


МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ
Заведующий кафедрой
Физики твердого тела и наноструктур
 (П.В.Середин)
31.08.2022

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
Б1.В.10 Программирование на языке высокого уровня

1. Код и наименование направления подготовки/специальности:

11.03.04 Электроника и микроэлектроника

2. Профиль подготовки/специализация:

Интегральная электроника и микроэлектроника

3. Квалификация выпускника: бакалавр

4. Форма обучения: очная

5. Кафедра, отвечающая за реализацию дисциплины: Кафедра физики твердого тела и наноструктур

6. Составители программы: Дубровский О.И., кандидат физ.-мат. наук, доцент

7. Рекомендована: НМС Физического факультета ВГУ

протокол №6 от 14.06.2022 г.

8. Учебный год: 2022/2023

Семестр: 2

9. Цели и задачи учебной дисциплины

Целью преподавания дисциплины является получение студентами знаний, необходимых при конструировании пользовательских программ для класса вычислительных задач в области электроники и нанoeлектроники.

Задачами изучения дисциплины являются:

- изучение базовых понятий алгоритмизации и программирования;
- освоение основ программирования на языке высокого уровня;
- ознакомление с принципами объектно-ориентированного подхода при разработке приложений.

10. Место учебной дисциплины в структуре ООП: часть, формируемая участниками образовательных отношений, блок Б1. Изучение дисциплины базируется на знаниях, полученных в школе при изучении информатики, а также при изучении курса "Информационные технологии" в 1-м семестре. Дисциплина "Языки программирования высокого уровня" предшествует и является основополагающей для изучения следующих дисциплин: Численные методы, Компьютерное моделирование материалов микро- и нанoeлектроники, Основы проектирования электронной компонентной базы.

11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями) и индикаторами их достижения:

Код	Название компетенции	Код(ы)	Индикатор(ы)	Планируемые результаты обучения
УК-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач:	УК-1.2	Определяет, интерпретирует и ранжирует информацию, требуемую для решения поставленной задачи	Знать: способы обработки информации; Уметь: применять методы обработки информации и ее интерпретации для решения поставленной задачи, Владеть: навыками поиска информации для решения поставленной задачи с применением различных методов
ПКВ-1	Способен выполнять расчет и проектирование электронных приборов, схем и устройств различного функционального назначения в соответствии с техническим заданием с использованием средств автоматизации проектирования	ПКВ-1.2	Определяет численные значения основных технических характеристик цифровых и аналоговых СФ-блоков	Знать: методы определения численных значений основных технических характеристик цифровых и аналоговых СФ-блоков; Уметь: определять численные значения основных технических характеристик цифровых и аналоговых СФ-блоков, программировать на языке высокого уровня; Владеть: навыками определения численных значений основных технических характеристик цифровых и аналоговых СФ-блоков
ПКВ-2	Способен выполнять моделирование элементов интегральных схем и принимать решения об уточнении первичного схемотехнического описания на основе результатов анализа и верификации результатов моделирования	ПКВ-2.1	Применяет средства САПР для реализации основных методов схемотехнического моделирования	Знать: основные методы схемотехнического моделирования; Уметь: применять средства САПР для реализации основных методов схемотехнического моделирования; Владеть: навыками применения средств САПР для реализации основных методов схемотехнического моделирования.

12. Объем дисциплины в зачетных единицах/час. — 6/216.

Форма промежуточной аттестации: экзамен

13. Трудоемкость по видам учебной работы

Вид учебной работы	Трудоемкость			
	Всего	По семестрам		
		2 семестр	№ семестра	...
Аудиторные занятия	84	84		
в том числе:	лекции	34	34	
	практические			
	лабораторные	50	50	
Самостоятельная работа	96	96		
в том числе: курсовая работа (проект)				
Форма промежуточной аттестации (экзамен – 36 час.)	36	36		
Итого:	216	216		

13.1. Содержание дисциплины

№ п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК *
1. Лекции			
1.1	Основные понятия пользовательского программирования	<u>Алгоритмы.</u> Понятие алгоритма. Свойства алгоритмов. Способы задания алгоритмов. Блок-схемы. <u>Программное обеспечение.</u> Уровни программного обеспечения. Понятие об операционной системе и ее основные функции. Классификация прикладных и служебных программных средств. Обеспечение интерфейса пользователя. Обеспечение автоматического запуска. Организация файловой системы. <u>Краткая характеристика языков программирования.</u> Лексика, синтаксис и семантика языка программирования. Уровни языков программирования. Краткий обзор языков программирования высокого уровня. Стандарты и версии языков. Эволюция языков. Программный продукт и его характеристики. <u>Системы программирования.</u> Состав системы программирования. Компиляторы и интерпретаторы. Редактор связей. Отладчики. Исходный и объектный модули, исполняемая программа. Интегрированная среда программирования.	-
1.2	Основы алгоритмического языка Паскаль	Типы данных и константы языка Паскаль. Выражения, операнды, операции. Структура программы на Паскале.	-
1.3	Реализация базовых алгоритмических конструкций на языке Паскаль.	<u>Линейные программы.</u> Оператор присваивания. Арифметические выражения. Типы в арифметических выражениях. Ввод и вывод данных. <u>Программирование разветвлений.</u> Логические выражения. Составной оператор. Условный оператор If . Оператор выбора Case . Оператор безусловного перехода GoTo . <u>Программирование циклов.</u> Оператор цикла с предусловием While . Оператор цикла с постусловием Repeat . Оператор цикла с параметром For . Процедуры Break и Continue . Сравнение работы операторов While, Repeat и For .	-

1.4	Структуры данных.	Массивы. Действия над массивами. Строки. Операции над строками. Строковые процедуры и функции. Записи. Оператор присоединения With . Множества. Операции над множествами.	-
1.5	Структурное и модульное программирование на языке Паскаль	<u>Процедуры и функции.</u> Структура процедуры. Структура функции. Локальные и глобальные метки, константы, типы и переменные. Формальные и фактические параметры. Параметры-значения. Параметры-переменные. Стек. Параметры-константы. Рекурсивные процедуры и функции. <u>Модули.</u> Заголовок модуля. Интерфейс модуля. Исполнительная часть модуля. Секция инициализации. Компиляция модулей. Использование модуля в основной программе. Использование идентификаторов модуля. Стандартные модули. Основные возможности стандартного модуля Crt. Программирование графики. Инициализация графического режима.	-
1.6	Организация ввода-вывода в Паскале. Работа с файлами	Общие принципы работы с файлами. Процедура Assign . Процедуры Reset и Rewrite . Считывание из файла и запись в файл. Процедура Close . Общие средства для работы с файлами. Текстовые файлы. Типизированные файлы. Нетипизированные файлы.	-
1.7	Динамические структуры данных	Статические и динамические переменные. Указатели и адреса. Присвоение значений указателю. Константы ссылочного типа. Некоторые функции работы с памятью. Динамически формируемые массивы и строки. Особенность отладки программ с динамическими переменными. Явный доступ в память. Процедуры и функции для работы с динамическими переменными. Структуры данных на основе указателей. Связанные списки. Двоичные деревья.	-
1.8	Программирование некоторых классических алгоритмов и задач на Паскале.	<u>Алгоритмы поиска.</u> Линейный поиск. Поиск с барьером. Двоичный (бинарный) поиск. Поиск по ключу в массиве. <u>Алгоритмы сортировки.</u> Сортировка выбором. Пузырьковая сортировка. Шейкерная сортировка. Сортировка включением. Знакомство с сортировкой файлов.	-
1.9	Основы объектно-ориентированного программирования	Классы и объекты. Алгоритмы поиска. Инкапсуляция. Наследование. Полиморфизм.	-
1.10	Среда визуального программирования Lazarus.	Организация проекта. Файлы проекта. Основные проектные операции: создание и сохранение проекта, оформление проекта. Работа с окнами проекта. Управление проектами. Модульный принцип построения программ. Структура головного файла проекта. Структура кода модулей. Создание форм. Размещение компонентов на форме. Родители и владельцы компонентов. Основные стандартные компоненты, их использование. Объекты и объектные классы: форма, кнопка, метка, контейнер, список. Разработка пользовательского интерфейса, согласованность интерфейса. Проектирование диалогов. Стандартные элементы диалога: панели, поля выбора, меню, кнопки, списки, полосы прокрутки, графические элементы. Свойства и методы компонентов. События. Обработка событий клавиатуры и мыши.	-

2. Лабораторные занятия			
2.1	Основные понятия пользовательского программирования		-
2.2	Основы алгоритмического языка Паскаль		-
2.3	Реализация базовых алгоритмических конструкций на языке Паскаль.	Лабораторная работа 1. Линейная программа. Вычисление логических выражений Лабораторная работа 2. Программирование ветвлений Лабораторная работа 3. Использование цикла с параметром Лабораторная работа 4. Использование цикла с условием Лабораторная работа 5. Вложенные циклы	-
2.4	Структуры данных.	Лабораторная работа 6. Обработка одномерных массивов Лабораторная работа 7. Работа с двумерными массивами Лабораторная работа 8. Обработка строк текста Лабораторная работа 9. Работа с записями Лабораторная работа 10. Работа с множествами	-
2.5	Структурное и модульное программирование на языке Паскаль	Лабораторная работа 11. Использование подпрограмм. Лабораторная работа 12. Использование стандартных модулей Лабораторная работа 13. Создание пользовательского модуля	-
2.6	Организация ввода-вывода в Паскале. Работа с файлами	Лабораторная работа 14. Создание и работа с текстовым файлом Лабораторная работа 15. Создание и работа с типизированным файлом	-
2.7	Динамические структуры данных	Лабораторная работа 16. Использование динамических массивов Лабораторная работа 17. Использование связанных списков	-
2.8	Программирование некоторых классических алгоритмов и задач на Паскале.	Лабораторная работа 18. Алгоритмы поиска Лабораторная работа 19. Алгоритмы сортировки	-
2.9	Основы объектно-ориентированного программирования	Лабораторная работа 20. Создание классов Лабораторная работа 21. Создание программы с использованием наследования	-
2.10	Среда визуального программирования Lazarus.	Лабораторная работа 22. Визуальное программирование в среде Lazarus	-

13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование темы (раздела) дисциплины	Виды занятий (количество часов)				
		Лекции	Практические	Лабораторные	Самостоятельная работа	Всего
1	Основные понятия пользовательского программирования	2		-	4	6
2	Основы алгоритмического языка Паскаль	2		-	8	12
3	Реализация базовых алгоритмических конструкций на языке Пас-	2		8	12	22

	каль.					
4	Структуры данных.	4		8	12	24
5	Структурное и модульное программирование на языке Паскаль	4		8	14	26
6	Организация ввода-вывода в Паскале. Работа с файлами	2		2	8	12
7	Динамические структуры данных	4		6	10	20
8	Программирование некоторых классических алгоритмов и задач на Паскале.	4		6	10	20
9	Основы объектно-ориентированного программирования	6		4	10	18
10	Среда визуального программирования Lazarus.	4		8	8	20
	Итого:	34		50	96	180

14. Методические указания для обучающихся по освоению дисциплины

Изучение дисциплины «Языки программирования высокого уровня» предусматривает осуществление учебной деятельности состоящей из двух частей: обучения студентов преподавателем и самостоятельной учебной деятельности студентов по изучению дисциплины.

В учебном процессе используются следующие образовательные технологии. По образовательным формам: лекции; лабораторные занятия; индивидуальные занятия. По преобладающим методам и приемам обучения: объяснительно-иллюстративные (объяснение, показ–демонстрация учебного материала и др.); активные (анализ учебной и научной литературы, составление схем и др.) и интерактивные, в том числе и групповые (взаимное обучение в форме подготовки и обсуждения докладов); информационные; компьютерные; мультимедийные (работа с сайтами академических структур, научно-исследовательских организаций, электронных библиотек и др., разработка презентаций, сообщений и докладов, работа с электронными обучающими программами и т.п.).

Основной, наиболее экономичной формой получения и усвоения информации, теоретических знаний в вузе является лекция, позволяющая воспринять значительную сумму основных знаний и потому способствующая повышению продуктивности всех других форм учебного труда. Подготовка к лекциям является одним из видов самостоятельной работы студентов. Студентам, чтобы хорошо овладеть учебным материалом, необходимо выработать навыки правильной и планомерной работы. Перед началом лекционных занятий надо просмотреть все, что было сделано в предыдущий раз. Это позволит сосредоточить внимание и восстановить в памяти уже имеющиеся знания по данному предмету. Кроме того, такой метод поможет лучше запомнить как старое, так и новое, углубит понимание того и другого, так как при этом устанавливаются связи нового со старым, что является не только обязательным, но и основным условием глубокого овладения материалом. Чем детальнее изучаемое ассоциируется с известным ранее, тем прочнее сохраняется в памяти.

Приступая к изучению нового материала, необходимо сосредоточиться, т.е. сконцентрировать внимание и не отвлекаться от выполняемой работы, помня, что желание запомнить является гарантией успешной работы, отсутствие же воли к запоминанию снижает эффект восприятия.

Записывать на лекции необходимо главное, не стремясь зафиксировать все слово в слово. Выбрать же главное без понимания предмета невозможно. Наличие собственного конспекта лекций позволяет еще раз ознакомиться, продумать, разобраться в новом материале, так как недостаточно хорошо понятые во время лекции положения могут быть

восстановлены в памяти, сопоставлены с другими, додуманы, дополнены, уяснены и расширены с помощью учебной литературы. Записи являются пособиями для повторения, дают возможность охватить содержание лекции и всего курса в целом.

При этом хорошо овладеть содержанием лекции – это:

- знать тему;
- понимать значение и важность ее в данном курсе;
- четко представлять план;
- уметь выделить основное, главное;
- усвоить значение примеров и иллюстраций;
- связать вновь полученные сведения о предмете или явлении с уже имеющимися;
- представлять возможность и необходимость применения полученных сведений.

Существует несколько общих правил работы на лекции:

- к прослушиванию лекций следует готовиться, что позволит в процессе лекции отделить главное от второстепенного;
- лекции необходимо записывать с самого начала, так как оно часто бывает ключом ко всей теме;
- так как дословно записать лекцию невозможно, то необходимо в конспекте отражать: формулы, определения, схемы, трудные места, мысли, примеры, факты и положения от которых зависит понимание главного, новое и незнакомое, неопубликованные данные, материал отсутствующий в учебниках и т.п.;
- записывать надо сжато;
- во время лекции важно непрерывно сохранять рабочую установку, умственную активность.

Изучение теоретического материала в данном курсе не ограничивается подготовкой к лекциям и работой на данном виде занятий. Лекционная часть курса органически взаимосвязана с иными видами работ: участием в лабораторных занятиях, подготовкой и сдачей экзамена по дисциплине, в структуре которых также большое значение имеет самостоятельная работа студента.

Лабораторное занятие является эффективной формой организации учебного процесса в высшем учебном заведении, которая основывается на самостоятельной работе студентов. Лабораторные занятия не только закрепляют теоретические знания, но и позволяют студенту глубоко изучать механизм применения этих знаний, овладевать важным для специалиста умением интеллектуального проникновения в те процессы, которые исследуют на лабораторном занятии. Под влиянием этой формы занятий студентов часто возникают новые идеи научного и технического характера, которые используются в курсовых, квалификационных, дипломных работах. Лабораторные занятия в значительной степени обеспечивают отработку умений и навыков принятия практических решений в научной и производственной деятельности.

Приступая к работе в лаборатории, студенту следует знать, что любое несоблюдение расписания занятий и дисциплины будет считаться нарушением его служебных обязанностей. Преподаватель, который впервые встречается со студентами на вводном занятии, должен ознакомить их с общими правилами работы в лаборатории, которые они обязаны неукоснительно выполнять.

Успех проведения конкретного лабораторного занятия зависит от его подготовки, которая включает: глубокое изучение студентами теоретического материала; подготовку необходимой учебно-материальной базы и документации (инструкций, методических разработок и т.п.); подготовку преподавателя и студентов.

Подготовку к лабораторному занятию осуществляют в несколько этапов: предварительная подготовка, начало работы, ее выполнение, составление отчета и оценки работы преподавателем.

Предварительную подготовку к работе в лаборатории осуществляют в отведенное для самостоятельной работы время. Готовясь к ней, студент прежде всего должен осоз-

нать ее цель, усвоить теоретический материал, добиться четкого представления о физических процессах, которые исследуются на лабораторном занятии.

С целью качественного выполнения лабораторной работы преподаватели проверяют готовность студентов. Это происходит в форме беседы с каждым студентом, в процессе которой выявляют знания теоретического материала по теме работы, или в форме компьютерного тестирования по этим же вопросам. Таким образом выявляют уровень теоретической подготовки студентов, практические навыки, умение применять знания для решения практических задач.

После экспериментальной части работы студенты должны ответить на контрольные вопросы, преподаватель использует для оценки знаний и экспериментальных умений и навыков студента при зачете его работы.

Завершается лабораторная работа оформлением индивидуального отчета и его защитой перед преподавателем. Итоговые оценки выставляют в журнале учета выполнения лабораторных работ и учитывают при проведении зачёта по дисциплине.

Следовательно, проведение занятия предусматривает следующие этапы: предварительный контроль подготовленности студентов к выполнению конкретной лабораторной работы; выполнения конкретных задач в соответствии с предложенной тематикой; оформление индивидуального отчета; оценивания преподавателем результатов работы студентов.

Самостоятельная работа обучающихся наряду с аудиторной представляет одну из форм учебного процесса и является существенной ее частью, что наиболее ярко представлено в процессе подготовки бакалавров. Последнее обусловлено тем, что самостоятельная работа предназначена для формирования навыков самостоятельной работы как вообще, так и в учебной, научной деятельности, формирование и развитие способности принимать на себя ответственность, самостоятельно решать проблему, находить конструктивные решения, выход из кризисной ситуации и т.д.

Самостоятельная работа формирует самостоятельность не только как совокупность умений и навыков, но и как черту характера, играющую существенную роль в структуре личности современного специалиста высшей квалификации. Она воспитывает самостоятельность как черту характера. Никакие знания, полученные на уровне пассивного восприятия, не ставшие объектом собственной умственной или практической работы, не могут считаться подлинным достоянием человека.

Давая возможность расширять и обогащать знания, умения по индивидуальным направлениям, самостоятельная работа обучающегося позволяет создать разносторонних специалистов. В процессе самостоятельной работы развивают творческие возможности обучающегося, при этом самостоятельная работа завершает задачи всех видов учебной работы.

Самостоятельная работа - это планируемая работа обучающихся, выполняемая по заданию и при методическом руководстве преподавателя, но без его непосредственного участия. Преподаватель, ведущий занятия, организует, направляет самостоятельную работу обучающихся и оказывает им необходимую помощь. Однако самостоятельность обучающихся должна превышать объем работы, контролируемой преподавателем работы, и иметь в своей основе индивидуальную мотивацию обучающегося по получению знаний, необходимых и достаточных для будущей профессиональной деятельности в избранной сфере. Преподаватель при необходимости может оказывать содействие в выработке и коррекции данной мотивации, лежащей в основе построения самостоятельной деятельности обучающегося по изучению дисциплины, получению необходимых знаний и навыков.

Результат обучения и самостоятельной работы обучающегося предполагает наличие следующих составляющих:

- понимание методологических основ построения изучаемых знаний;
- выделение главных структур учебного курса;
- формирование средств выражения в данной области;

- построение методик решения задач и ориентации в проблемах (ситуациях).

Самостоятельная работа студента при изучении дисциплины «Языки программирования высокого уровня» включает в себя:

изучение теоретической части курса	– 26 часов
подготовку к лабораторным занятиям	– 40 часов
написание отчетов по лабораторным работам	– 30 часов
итога	– 96 часов

Подготовка к экзамену – 36 часов

15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины

а) основная литература:

№ п/п	Источник
1.	Немнюгин С. А. Turbo Pascal. Программирование на языке высокого уровня / С. А. Немнюгин. СПб: Питер, 2007. – 543 с.
2.	Алексеев Е.Р. Free Pascal и Lazarus: Учебник по программированию / Е.Р. Алексеев, О.В. Чеснокова, Т.В. Кучер – М.: ALT Linux; Издательский дом ДМК-пресс, 2010. – 440 с.
3.	Зеленяк О. П. Практикум программирования на Turbo Pascal. Задачи, алгоритмы и решения / О.П. Зеленяк. – Москва : ДМК Пресс, 2009. – 311 с. // Электронно-библиотечная система. – URL : http://e.lanbook.com
4.	Павловская Т.А. Паскаль. Программирование на языке высокого уровня / Т.А. Павловская - СПб: Питер, 2006. – 392 с.
5.	Мансуров К.Т. Основы программирования в среде Lazarus. / К.Т. Мансуров - 2010. - 772 с.
6.	Дубровский О.И. Задачи по программированию. Ч. 1. Базовые алгоритмические конструкции. Практикум / О.И. Дубровский, Е.Р. Лихачев, С.И. Курганский. – Воронеж. Воронежский государственный университет, 2005. – 20 с.
7.	Дубровский О.И. Задачи по программированию. Ч. 2. Структуры данных. / О.И. Дубровский, Е.Р. Лихачев, С.И. Курганский. – Воронеж. Воронежский государственный университет, 2006. – 28 с.

б) дополнительная литература:

№ п/п	Источник
8.	Рапаков Г.Г. Программирование на языке Pascal / Г.Г. Рапаков, С.Ю. Ржеуцкая – СПб.: БХВ-Петербург, 2004. – 480 с.
9.	Фаронов В.В. Турбо Паскаль 7.0. Практика программирования / В.В. Фаронов – М.: Нолидж, 2003, - 415 с.
10.	Программирование на языке Паскаль / О.Ф. Ускова [и др.]; под ред. О.Ф. Усковой – СПб. [и др.] : Питер, 2005. – 333 с.
11.	Программирование на языке Паскаль : задачник / О.Ф. Ускова [и др.]; под ред. О.Ф. Усковой – СПб. [и др.] : Питер, 2003. – 336 с.
12.	Бежанова М.М. Практическое программирование. Приемы создания программ на языке Паскаль / М.М. Бежанова, Л.А. Москвина – М.: Научный мир, 2000. – 270 с..
13.	Ставровский А. TurboPascal 7.0 / А Ставровский - Киев:ВНУ: Ирина, - 2000, 399 с.
14.	Грызлов В.И. Турбо Паскаль 7.0 / В.И. Грызлов, Т.П. Грызлова – М.: ДМК, 1998, - 400 с.

в) информационные электронно-образовательные ресурсы:

№ п/п	Источник
15.	http://www.lib.vsu.ru – Зональная научная библиотека ВГУ
16.	http://www.moodle.vsu.ru - Образовательный портал электронного университета ВГУ.
17.	https://lanbook.com – ЭБС «Лань»
18.	https://biblioclub.ru – ЭБС «Университетская библиотека онлайн»
19.	http://www.intuit.ru - Национальный Открытый Университет «ИНТУИТ»
20.	http://ips.ifmo.ru/courses/pascal - Курс лекций "Язык программирования Pascal".
21.	www.lazarus.org

16. Перечень учебно-методического обеспечения для самостоятельной работы

№ п/п	Источник
1.	Дубровский О.И. Задачи по программированию. Ч. 1. Базовые алгоритмические конструкции. Практикум / О.И. Дубровский, Е.Р. Лихачев, С.И. Курганский. – Воронеж. Воронежский государственный университет, 2005. – 20 с.
2.	Дубровский О.И. Задачи по программированию. Ч. 2. Структуры данных. / О.И. Дубровский, Е.Р. Лихачев, С.И. Курганский. – Воронеж. Воронежский государственный университет, 2006. – 28 с.

17. Образовательные технологии, используемые при реализации учебной дисциплины, включая дистанционные образовательные технологии (ДОТ), электронное обучение (ЭО), смешанное обучение):

При реализации дисциплины используются активные и интерактивные методы и технологии профессионального обучения.

18. Материально-техническое обеспечение дисциплины:

Лабораторные работы выполняются в компьютерном классе физического факультета, оснащенном 14-ю персональными компьютерами с установленным программным обеспечением (интегрированная среда программирования Free Pascal, среда визуального программирования Lazarus).

19. Оценочные средства для проведения текущей и промежуточной аттестаций

Порядок оценки освоения обучающимися учебного материала определяется содержанием следующих разделов дисциплины:

№ п/п	Наименование раздела дисциплины (модуля)	Компетенция(и)	Индикатор(ы) достижения компетенции	Оценочные средства
1.	Основные понятия пользовательского программирования	УК-1	УК-1.2	Устный опрос
2.	Основы алгоритмического языка Паскаль	УК-1	УК-1.2	Устный опрос
3.	Реализация базовых алгоритмических конструкций на языке Паскаль.	УК-1 ПКВ-1	УК-1.2 ПКВ-1.2	Лаб. работы 1 – 5
4.	Структуры данных.	УК-1 ПКВ-1	УК-1.2 ПКВ-1.2	Лаб. работы 6 – 10
5.	Структурное и модульное программирование на языке Паскаль	УК-1 ПКВ-1	УК-1.2 ПКВ-1.2	Лаб. работы 11 – 13
6.	Организация ввода-вывода в Паскале. Работа с файлами	УК-1 ПКВ-1	УК-1.2 ПКВ-1.2	Лаб. работы 14, 15
7.	Динамические структуры данных	УК-1 ПКВ-1	УК-1.2 ПКВ-1.2	Лаб. работы 16, 17
8.	Программирование некоторых классических алгоритмов и задач на Паскале.	УК-1 ПКВ-1	УК-1.2 ПКВ-1.2	Лаб. работы 18, 19
9.	Основы объектно-ориентированного программирования	УК-1 ПКВ-1 ПКВ-2	УК-1.2 ПКВ-1.2 ПКВ-2.1	Лаб. работы 20, 21
10.	Среда визуального программирования Lazarus.	УК-1 ПКВ-1 ПКВ-2	УК-1.2 ПКВ-1.2 ПКВ-2.1	Лаб. работы 22
Промежуточная аттестация форма контроля - экзамен				Комплект КИМ

20 Типовые оценочные средства и методические материалы, определяющие процедуры оценивания

20.1 Текущий контроль успеваемости

Текущий контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств: отчеты о выполнении лабораторных работ.

Перечень тем лабораторных работ

Лабораторная работа 1. Линейная программа. Вычисление логических выражений

Лабораторная работа 2. Программирование ветвлений

- Лабораторная работа 3. Использование цикла с параметром
- Лабораторная работа 4. Использование цикла с условием
- Лабораторная работа 5. Вложенные циклы
- Лабораторная работа 6. Обработка одномерных массивов
- Лабораторная работа 7. Работа с двумерными массивами
- Лабораторная работа 8. Обработка строк текста
- Лабораторная работа 9. Работа с записями
- Лабораторная работа 10. Работа с множествами
- Лабораторная работа 11. Использование подпрограмм.
- Лабораторная работа 12. Использование стандартных модулей
- Лабораторная работа 13. Создание пользовательского модуля
- Лабораторная работа 14. Создание и работа с текстовым файлом
- Лабораторная работа 15. Создание и работа с типизированным файлом
- Лабораторная работа 16. Использование динамических массивов
- Лабораторная работа 17. Использование связанных списков
- Лабораторная работа 18. Алгоритмы поиска
- Лабораторная работа 19. Алгоритмы сортировки
- Лабораторная работа 20. Создание классов
- Лабораторная работа 21. Создание программы с использованием наследования
- Лабораторная работа 22. Визуальное программирование в среде Lazarus

Для текущего контроля успеваемости используется устный опрос, отчеты о ходе выполнения лабораторных работ, на основе которых выставляется предварительная оценка *отлично/хорошо/удовлетворительно/неудовлетворительно*.

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала предварительных оценок
Обучающийся в полной мере владеет понятийным аппаратом данной области науки (теоретическими основами дисциплины), способен иллюстрировать ответ примерами, фактами, применять теоретические знания при выполнении лабораторных работ	Повышенный уровень	<i>Отлично</i>
Обучающийся владеет понятийным аппаратом данной области науки (теоретическими основами дисциплины), допускает незначительные ошибки при выполнении лабораторных задач	Базовый уровень	<i>Хорошо</i>
Обучающийся владеет частично теоретическими основами дисциплины, фрагментарно способен выполнять лабораторные задания	Пороговый уровень	<i>Удовлетворительно</i>
Неудовлетворительное выполнение тестовых заданий. Обучающийся демонстрирует отрывочные, фрагментарные знания, допускает грубые ошибки при выполнении лабораторных работ	–	<i>Неудовлетворительно</i>

20.2 Промежуточная аттестация

Промежуточная аттестация по дисциплине осуществляется с помощью следующих оценочных средств:

Комплект КИМ

Контрольно-измерительный материал № 1

1. Структурированные типы данных (*строки, операции над строками, строковые процедуры и функции*).
2. Модули (структура модулей).

Контрольно-измерительный материал № 2

1. Простые типы данных и константы языка Паскаль.
2. Динамические переменные.

Контрольно-измерительный материал № 3

1. Простые типы данных (целые, вещественные, логические)
2. Основные возможности модуля CRT.

Контрольно-измерительный материал № 4

1. Структурированные типы данных (запись, файл).
2. Процедуры пользователя.

Контрольно-измерительный материал № 5

1. Простые типы данных (символьный, перечисляемый, тип-диапазон).
2. Полиморфизм

Контрольно-измерительный материал № 6

1. Структурированные операторы (оператор цикла с параметром, сравнение работы операторов цикла).
2. Классы и объекты

Контрольно-измерительный материал № 7

1. Выражения, операнды, операции (операции со строками, операции с множествами, операции отношения)
2. Инкапсуляция

Контрольно-измерительный материал № 8

1. Структурированные операторы (составной, условный).
2. Стандартные процедуры и функции.

Контрольно-измерительный материал № 9

1. Структурированные операторы (оператор выбора, оператор над записями).
2. Функции пользователя.

Контрольно-измерительный материал № 10

1. Структура программы на Паскале.
2. Наследование.

Контрольно-измерительный материал № 11

1. Структурированные типы данных (записи, оператор над записями).
2. Локализация имен. Глобальные и локальные объекты. Побочные эффекты при использовании подпрограмм.

Описание технологии проведения промежуточной аттестации

Промежуточная аттестация по дисциплине – экзамен. В приложение к диплому вносится оценка *отлично/хорошо/удовлетворительно*.

Оценка уровня освоения дисциплины «Языки программирования высокого уровня» осуществляется по следующим показателям:

- качество и своевременность выполнения лабораторных работ;
- полнота ответов на вопросы контрольно-измерительного материала;
- полнота ответов на дополнительные вопросы.

Критерии оценки освоения дисциплины «Языки программирования высокого уровня»:

– оценка *отлично* выставляется при полном соответствии работы студента всем вышеуказанным показателям. Соответствует высокому (углубленному) уровню сформированности компетенций: компетенции сформированы полностью, проявляются и используются систематически, в полном объеме. Данный уровень превосходит, по крайней мере, по одному из перечисленных выше показателей повышенный (продвинутый) уровень;

– оценка *хорошо* выставляется в случае, если работа студента при освоении дисциплины не соответствует одному из перечисленных показателей или в случае предоставления курсовых работ и отчетов по лабораторным работам позже установленного срока. Соответствует повышенному (продвинутому) уровню сформированности компетенций: компетенции в целом сформированы, но проявляются и используются фрагментарно, не в полном объеме. Данный уровень превосходит, по крайней мере, по одному из перечисленных выше показателей пороговый (базовый) уровень;

– оценка *удовлетворительно* выставляется в случае, если работа студента при освоении дисциплины не соответствует любым двум из перечисленных показателей. Соответствует пороговому (базовому) уровню сформированности компетенций: компетенции сформированы в общих чертах, проявляются и используются ситуативно, частично. Данный уровень обязателен для всех осваивающих основную образовательную программу;

– оценка *неудовлетворительно* выставляется в случае несоответствия работы студента всем показателям, его неорганизованности, безответственности и низкого качества работы при выполнении лабораторных работ, предусмотренных программой дисциплины.

Факт невыполнения требований, предъявляемых к студенту при освоении дисциплины «Языки программирования высокого уровня» и отраженных в вышеперечисленных критериях, фиксируется в ведомости оценкой *неудовлетворительно*.

Если студент не осваивает дисциплину в установленном программой объеме и в сроки, определенные графиком учебного процесса, он не допускается к промежуточной аттестации по данному виду учебной работы.

20.3 Фонд оценочных средств сформированности компетенций студентов, рекомендуемый для проведения диагностических работ:

ПК-1: Способен выполнять расчет и проектирование электронных приборов, схем и устройств различного функционального назначения в соответствии с техническим заданием с использованием средств автоматизации проектирования

Индикатор: ПК-1.2

ПК-2: Способен выполнять моделирование элементов интегральных схем и принимать решения об уточнении первичного схемотехнического описания на основе результатов анализа и верификации результатов моделирования

Индикатор: ПК-2.1

Перечень заданий для проверки сформированности компетенции:

1) Закрытые задания (тестовые, средний уровень сложности):

1. Определить значение следующего выражения:

$c \text{ and } (b > 5) \text{ and } (a < 20) \text{ and } (b < 30)$

если $a = 10, b = 20, c = \text{true}$

а) false

б) 30

в) 0.0

г) true

2. Определить значение следующего выражения:

$[2, 5..8] \geq [1..3, 8]$

а) [2,8]

б) true

в) false

г) [1..3,5..8]

3. Определить значение следующего выражения:

$[7..12] * [9..18]$

а) [7,8]

б) true

в) false

г) [9..12]

4. Каким будет значение переменной *a* после выполнения операторов:

```
a:=3;  
if a>4 then inc(a,2); inc(a,3);
```

- a) 3 б) 8 **в) 6** г) 5

5. Сколько раз будет выполнен оператор $n := n + 1$ в ходе выполнения оператора

```
while B do n:=n+1,
```

если *n* – это переменная типа *integer*, а $B = (n > 0) \text{ and } (n < 0)$;

- a) 1 **б) 0** в) 2 г) 2147483647

6. Описание функции *a* имеет вид:

```
function a (b,c : integer): integer;  
  var m,i: integer;  
begin  
  m := 0;  
  for i := b to c do  
    if i > 0 then m := m + sqr(i);  
  a := m;  
end;
```

Какое значение будет присвоено переменной *F*, если в основной программе имеется оператор $F := a(-3, 6)$;

- a) 15 б) 21 **в) 91** г) 105

7. Как называется алгоритмическая структура, которая в зависимости от истинности заданного условия определяет выбор для исполнения одной из двух групп операторов?

- a) следование б) повторение **в) ветвление** г) неполная альтернатива

8. Как называется автономно компилируемая программная единица, включающая в себя различные компоненты раздела описаний?

- a) функция **б) модуль** в) процедура г) файл

9. С помощью какой процедуры файл открывается для чтения?

- a) assign **б) reset** в) rewrite г) append

10. Какую процедуру нельзя использовать при работе с типизированным файлом?

- a) read б) write в) rewrite **г) readln**

2) Открытые задания (тестовые, повышенный уровень сложности):

1. Определить значение следующего выражения:

```
not true or false
```

Ответ: **false**

2. Запишите на языке *pascal* выражение, истинное при выполнении указанного условия и ложное в противном случае: числа *x*, *y*, *z* равны между собой

Ответ: **(x = y) and (x = z)**

3. Заменить оператором присваивания следующий условный оператор

```
if a then x := true  
  else if b then x:=c  
      else x:=false;
```

Ответ: **a or b and c**

4. Какое описание могло бы быть для переменной b, если она используется в операторе

```
b ['m',-2] := 'october';
```

Ответ: **var b: array[char,-10..0] of string;**

5. Упростите выражение $A*B - A$ (A и B – переменные типа-множество)/

Ответ: []

6. Определить значения переменных p и q после выполнения операторов, если переменная k имеет значение 6:

```
p := true; q := 1;
case k mod 10 of
  3,2,7,5 : q := k;
  1:;
  4,8: begin
          p := false;   q := 2
        end;
  9,6: begin
          p := false;   q := 3
        end;
end;
```

Ответ: **p = false, q = 3**

7. Что будет выведено на экран монитора в результате работы программы:

```
var i, k: integer;
begin
  k := 2;
  for i:=k to k +2 do
    begin
      k := k+1; writeln(i, ' ', k)
    end
end.
```

Ответ: **2 3**

3 4

4 5

8. Что будет выведено на экран монитора в результате работы программы:

```
const a:array[1..8] of char = ('a','b','c','r','c','a','a','b');
var i: integer; k: string;
begin
  k := ' ';
  for i:=8 downto 2 do
    if a[i] < 'c' then k := k + a[i];
  write(k)
end.
```

Ответ: **baab**

9. Определите, что будет выведено на экран монитора в результате работы следующей программы:

```
program ex1;
var zn : integer;
```

```

function p (n : integer) : integer;
  var an, i : integer;
  begin
    an := 1;
    for i := 1 to n do
      an:= an * i;
    p := an;
  end;
begin
  zn := p (5);
  writeln('zn = ', zn);
end.

```

Ответ: **zn = 32**

10. Определите, что будет выведено на экран монитора в результате работы следующей программы:

```

var a: string;
  procedure P(a: char; var d: char);
  begin
    a := 'в'; d := 'с';
  end;
begin
  a := 'глаз';
  P(a[1], a[4]);
  write(a)
end.

```

Ответ: **глас**

3) Открытые задания (повышенный уровень сложности):

1. Какую задачу решает следующий фрагмент программы?

```

var x, y, z, t: real;
  if x>y then z:=0
    else z:=y-x;
  t:=x+z;.

```

.....*Ответ:*

Находит максимальное из двух чисел, хранящихся в переменных x и y и помещает его в переменную t.

2. Каково действие оператора?

```

if false then x := 1

```

.....*Ответ:*

Не выполняет никаких действий, т.е. трактуется как пустой оператор.

3. Описать и заполнить 26-элементный массив всеми буквами латинского алфавита так, чтобы индексом и значением каждого элемента была соответствующая буква.

.....*Ответ:*

```

var c:char; mas: array['a'..'z'] of char;
begin
  for c := 'a' to 'z' do
    mas[c] := c;
  end.

```


4. Написать программу, определяющую количество цифр в натуральном числе n.

.....*Ответ:*

```
var num,i:integer;
begin
  readln(num);
  i:=0;
  while num<>0 do
  begin
    num:=num div 10;
    i:=i+1;
  end;
  writeln ('number of digits: ', i);
end.
```

5. Восстановить условие задачи, которую решает следующая программа:

```
program symbols;
var k0, k1, k2, n: integer; ch:char;
begin
  K0:=0; K1:=0; K2:=0;
  Read(ch);
  while ch<>'.' do
  begin
    N:=ord(ch) - ord('A');
    Case n of
      0: k0:=k0+1;
      1: k1:=k1+1;
      2: k2:=k2+1
    end;
    read(ch)
  end;
  readln;
  writeln(k0,k1,k2)
end.
```

.....*Ответ:*

Поочередно вводятся символы. Программа подсчитывает количество введенных букв А, В и С. Анализ последовательности заканчивается при вводе символа '.'.

6. Сформулируйте назначение следующей процедуры

```
procedure C(x,y,z: real; var p: integer);
begin
  if sqr(x) + sqr(y) <= sqr(z) then p:=1
  else p:=0
end;
```

.....*Ответ:*

Определяет, попадает ли точка с координатами x и y внутрь круга радиусом z. Если попадает, то переменной p присваивается значение 1, в противном случае – 0.

7. Сформулируйте назначение следующей функции:

```
function a (b,c : integer): integer;
  var m,i: integer;
begin
  m := 0;
```

```

    for i := b to c do
        if i > 0 then m := m + sqr(i);
    a := m;
end;

```

.....*Ответ:*

Вычисляет сумму квадратов целых чисел, лежащих в интервале от b до c.

8. Сформулируйте назначение следующей функции:

```

function b (a,c : integer): integer;
var d,j : integer;
begin
    d := 0;
    for j := a to c do
        if j mod 2 = 0 then d := d + sqr( j );
    b := d;
end;

```

.....*Ответ:*

Вычисляет сумму квадратов чётных чисел, лежащих в интервале от a до c.

9. Что описывается в интерфейсной части модуля?

.....*Ответ:*

Интерфейсная часть открывается зарезервированным словом Interface. В этой части содержатся объявления всех глобальных объектов модуля (типов, констант, переменных и подпрограмм), которые должны стать доступными основной программе и/или другим модулям. При объявлении глобальных подпрограмм в интерфейсной части указывается только их заголовок.

10. Какая задача решается с помощью следующей программы:

```

program WhatIsIt;
var
    f: file of integer;
    i: integer;
begin
    assign(f, 'pr.dat');
    reset (f);
    while not Eof(f) do
        begin
            read(f,i);
            writeln(i);
        end;
    close(f)
end.

```

.....*Ответ:*

Выводит содержимое типизированного файла целых чисел на экран монитора.

Критерии и шкалы оценивания заданий ФОС:

Для оценивания выполнения заданий используется балльная шкала:

1) закрытые задания (тестовые, средний уровень сложности):

- 1 балл –указан верный ответ;
- 0 баллов – указан неверный ответ (полностью или частично неверный)

2) открытые задания (тестовые, повышенный уровень сложности).

- 2 балла – указан верный ответ,

- 0 баллов – указан неверный ответ (полностью или частично неверный)

3} открытые задания (повышенный уровень сложности):

- 5 баллов – задание выполнено верно (получен правильный ответ, обоснован (аргументирован) ход выполнения (при необходимости));
- 2 балла – выполнение задания содержит незначительные ошибки, но приведен правильный ход рассуждений, или получен верный ответ, но отсутствует обоснование хода его выполнения (если оно было необходимым), или задание выполнено неполностью, но получены промежуточные (частичные) результаты, отражающие правильность хода выполнения задания, или в случае, если задание состоит из нескольких подзаданий, верно выполнено 50% таких подзаданий;
- 0 баллов – задание не выполнено или выполнено неверно (получен неправильный ответ, ход выполнения ошибочен или содержит грубые ошибки).

ЛИСТ СОГЛАСОВАНИЙ

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Направление: 11.03.04 Электроника и наноэлектроника
шифр и наименование направления/специальности

Дисциплина: Б1.В.10 Языки программирования высокого уровня
код и наименование дисциплины

Профиль подготовки: Интегральная электроника и наноэлектроника
в соответствии с Учебным планом

Форма обучения: очная

Учебный год: 2020-2021

Ответственный исполнитель -

<u>Зав.кафедрой ФТТиНС</u> <i>должность, подразделение</i>	_____	<u>(Э.П. Домашевская)</u> <i>расшифровка подписи</i>	01.09.2020
---	-------	---	------------

Исполнители:

<u>Доцент каф. ФТТиНС</u> <i>должность, подразделение</i>	_____	<u>(О.И. Дубровский)</u> <i>расшифровка подписи</i>	01.09.2020
--	-------	--	------------

_____	_____	_____	__ . __ 20__
-------	-------	-------	--------------

должность, подразделение

подпись

расшифровка подписи

СОГЛАСОВАНО:

Куратор ООП ВО направления 11.03.04	_____	<u>(Г.В. Быкадорова)</u> <i>расшифровка подписи</i>	01.09.2020
--	-------	--	------------

подпись

Зав.отделом обслуживания ЗНБ	_____	<u>(Н.В. Белодедова)</u> <i>расшифровка подписи</i>	01.09.2020
---------------------------------	-------	--	------------

подпись

Рекомендована НМС физического факультета, протокол № 4 от 01.09.2020
(наименование факультета, структурного подразделения)