входного 9A текстового файла

9B типизированного файла

заданного количества

10A целых чисел

10B действительных чисел

и вывода их на экран для просмотра

(только приложение с графическим интерфейсом)

* * *

на следующем этапе эти программы будут использоваться для сравнения алгоритмов сортировки по эффективности.

NB для сортировок продумать, как определять количество сравнений и пересылок, какие части следует оформить в виде подпрограмм.

5. Методические материалы, определяющие процедуры оценивания компетенции

5.1 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

5.2 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.

5.3 Критерии оценивания зачета

Оценка **«зачтено»** выставляется обучающемуся, если обучающийся почти ответил на все вопросы, поставленные преподавателем на защите.

Оценка **«не зачтено»** выставляется обучающемуся, если обучающийся не проявил глубоких теоретических знаний при ответе на вопросы

5.4 Критерии оценивания тестирования

При тестировании все верные ответы берутся за 100%.

90%-100% отлично

75%-90% хорошо

60%-75% удовлетворительно

менее 60% неудовлетворительно

5.5 Критерии оценивания качества выполнения контрольной работы

Оценка **«зачтено»** выставляется, если представленные программы решают поставленную задачу и обучающийся может обосновать достоверность результата.

Оценка **«незачтено»** – если представленные программы не решают поставленную задачу или обучающийся не может обосновать достоверность результата.

5.6 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.