

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

УТВЕРЖДАЮ

Заведующий кафедрой
Ядерной физики



Кадменский С.Г.

17.06.2021 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
Б1.О.28 Прикладное программное обеспечение

1. Шифр и наименование направления подготовки:

03.03.02 Физика

2. Профиль подготовки/специализации: Ядерная физика

3. Квалификация (степень) выпускника: бакалавр

4. Форма образования: очная

5. Кафедра, отвечающая за реализацию дисциплины: кафедра ядерной физики

6. Составители программы: к.ф.м.н., доцент Любашевский Дмитрий Евгеньевич, к.ф.м.н., доцент Титова Лариса Витальевна

7. Рекомендована: Научно-методическим советом физического факультета, протокол № 6 от 17.06.2021

РП продлена на 2022-2023 учебный год НМС физического факультета 14.06.2022, протокол №6.

(отметки о продлении вносятся вручную)

8. Учебный год: 2021-2022

Семестр(ы): 1

9. Цели и задачи учебной дисциплины:

Целями освоения учебной дисциплины являются:

- приобретение базовых знаний и навыков в области практики классического программирования, знакомство с основными принципами и подходами к программированию, формирование культуры разработки программных продуктов, обучение работе с научно-технической литературой и технической документацией по программному обеспечению ПЭВМ.

Задачи учебной дисциплины:

- получить практические навыки работы с современными визуальными средами программирования;
- получить навыки проектирования программ со сложным графическим интерфейсом.

10. Место учебной дисциплины в структуре ООП:

учебная дисциплина относится к обязательной части блока Б1.

Это первая дисциплина, изучаемая в области информатики и программирования. «Прикладное программное обеспечение» является предшествующей для следующих дисциплин:

- Новые информационные технологии в науке и образовании; Системы программного обеспечения;
- Компьютерные технологии в науке и образовании;
- Вычислительная физика (Практикум на ЭВМ);
- Специальный компьютерный практикум;

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
ОП К-3	<i>Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности</i>	ОПК-3.1	Знает основные положения теории информации, принципов построения систем обработки и передачи информации, основы подхода к анализу информационных процессов; современные аппаратные программные средства вычислительной техники, принципы организации информационных систем, современные информационные технологии	Знать: понятие алгоритма и структуры управления; традиционные структуры данных; идеи, лежащие в основе процедурного программирования, реализацию вызова процедур в языках с блочной структурой, рекурсию; Уметь: применять требования методологии структурного программирования при проектировании информационных моделей; разрабатывать и записывать на языке высокого уровня алгоритмы решения классических задач программирования; Владеть: навыками практического программирования конкретных задач в определенной языковой среде; применять Средства структурного, модульного и объектно-ориентированного программирования для решения задач.
		ОПК-3.2	Владеет навыками работы с компьютером, использует современные информационно-коммуникационные технологии в	Знать: традиционные структуры данных; основные требования методологии структурного программирования, как Технологической основы разработки качественных программных компонентов; Уметь: применять требования методологии структурного

			<p>профессиональной деятельности; использовать информационные технологии для решения физических задач;</p>	<p>программирования при проектировании информационных моделей; разрабатывать и записывать на языке высокого уровня алгоритмы решения классических задач программирования; реализовывать технологию проектирования сверху-вниз; выбирать оптимальную структуру для представления данных.</p> <p>Владеть: навыками практического программирования конкретных задач в определенной языковой среде; применять средства структурного, модульного и объектно-ориентированного программирования для решения задач.</p>
		ОПК-3.3	<p>Использует современные информационные технологии и программное обеспечение при решении задач профессиональной деятельности</p>	<p>Знать: основные этапы компьютерного решения задач; основные требования методологии структурного программирования, как технологической основы разработки качественных программных компонентов;</p> <p>Уметь: разрабатывать и записывать на языке высокого уровня алгоритмы решения классических задач программирования; реализовывать технологию проектирования сверху-вниз; выбирать оптимальную структуру для представления данных.</p> <p>Владеть: навыками практического программирования конкретных задач в определенной языковой среде; применять средства структурного, модульного и объектно-ориентированного программирования для решения задач.</p>
		ОПК-3.4	<p>Соблюдает требования информационной безопасности при использовании современных информационных технологий и программного обеспечения.</p>	<p>Знать: основные этапы компьютерного решения задач; основные требования методологии структурного программирования, как технологической основы разработки качественных программных компонентов;</p> <p>Уметь: разрабатывать и записывать на языке высокого уровня алгоритмы решения классических задач программирования; реализовывать технологию проектирования сверху-вниз; выбирать оптимальную структуру для представления данных.</p> <p>Владеть: навыками практического программирования конкретных задач в определенной языковой среде; применять средства структурного, модульного и объектно-ориентированного программирования для решения задач.</p>

12. Объем дисциплины в зачетных единицах/час.: 3 /108.

Форма промежуточной аттестации зачет

13. Трудоемкость по видам учебной работы

Вид учебной работы	Трудоемкость			
	Всего	По семестрам		
		№ семестра 1	№ семестра	...
Аудиторные занятия	54	54		
в том числе: лекции	18	18		
практические				
лабораторные	36	36		
Самостоятельная работа	54	54		
Форма промежуточной аттестации (зачет – 0 час. / экзамен – час.)				
Итого	108	108		

13.1. Содержание дисциплины

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины
1. Лекции		
1	Введение	Основные понятия информатики; модели решения функциональных и вычислительных задач; алгоритмизация и программирование; языки программирования высокого уровня; структура программного обеспечения. Основные этапы компьютерного решения задач. Критерии качества программы. Обрабатываемые данные. Управляющие структуры. Метод последовательного уточнения действий. Основные идеи структурного программирования.
2	Языки программирования. Программы.	Постановка задачи и спецификация программы; способы записи алгоритма. Концепция языка высокого уровня. Словарь и синтаксис языка. Основные понятия языка: алфавит языка, имена и идентификаторы (служебные и пользовательские), описания (объявления), операторы, процедуры и функции.
3	Концепция данных. Классификация типов данных.	Концепция данных: определение данных, константы и переменные, концепция типов значений, объединение данных в структуры, простые и сложные (производные) типы данных. Классификация типов данных. Константы: простые и типизированные (структурные).

4	Простые стандартные типы данных	Целочисленный; вещественный: с фиксированной точкой, с плавающей точкой, диапазон и точность. Основные операции, выражения, стандартные функции. Символьный (литерный) тип; логический тип. Основные операции, применимые к указанным типам, выражения, стандартные функции.
5	Структура программы. Ввод и вывод данных.	Структура программы. Правила записи текста программы. Ввод и вывод данных. Комментарии и их использование.
6	Операторы языка	Оператор присваивания, логический оператор присваивания. Примеры и задачи. Условный оператор, оператор безусловного перехода. Примеры и задачи. Операторы циклов. Примеры и задачи.
7	Сложные типы данных: массивы	Одномерные массивы (типы индексов, функциональное назначение, представление в памяти). Основные операции над массивами, понятие статического и динамического массива. Многомерные массивы (типы индексов, функциональное назначение, представление в памяти). Примеры задач с использованием массивов, в том числе: <ul style="list-style-type: none"> - поиск в массиве; - частота повторения символов в тексте, коды которых упорядочены; - упорядочение данных в массиве и проверка упорядоченности и т.д. Элементарные понятия сложности алгоритмов.
8	Процедуры и функции	Понятие подпрограммы. Процедуры и функции. Понятие процедуры и функции, отличия, функциональное назначение, синтаксис, вызов процедуры и функции. Способы обмена данными с процедурой и функцией. Параметры, способы передачи параметров. Области видимости переменных. Локальные и глобальные переменные. Время жизни переменных (динамика). Рекурсия. Процедурный тип данных. Параметры-процедуры и параметры-функции.
9	Строковые типы данных	Строковые типы данных. Функциональное назначение, представление в памяти, Основные процедуры и функции. Примеры использования.
10	Нестандартные типы данных	Перечислимый тип, его использование, стандартные функции. Оператор выбора. Ограниченный тип (диапазон).
11	Сложные типы данных: множества	Задание множественного типа и множественная переменная. Внутреннее представление множеств. Операции над множествами: объединение, пересечение, разность. Отношения: равенство, неравенство, включение. Проверка принадлежности к множеству. Примеры

		использования множественного типа.
12	Сложные типы данных: записи	Записи: простейшие записи, иерархические записи. Оператор присоединения. Записи с вариантами. Внутреннее представление, функциональное назначение и примеры использования.
13	Работа с внешними данными (файлы)	Файловый тип. Файл, базовый тип. Типы файлов: типизированные, текстовые, нетипизированные файлы. Действия над файлами: создание файла, просмотр файла. Организация прямого доступа к типизированному файлу: добавление и редактирование данных. Копирование файлов. Стандартные процедуры. Слияние отсортированных файлов. Примеры работы с нетипизированными файлами. Текстовые файлы. Структура текстового файла: признак конца строки. Стандартные текстовые файлы. Стандартные процедуры для работы с текстовыми файлами. Примеры работы с текстовыми файлами.
14	Культура разработки программного обеспечения	Примеры хорошей и плохой практики программирования
2. Лабораторные работы		
1	Простые стандартные типы данных	Целочисленные типы данных; вещественный: с фиксированной точкой, с плавающей точкой, диапазон и точность. Основные операции, выражения, стандартные функции. Символьный (литерный) тип; логический тип. Основные операции, применимые к указанным типам, выражения, стандартные функции.
2	Структура программы. Ввод и вывод данных.	Структура программы. Правила записи текста программы. Операторы ввода и вывода данных. Комментарии и их использование в программе.
3	Операторы языка	Оператор присваивания, логический оператор присваивания. Примеры и задачи. Условный оператор, оператор безусловного перехода. Примеры и задачи. Оператор множественного выбора. Операторы циклов. Примеры и задачи.
4	Сложные типы данных: массивы	Одномерные массивы. Основные операции над массивами: Примеры задач с использованием массивов: <ul style="list-style-type: none"> - создание, вывод на экран; - поиск элементов в массиве; - частота повторения элементов в массиве (частотный анализ); - упорядочение данных в массиве и проверка упорядоченности и т.д. - оценка быстродействия алгоритмов сортировки массивов. Многомерные массивы. Примеры задач с использованием массивов: <ul style="list-style-type: none"> - создание, вывод на экран; - поиск в массиве;

		<ul style="list-style-type: none"> - упорядочение данных в массиве; - применение многомерных массивов в линейной алгебре (операции над векторами и матрицами)
5	Процедуры и функции	Стандартные процедуры и функции и их применение. Создание пользовательских процедур и функций, вызов процедур и функций. Локальные и глобальные переменные.
6	Строковые типы данных	Строковые типы данных. Основные процедуры и функции. Примеры использования.
7	Нестандартные типы данных	Перечислимый тип, его использование, стандартные функции. Оператор выбора. Ограниченный тип (диапазон).
8	Сложные типы данных: множества	Задание множественного типа и множественная переменная. Внутреннее представление множеств. Операции над множествами: объединение, пересечение, разность. Отношения: равенство, неравенство, включение. Проверка принадлежности к множеству. Примеры использования множественного типа.
9	Сложные типы данных: записи	Записи: простейшие записи, иерархические записи. Записи с вариантами. Примеры использования.

13.2. Темы (разделы) дисциплины и виды занятий

№ п / п	Наименование темы (раздела) дисциплины	Виды занятий (часов)				Всего
		Лекци и	Практическ и е	Лабораторн ые	Самостоятель ная работа	
1	Введение	1		0	1	2
2	Языки программирования. Программы.	1		0	1	2
3	Концепция данных. Классификация типов данных.	1		0	3	4
4	Простые стандартные типы данных	1		2	2	5
5	Структура программы. Ввод и вывод данных.	1		2	3	6
6	Операторы языка	1		6	4	11
7	Сложные типы данных: массивы	2		8	5	15
8	Процедуры и функции	3		4	6	13
9	Строковые типы данных	2		4	4	10
10	Нестандартные типы данных	1		2	5	8
11	Сложные типы данных: множества	1		2	6	9
12	Сложные типы данных: записи	1		2	5	8
13	Работа с внешними данными (файлы)	1		4	6	11
14	Культура разработки программного обеспечения	1		0	3	4
	Итого:	18		3 6	5 4	10 8

14. Методические указания для обучающихся по освоению дисциплины

(рекомендации обучающимся по освоению дисциплины: работа с конспектами лекций,

презентационным материалом, выполнение практических заданий, тестов, заданий текущей аттестации и т.д.)

1. работа с конспектами лекций,
2. выполнение практических заданий, тестов

3. выполнение практических заданий, тестов, заданий текущей аттестации.

Изучение дисциплины «Информатика» 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; практические занятия; индивидуальные занятия; контрольные работы. 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.).

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 и быстрее вспомнить, когда требуется.

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.

Следует помнить о том, что через лекцию передается не только систематизированный теоретический материал, но и постигается методика научного исследования и умение самостоятельно работать, анализировать различного рода явления.

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:
 - лекции по каждому предмету записывать удобнее в отдельных тетрадях, оставляя широкие поля для пометок;
 - 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: написанием рефератов, выполнением лабораторных работ, подготовкой и сдачей зачета по дисциплине, в структуре которых также большое значение имеет самостоятельная работа студента.

Самостоятельная работа студентов 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 студента 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 студентов, выполняемая по заданию и при методическом руководстве преподавателя, но без его непосредственного участия. Преподаватель, ведущий занятия, организует, направляет самостоятельную работу студентов и оказывает им необходимую помощь. Однако самостоятельность студентов 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 студента по изучению дисциплины, получению необходимых знаний и навыков.

Получение образования предполагает обучение решению задач определенной сферы деятельности. Однако как бы хорошо не обучались учащиеся способам решения задач в аудитории, сформировать средства практической деятельности не удастся, так как каждый случай практики особый и для его решения следует выработать особый профессиональный стиль мышления.

Основой самостоятельной работы служит научно-теоретический курс, комплекс полученных студентом знаний. 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.

Результат обучения и самостоятельной работы студента 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 «Информатика» включает в себя:

подготовку и участие в изучении теоретической части курса, подготовку к лабораторным занятиям, подготовку к экзамену.

15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины (список литературы оформляется в соответствии с требованиями ГОСТ и используется общая сквозная нумерация для всех видов источников)

а) основная литература:

№ п/п	Источник
1	Алексеев Е. Программирование на Free Pascal и Lazarus/ Е.Алексеев, О.Чеснокова, Т.Кучер.– М.: Национальный Открытый Университет «ИНТУИТ», 2016.– 552 с. // «Университетская библиотека online: электронно-библиотечная система.– URL: http:// biblioclub.ru »
2	Программирование и основы алгоритмизации: учебное пособие / В. К.Зольников, П. Р.Машевич, В. И.Анциферова, Н. Н.Литвинов – Воронеж: Воронежская государственная лесотехническая академия, 2011.– 341 с. // «Университетская библиотека online: электронно-библиотечная система.– URL: http:// biblioclub.ru »
3	Ачкасов В. Программирование на Lazarus/ В.Ачкасов.– М.: Национальный Открытый Университет «ИНТУИТ», 2016.– 521 с. // «Университетская библиотека online: электронно-библиотечная система.– URL: http:// biblioclub.ru »
4	Семакин И.Г. Основы алгоритмизации и программирования : [учебник для студ. образоват. учреждений сред. проф. образования] / И.Г. Семакин, А.П. Шестаков .— 3-е изд., стер. — М. : Академия, 2012 .— 391 с (12 шт)

б) дополнительная литература:

№ п/п	Источник
5	Вирт, Никлаус. Алгоритмы и структуры данных с примерами на Паскале / Никлаус Вирт; пер. с англ. Д.Б. Подшивалова .— 2-е изд., испр. — СПб. : Невский диалект, 2005 .— 351 с. : ил. — (Библиотека программиста) .— Библиогр. в конце глав .— Указ. программ : с.345 (13 экз.)
6	Немнюгин С. А. Turbo Pascal Учебное пособие для студ. вузов. / С.А. Немнюгин. — СПб. : Питер, 2001. — 491 с. (45 экз.)
7	Фаронов В.В. Турбо Паскаль 7.0, Практика программирования / В.В. Фаронов. — М.: Нолидж.— 415 с. (15 экз.)
8	Вирт, Никлаус. Систематическое программирование. : Введение / Н. Вирт; пер. с англ. В.С. Штаркмана; под ред. Ю.М. Баяковского .— М. : Мир, 1977 .— 183 с.
9	Фаронов В. В. Turbo Pascal : учеб. пособие для студ. вузов, обуч. по направлению подготовки "Информатика и вычислительная техника" / В.В. Фаронов .— СПб. [и др.] : Питер, 2007 .— 366 с. : ил .— (Учебное пособие) .— Алф. указ. : с.360-366.
10	Вирт, Никлаус. Алгоритмы и структуры данных / Н. Вирт ; пер. с англ. Д. Б. Подшивалова .— М. : Мир, 1989 .— 360 с. : ил.
11	Абрамов В. Г. Введение в язык паскаль: Учеб. пособие для студ. вузов, обуч. по специальности "Прикладная математика"/ В.Г. Абрамов, Н.П. Трифонов, Г.Н. Трифонова.--М.: Наука Гл. ред. физ.-мат. лит, 1988.—318 с.

в) информационные электронно-образовательные ресурсы (официальные ресурсы интернет)*:

№ п/п	www.lib.vsu.ru – ЗНБ ВГУ Источник
12	Ищенко, В.А. Компьютерный практикум по информатике : учебно-методическое пособие для вузов / В.А. Ищенко, Е.К. Нагина ; Воронеж. гос. ун-т .— Воронеж : ЛОП ВГУ, 2006 .— 79 с. : ил, табл. — <URL: http://www.lib.vsu.ru/elib/texts/method/vsu/nov06114.pdf >
13	Мансуров К.Т. Основы программирования в среде Lazarus, 2010. — 772 с.: ил. (http://www.freepascal.ru/download/pdf/osnovy_programmirovaniya_v_srede_lazarus.pdf)
14	Основы программирования в системе Pascal ABC [Электронный ресурс] : учебно-методическое пособие для вузов / Воронеж. гос. ун-т ; сост. : А.В. Копытин, Д.И. Соломатин, А.И. Другалев .— Электрон. текстовые дан. — Воронеж : ИПЦ ВГУ, 2009 .— Свободный доступ из интрасети ВГУ .— Текстовый файл.— Windows 2000 ; Adobe Acrobat Reader .— <URL: http://www.lib.vsu.ru/elib/texts/method/vsu/m09-125.pdf >.
15	https://edu.vsu.ru

* Вначале указываются ЭБС, с которыми имеются договора у ВГУ, затем открытые электронно-образовательные ресурсы

16. Перечень учебно-методического обеспечения для самостоятельной работы (учебно-методические рекомендации, пособия, задачки, методические указания по выполнению практических (контрольных), курсовых работ и др.)

№ п/п	Источник
1	Андреева Т.А. Программирование на языке Pascal [Электронный ресурс] Т.А. Андреева . — Электр. текстовые данные . — М. : Интернет - Университет Информационных Технологий ИН(ТУИТ), 2016. — 277 с. — 5-9556-0025-6. — Режим доступа : http://www.iprbookshop.ru/52215.html
2	Обучение программированию Язык Pascal [Электронный ресурс] / НА. Таранухаи [др.]. — Электро-текстовые данные . — М. : СОЛОНП-

	РЕСС 5-91359-050-3. — Режим доступа : http://www.iprbookshop.ru/8633.html
3	Комлев Н.Ю. Самоучитель игры на Паскале . ABC инемного Турбо Электронный ресурс] / НЮ. Комлев — Электрон текстовые данные . — М. : СОЛОНП- РЕСС ; 2013. — 256 с. — 978-5-91359-112-8. — Режим доступа :

	http://www.iprbookshop.ru/20905.html
4	Практикум на ЭВМ Часть 1 [Электронный ресурс] : учебное пособие / . — Электрон текстовые данные . — М. : Евразийский открытый институт , 2012. — 263 с. — 978-5-374-00600-1. — Режим доступа : http://www.iprbookshop.ru/14644.html

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

При проведении занятий по дисциплине используются следующие образовательные технологии:

- активные и интерактивные формы проведения занятий;
- компьютерные технологии при проведении занятий;
- презентационные материалы и технологии при объяснении материала на лекционных занятиях;
- специализированное оборудование при проведении лабораторных работ;
- разбор конкретных ситуаций при постановке целей и задач к разработке прикладных программ, при выборе программного обеспечения по установленным критериям, при разработке программ по предусмотренным алгоритмам и методам

Для самостоятельной работы используется ЭБС Университетская библиотека online - www.lib.vsu.ru - ЗНБ ВГУ. Программное обеспечение, применяемое при реализации дисциплины – Microsoft Windows, LibreOffice, CodeBlocks, Adobe Reader, Mozilla FireFox.

Дистанционные образовательные технологии (ДОТ) применяются с использованием образовательного портала «Электронный университет ВГУ».

18. Материально-техническое обеспечение дисциплины:

Учебная аудитория для проведения занятий лекционного типа: ноутбук, мультимедиа-проектор, экран	г. Воронеж, Университетская пл., 1, ауд. 510П
Компьютерный класс с доступом к сети «Интернет»: компьютеры (мониторы, системные блоки) (15 шт.)	г. Воронеж, Университетская пл., 1, ауд. 313А

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
-------	--	----------------	-------------------------------------	--------------------

№ п/п	Наименование раздела дисциплины (модуля)	Компетенция(и)	Индикатор(ы) достижения компетенции	Оценочные средства
1.	Введение Языки программирования. Программы. Концепция данных. Классификация типов данных. Простые стандартные типы данных Структура программы. Ввод и вывод данных. Операторы языка	ОПК-3	ОПК-3.1 ОПК-3.2	Вопросы КИМ
	Сложные типы данных:		ОПК-3.3 ОПК-3.4	Вопросы КИМ
	массивы Процедуры и функции Строковые типы данных Нестандартные типы данных Сложные типы данных: множества Сложные типы данных: записи Работа с внешними данными (файлы) Культура разработки			
Промежуточная аттестация форма контроля - зачет				<i>Перечень вопросов Практическое задание</i>

* В графе «ФОС» в обязательном порядке перечисляются оценочные средства текущей и промежуточной аттестаций.

20 Типовые оценочные средства и методические материалы, определяющие процедуры оценивания

20.1 Текущий контроль успеваемости

Контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств:

Пример:

Для оценивания результатов обучения на зачете используются следующие показатели):

- 1) знание учебного материала и владение понятийным аппаратом дисциплины
- 2) умение связывать теорию с практикой (решение задач);
- 3) умение иллюстрировать ответ примерами.
- 4) умение решать задачи по курсу дисциплины

Для оценивания результатов обучения на экзамене (зачете с оценкой) используется 4- балльная шкала: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно». Для оценивания результатов обучения на зачете используется – зачтено, не зачтено
Соотношение показателей, критериев и шкалы оценивания результатов обучения.

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок
Отличное знание теоретического материала, правильное и эффективное решение задачи, правильные ответы на тестовые вопросы. Должны быть выполнены ВСЕ лабораторныеработы по дисциплине.	зачтено	Отлично
Задача не решена или серьезные пробелы в знании теоретического материала (с незнанием могут быть связаны и грубые ошибки в ответе на тестовые вопросы)	–	Не зачтено

20.2 Промежуточная аттестация

Перечень вопросов зачету:

УТВЕРЖДАЮ
заведующий кафедрой ядерной физики
Кадменский С.Г.
. .20

Направление подготовки 03.03.02 Физика
Дисциплина: Б.1 В 16.01
Программирование Вид контроля: зачет

Контрольно-измерительный материал №1

Группа ФИО

Тест по программированию на языке Паскаль

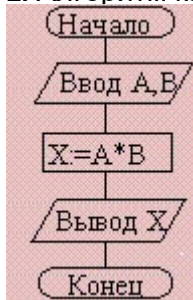
1. Логическое выражение

$(N \text{ Mod } 10 \text{ Mod } 2 = 0) \text{ Or } (N \text{ Div } 10 \text{ Mod } 10 \text{ Mod } 2 = 0) \text{ Or } (N \text{ Div } 100 \text{ Mod } 2 = 0)$

должно принимать значение TRUE тогда и только тогда, когда истинно высказывание

- 1) в трёхзначном натуральном числе все цифры чётные;
- 2) в трёхзначном натуральном числе одна чётная цифра;
- 3) в трёхзначном натуральном числе две чётных цифры;
- 4) в трёхзначном натуральном числе хотя бы одна чётная цифра;
- 5) в трёхзначном натуральном числе нет чётных цифр.

2. Алгоритм какого типа изображен на блок-схеме?



- 1) циклический;
- 2) разветвляющийся;
- 3) вспомогательный;
- 4) линейный;
- 5) комбинация разветвляющегося и цикла.

3. В каком из условных операторов допущена синтаксическая ошибка?

- 1) if B = 0 then Writeln('Деление на нуль невозможно.')
- 2) if a > b then max := a else max := b;
- 3) if (a > b) and (b > 0) then c := a + b;
- 4) if a < b then min := a; else min := b;
- 5) if (a <> b) and (b > 0) then c := a - b;

4. Оператор выбора начинается словом ...

- 1) case;
- 2) if;
- 3) var;
- 4) else;
- 5) until.

5. Из следующих утверждений выберите истинное:

- 1) переменная, используемая в операторе For, может быть типа real;
- 2) в операторе While проверка условия осуществляется после тела цикла;
- 3) перед Else всегда ставится символ «;» ;
- 4) оператором Write производится вывод данных на экран.
- 5) оператором Readln производится вывод данных на экран.

6. Что будет выведено на экран в результате выполнения следующих инструкций?

```
for i:=1 to 5  
do write('*');
```

- 1) Будут выведены пять звездочек в одной строке.
- 2) Будут выведены цифры от 1 до 5 в одной строке.
- 3) Будут выведены пять звездочек столбиком.
- 4) Будут выведены цифры от 1 до 5 столбиком.
- 5) Будут выведены звездочки и апострофы.

7. Чему будет равно значение переменной n после выполнения следующих инструкций?

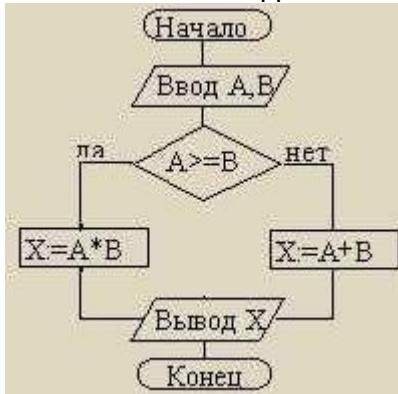
```
n:=0;
while n<=5 do
  n:=n+1;
```

- 1) 1; 2) 3; 3) 0; 4) 6; 5) 5.

8. Для обозначения числа 3.14 нужно использовать переменную типа ...

- 1) real; 2) integer; 3) string; 4) char; 5) longint.

9. После исполнения фрагмента программы, изображенного на блок-схеме



при A = 5, B = 4 значение X будет равно 1) 20; 2) 9; 3) 5; 4) 4; 5) 1.

10. Цикл с предусловием выполняется так:

- 1) выполняется тело цикла, изменяется параметр цикла, проверяется условие продолжения выполнения цикла;
- 2) изменяется параметр цикла, проверяется условие продолжения выполнения цикла, выполняется тело цикла;
- 3) проверяется условие продолжения выполнения цикла, выполняется тело цикла;
- 4) тело цикла выполняется N раз (N — натуральное);
- 5) определяется, сколько раз должен быть выполнен цикл, и далее цикл с предусловием сводится к циклу с параметром.

11. Цикл в фрагменте программы

```
P := 4; Repeat P := P * 0.1 Until P < 0.0001;
```

будет исполнен

- 1) 0 раз; 2) 1 раз; 3) 4 раза; 4) 5 раз; 5) бесконечное число раз.

12. Из перечисленных ниже в программе обязателен

- 1) раздел Var; 2) раздел Const; 3) раздел Type; 4) раздел Label; 5) раздел Begin ... End.

13. Значения переменных p и d после выполнения фрагмента алгоритма

```
k := 47; Case k Mod 9 Of
```

```
5: Begin d := k; p := True End;
```

```
0..2: Begin d := 2; p := False
```

```
End; 8: Begin d := 1; p :=
```

```
False End Else Begin d := 1;
```

```
p := True End
```

```
End;
```

равны

- 1) p = True, d = 1; 2) p = False, d = 2; 3) p = False, d = 3; 4) p = True, d = 47; 5) p = True, d = 2.

14. Какая из приведенных серий операторов определяет и печатает индекс последнего отрицательного элемента в линейном массиве из n элементов?

а) i := n; While (i >= 1) And (m[i] > 0) Do i:=i-1;

If i < 1 Then WriteLn ('i = 0') Else WriteLn ('i = ', i);

б) k := 0; For i := 1 To n Do If m[i] < 0 Then k := i; WriteLn ('i = ',

k); в) i := n; Repeat i := i - 1 Until (m[i] < 0); WriteLn ('i = ', i);

- 1) а, б; 2) б, в; 3) а, б, в; 4) б; 5) ни один из ответов 1–4 не верен.

15. Фрагмент

```
программы S := A; A :=
```

```
B; B := S; выполняет
```

- 1) обмен значений переменных A, B;
- 2) присваивание переменным A, B значения S;
- 3) замена значения переменной A значением переменной B;
- 4) во фрагменте не выполняется никаких действий;
- 5) замена значения переменной B значением переменной A.

УТВЕРЖДАЮ

Направление подготовки 03.03.02 Физика
Дисциплина: Б.1 В 16.01
Программирование Вид контроля: зачет

Контрольно-измерительный материал №2
Группа ФИО

Тест по программированию на языке Паскаль

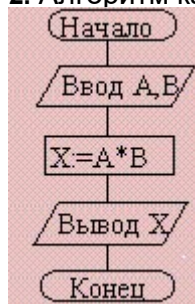
1. Логическое выражение

$(N \bmod 10 \bmod 2 = 0) \text{ Or } (N \text{ Div } 10 \bmod 10 \bmod 2 = 0) \text{ Or } (N \text{ Div } 100 \bmod 2 = 0)$

должно принимать значение TRUE тогда и только тогда, когда истинно высказывание

- 1) в трёхзначном натуральном числе хотя бы одна чётная цифра;
- 2) в трёхзначном натуральном числе одна чётная цифра;
- 3) в трёхзначном натуральном числе две чётных цифры;
- 4) в трёхзначном натуральном числе все цифры чётные;
- 5) в трёхзначном натуральном числе нет чётных цифр.

2. Алгоритм какого типа изображен на блок-схеме?



- 1) циклический;
- 2) линейный;
- 3) вспомогательный;
- 4) разветвляющийся;
- 5) комбинация разветвляющегося и цикла.

3. В каком из условных операторов допущена синтаксическая ошибка?

- 1) if B = 0 then Writeln('Деление на нуль невозможно.')
- 2) if a > b then max := a else max := b;
- 3) if (a>b) and (b>0) then c:=a+b;
- 4) if (a<>b) and (b>0) then c:=a-b;
- 5) if a < b then min := a; else min := b;

4. Оператор выбора начинается словом ...

- 1) case;
- 2) if;
- 3) var;
- 4) else;
- 5) until.

5. Из следующих утверждений выберите истинное:

- 1) переменная, используемая в операторе For, может быть типа real;
- 2) в операторе While проверка условия осуществляется после тела цикла;
- 3) перед Else всегда ставится символ «;» ;
- 4) оператором Write производится вывод данных на экран.
- 5) оператором Readln производится вывод данных на экран.

6. Что будет выведено на экран в результате выполнения следующих инструкций?

```
for i:=1 to 5  
do write('*');
```

- 6) Будут выведены цифры от 1 до 5 столбиком.
- 7) Будут выведены цифры от 1 до 5 в одной строке.
- 8) Будут выведены пять звездочек столбиком.
- 9) Будут выведены пять звездочек в одной строке.
- 10) Будут выведены звездочки и апострофы.

7. Чему будет равно значение переменной n после выполнения следующих инструкций?

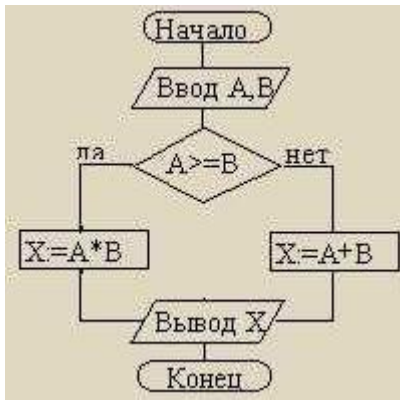
```
n:=0;  
while n<=5 do  
n:=n+1;
```

- 1) 1;
- 2) 5;
- 3) 0;
- 4) 6;
- 5) 3.

8. Для обозначения числа 3.14 нужно использовать переменную типа ...

- 1) byte;
- 2) integer;
- 3) string;
- 4) real;
- 5) boolean.

9. После исполнения фрагмента программы, изображенного на блок-схеме



при $A = 5$, $B = 4$ значение X будет равно 1) 1; 2) 9; 3) 5; 4) 4; 5) 20.

10. Цикл с предусловием выполняется так:

1) проверяется условие продолжения выполнения цикла,

выполняется тело цикла;

2) изменяется параметр цикла, проверяется условие продолжения выполнения цикла, выполняется тело цикла;

3) выполняется тело цикла, изменяется параметр цикла, проверяется условие продолжения выполнения цикла;

4) тело цикла выполняется N раз (N — натуральное);

5) определяется, сколько раз должен быть выполнен цикл, и далее цикл с предусловием сводится к циклу с параметром.

11. Цикл в фрагменте программы

$P := 4$; Repeat $P := P * 0.1$ Until $P < 0.0001$;

будет исполнен

1) 0 раз; 2) 1 раз; 3) 4 раза; 4) 5 раз; 5) бесконечное число раз.

12. Из перечисленных ниже в программе обязателен

1) раздел Var; 2) раздел Const; 3) раздел Type; 4) раздел Begin ... End; 5) раздел Label

13. Значения переменных p и d после выполнения фрагмента алгоритма

$k := 47$; Case $k \text{ Mod } 9$ Of

5: Begin $d := k$; $p := \text{True}$ End;

0..2: Begin $d := 2$; $p := \text{False}$

End; 8: Begin $d := 1$; $p :=$

False End Else Begin $d := 1$;

$p := \text{True}$ End

End;

равн

ы 1) $p = \text{True}$, $d = 1$; 2) $p = \text{True}$, $d = 47$; 3) $p = \text{False}$, $d = 3$; 4) $p = \text{False}$, $d = 2$; 5) $p = \text{True}$, $d = 2$.

14. Какая из приведенных серий операторов определяет и печатает индекс последнего отрицательного элемента в линейном массиве из n элементов?

а) $i := n$; While $(i >= 1)$ And $(m[i] > 0)$ Do $i := i - 1$;

If $i < 1$ Then WriteLn ('i = 0') Else WriteLn ('i = ', i);

б) $k := 0$; For $i := 1$ To n Do If $m[i] < 0$ Then $k := i$; WriteLn ('i = ',

k); в) $i := n$; Repeat $i := i - 1$ Until $(m[i] < 0)$; WriteLn ('i = ', i);

1) а, б; 2) б, в; 3) а, б, в; 4) б; 5) ни один из ответов 1–4 не верен.

15. Фрагмент

программы $S := A$; $A :=$

B ; $B := S$; выполняет

1) замена значения переменной A значением переменной B ;

2) присваивание переменным A , B значения S ;

3) обмен значений переменных A , B ;

4) во фрагменте не выполняется никаких действий;

5) замена значения переменной B значением переменной A .

Преподаватель

УТВЕРЖДАЮ
заведующий кафедрой ядерной физики
Кадменский С.Г.

Направление подготовки 03.03.02 Физика
Дисциплина: Б.1 В 16.01
Программирование Вид контроля: зачет

Тест по программированию на языке Паскаль

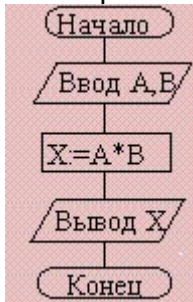
1. Логическое выражение

$(N \text{ Mod } 10 \text{ Mod } 2 = 0) \text{ Or } (N \text{ Div } 10 \text{ Mod } 10 \text{ Mod } 2 = 0) \text{ Or } (N \text{ Div } 100 \text{ Mod } 2 = 0)$

должно принимать значение TRUE тогда и только тогда, когда истинно высказывание

- 1) в трёхзначном натуральном числе хотя бы одна чётная цифра;
- 2) в трёхзначном натуральном числе нет чётных цифр;
- 3) в трёхзначном натуральном числе две чётных цифры;
- 4) в трёхзначном натуральном числе все цифры чётные;
- 5) в трёхзначном натуральном числе одна чётная цифра;

2. Алгоритм какого типа изображен на блок-схеме?



- 1) циклический;
- 2) комбинация разветвляющегося и цикла;
- 3) вспомогательный;
- 4) разветвляющийся;
- 5) линейный

3. В каком из условных операторов допущена синтаксическая ошибка?

- 1) if (B = 0) then Writeln('Деление на нуль невозможно.')
- 2) if a > b then max := a; else max := b;
- 3) if (a>b) and (b>0) then c:=a+b;
- 4) if (a<>b) and (b>0) then c:=a-b;
- 5) if a < b then min := a else min := b;

4. Оператор выбора начинается словом ...

- 1) repeat;
- 2) case;
- 3) while;
- 4) else;
- 5) until.

5. Из следующих утверждений выберите истинное:

- 1) переменная, используемая в операторе For, может быть типа real;
- 2) в операторе While проверка условия осуществляется после тела цикла;
- 3) перед Else всегда ставится символ «;» ;
- 4) оператором Write производится вывод данных на экран.
- 5) оператором Readln производится вывод данных на экран.

6. Что будет выведено на экран в результате выполнения следующих инструкций?

```

for i:=1 to 5
do write('*');
    
```

- 11) Будут выведены цифры от 1 до 5 столбиком.
- 12) Будут выведены цифры от 1 до 5 в одной строке.
- 13) Будут выведены пять звездочек столбиком.
- 14) Будут выведены пять звездочек в одной строке.
- 15) Будут выведены звездочки и апострофы.

7. Чему будет равно значение переменной n после выполнения следующих инструкций?

```

n:=0;
while n<=5 do
n:=n+1;
    
```

- 1) 0;
- 2) 5;
- 3) 1;
- 4) 6;
- 5) 3.

8. Для обозначения числа 9.8 нужно использовать переменную типа ...

- 1) byte;
- 2) integer;
- 3) string;
- 4) real;
- 5) boolean.

9. После исполнения фрагмента программы, изображенного на блок-схеме



при A = 5, B = 4 значение X будет равно 1) 1; 2) 9; 3) 20; 4) 4; 5) 5.

10. Цикл с параметром (for) выполняется так:
- 1) тело цикла выполняется N раз (N — натуральное);
 - 2) изменяется параметр цикла, проверяется условие продолжения выполнения цикла, выполняется тело цикла;
 - 3) выполняется тело цикла, изменяется параметр цикла, проверяется условие продолжения выполнения цикла;
 - 4) проверяется условие продолжения выполнения цикла, выполняется тело цикла;
 - 5) определяется, сколько раз должен быть выполнен цикл, и далее цикл с предусловием сводится к циклу с параметром.

11. Цикл в фрагменте программы

```
P := 4; Repeat P := P * 0.1 Until P < 0.0001;
```

будет исполнен

- 1) 0 раз; 2) 1 раз; 3) 4 раза; 4) 5 раз; 5) бесконечное число раз.

12. Из перечисленных ниже в программе обязателен

- 1) раздел Var; 2) раздел Const; 3) раздел Type; 4) раздел Begin ... End;
- 5) раздел Label

13. Значения переменных p и d после выполнения фрагмента алгоритма

```
k := 47; Case k Mod 9 Of
```

```
5: Begin d := k; p := True End;
```

```
0..2: Begin d := 2; p := False
```

```
End; 8: Begin d := 1; p :=
```

```
False End Else Begin d := 1;
```

```
p := True End
```

```
End;
```

равны

- 1) p = True, d = 1; 2) p = True, d = 47; 3) p = False, d = 3; 4) p = False, d = 2; 5) p = True, d = 2.

14. Какая из приведенных серий операторов определяет и печатает индекс последнего отрицательного элемента в линейном массиве из n элементов?

```
а) i := n; While (i >= 1) And (m[i] > 0) Do i:=i-1;
```

```
If i < 1 Then WriteLn ('i = 0') Else WriteLn ('i = ', i);
```

```
б) k := 0; For i := 1 To n Do If m[i] < 0 Then k := i; WriteLn ('i = ',
```

```
k); в) i := n; Repeat i := i - 1 Until (m[i] < 0); WriteLn ('i = ', i);
```

- 1) а, б; 2) б, в; 3) а, б, в; 4) б; 5) ни один из ответов 1–4 не верен.

15. Фрагмент

```
программы S := A; A :=
```

```
B; B := S; выполняет
```

- 1) замена значения переменной A значением переменной B;
- 2) присваивание переменным A, B значения S;
- 3) замена значения переменной B значением переменной A;
- 4) во фрагменте не выполняется никаких действий;
- 5) обмен значений переменных A, B.

Преподаватель

Перечень практических заданий

1. Даны координаты двух противоположных вершин прямоугольника: (x_1, y_1) , (x_2, y_2) . Стороны прямоугольника параллельны осям координат. Найти периметр и площадь данного прямоугольника.
2. Дано целое число N и набор из N целых чисел. Найти номера первого и последнего минимального элемента из данного набора и вывести их в указанном порядке.
3. Даны целые положительные числа A и B ($A < B$). Вывести все целые числа от A до B включительно; при этом каждое число должно выводиться столько раз, каково его значение (например, число 3 выводится 3 раза).
4. Дано целое число N и набор из N целых чисел. Найти количество элементов, содержащихся между первым и последним максимальным элементом. Если в наборе имеется единственный максимальный элемент, то вывести 0.
5. Дано значение температуры T в градусах Фаренгейта. Определить значение этой же температуры в градусах Цельсия. Температура по Цельсию T_C и температура по Фаренгейту T_F связаны следующим соотношением: $T_C = (T_F - 32) \cdot 5/9$.
6. Дан массив размера N и целые числа K и L ($1 \leq K \leq L \leq N$). Найти среднее арифметическое элементов массива с номерами от K до L включительно.
7. Найти корни квадратного уравнения $A \cdot x^2 + B \cdot x + C = 0$, заданного своими коэффициентами A , B , C (коэффициент A не равен 0), если известно, что дискриминант уравнения положителен. Вывести вначале меньший, а затем больший из найденных корней.
8. Дан массив ненулевых целых чисел размера N . Проверить, чередуются ли в нем положительные и отрицательные числа. Если чередуются, то вывести 0, если нет, то вывести порядковый номер первого элемента, нарушающего закономерность.
9. Дано трехзначное число. Вывести число, полученное при перестановке цифр десятков и единиц исходного числа (например, 123 перейдет в 132).
10. Дан целочисленный массив A размера N . Переписать в новый целочисленный массив B того же размера вначале все элементы исходного массива с четными номерами, а затем — с нечетными: $A_2, A_4, A_6, \dots, A_1, A_3, A_5, \dots$.
11. Дано целое число N (> 0). Используя операции деления нацело и взятия остатка от деления, найти количество и сумму его цифр.
12. Дана матрица размера $M \times N$. Найти номер ее строки с наибольшей суммой элементов и вывести данный номер, а также значение наибольшей суммы.
13. Дни недели пронумерованы следующим образом: 0 — воскресенье, 1 — понедельник, 2 — вторник, ..., 6 — суббота. Дано целое число K , лежащее в диапазоне 1–365. Определить номер дня недели для K -го дня года, если известно, что в этом году 1 января было четвергом.
14. Описать функцию $Even(K)$ логического типа, возвращающую $True$, если целый параметр K является четным, и $False$ в противном случае. С ее помощью найти количество четных чисел в наборе из 10 целых чисел.
15. Даны целые числа a , b , c . Проверить истинность высказывания: «Существует треугольник со сторонами a , b , c ».
16. Даны целые положительные числа M , N , число D и набор из M чисел. Сформировать матрицу размера $M \times N$, у которой первый столбец совпадает с исходным набором чисел, а элементы каждого следующего столбца равны сумме соответствующего элемента предыдущего столбца и числа D (в результате каждая строка матрицы будет содержать элементы арифметической прогрессии).
17. Дано целое число N (> 0). Используя один цикл, найти сумму $1/(1!) + 1/(2!) + 1/(3!) + \dots + 1/(N!)$ (выражение $N!$ — N -факториал — обозначает произведение всех целых чисел от 1 до N : $N! = 1 \cdot 2 \cdot \dots \cdot N$). Полученное число является приближенным значением константы $e = \exp(1)$.
18. Дан массив A размера N (≤ 6). Упорядочить его по возрастанию методом сортировки простым выбором: найти максимальный элемент массива и поменять его местами с последним ($N-1$) элементом; выполнить описанные действия $N - 1$ раз, каждый раз уменьшая на 1 количество анализируемых элементов и выводя содержимое массива.

19. Даны две переменные целого типа: A и B. Если их значения не равны, то присвоить каждой переменной сумму этих значений, а если равны, то присвоить переменным нулевые значения. Вывести новые значения переменных A и B.
20. Описать функцию RingS(R1, R2) вещественного типа, находящую площадь кольца, заключенного между двумя окружностями с общим центром и радиусами R1 и R2 (R1 и R2 — вещественные, $R1 > R2$). С ее помощью найти площади трех колец, для которых даны внешние и внутренние радиусы.

21. Даны три переменные вещественного типа: A, B, C. Если их значения упорядочены по возрастанию, то удвоить их; в противном случае заменить значение каждой переменной на противоположное. Вывести новые значения переменных A, B, C.
22. Дан целочисленный массив размера N. Удалить из массива все одинаковые элементы, оставив их последние вхождения.
23. Дана строка-предложение на русском языке. Подсчитать количество содержащихся в строке знаков препинания.
24. Дан массив размера N. Поменять местами его минимальный и максимальный элементы.
25. Даны три переменные вещественного типа: A, B, C. Если их значения упорядочены по возрастанию, то удвоить их; в противном случае заменить значение каждой переменной на противоположное. Вывести новые значения переменных A, B, C.
26. Описать процедуру SortDec3(A, B, C), меняющую содержимое переменных A, B, C таким образом, чтобы их значения оказались упорядоченными по убыванию (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой процедуры упорядочить по убыванию два данных набора из трех чисел: (A₁, B₁, C₁) и (A₂, B₂, C₂).
27. Дано целое число N (> 0). Найти сумму
 28. $1 + 1/2 + 1/3 + \dots + 1/N$
29. Даны два массива A и B размера 5, элементы которых упорядочены по возрастанию. Объединить эти массивы так, чтобы результирующий массив C (размера 10) остался упорядоченным по возрастанию.
30. Единицы длины пронумерованы следующим образом: 1 — дециметр, 2 — километр, 3 — метр, 4 — миллиметр, 5 — сантиметр. Дан номер единицы длины (целое число в диапазоне 1–5) и длина отрезка в этих единицах (вещественное число). Найти длину отрезка в метрах.
31. Дан файл целых чисел. Создать два новых файла, первый из которых содержит четные числа из исходного файла, а второй — нечетные (в том же порядке). Если четные или нечетные числа в исходном файле отсутствуют, то соответствующий результирующий файл оставить пустым.
32. Дано целое число N (> 0). Найти двойной факториал N: $N!! = N \cdot (N-2) \cdot (N-4) \cdot \dots$
33. (последний сомножитель равен 2, если N — четное, и 1, если N — нечетное). Чтобы избежать целочисленного переполнения, вычислять это произведение с помощью вещественной переменной и вывести его как вещественное число.
34. Дан файл вещественных чисел. Заменить в файле каждый элемент, кроме начального и конечного, на его среднее арифметическое с предыдущим и последующим элементом.
35. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Найти длину самого длинного слова.
36. Описать процедуру TrianglePS(a, P, S), вычисляющую по стороне a равностороннего треугольника его периметр $P = 3 \cdot a$ и площадь $S = a^2 \cdot \sqrt{3} / 4$ (a — входной, P и S — выходные параметры; все параметры являются вещественными). С помощью этой процедуры найти периметры и площади трех равносторонних треугольников с данными сторонами.
37. Дана строка, содержащая по крайней мере один символ пробела. Вывести подстроку, расположенную между первым и последним пробелом исходной строки. Если строка содержит только один пробел, то вывести пустую строку.
38. Дан файл целых чисел. Создать новый файл, содержащий те же элементы, что и исходный файл, но в обратном порядке.
39. Даны строки S и S₀. Удалить из строки S первую подстроку, совпадающую с S₀. Если совпадающих подстрок нет, то вывести строку S без изменений.
40. Дано целое число K и файл, содержащий неотрицательные целые числа. Вывести K-й элемент файла (элементы нумеруются от 1). Если такой элемент отсутствует,

то вывести -1 .

41. Дана квадратная матрица порядка M . Обнулить элементы матрицы, лежащие одновременно выше главной диагонали и выше побочной диагонали. Условный оператор не использовать.
42. Дана строка, изображающая двоичную запись целого положительного числа. Вывести строку, изображающую десятичную запись этого же числа.

19.3.4 Тестовые задания

19.3.4 Перечень заданий для контрольных работ

19.3.5 Темы курсовых работ

19.3.6 Темы рефератов

19.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Оценка знаний, умений и навыков, характеризующая этапы формирования компетенций в рамках изучения дисциплины осуществляется в ходе текущей и промежуточной аттестаций.

Текущая аттестация проводится в соответствии с Положением о текущей аттестации обучающихся по программам высшего образования Воронежского государственного университета. Текущая аттестация проводится в форме(ах) (указать нужное): устного опроса (индивидуальный опрос, фронтальная беседа, доклады); письменных работ (контрольные, эссе, сочинения, выполнение практико-ориентированных заданий, лабораторные работы и пр.); тестирования; оценки результатов практической деятельности (курсовая работа, портфолио и др.). Критерии оценивания приведены выше.

Промежуточная аттестация проводится в соответствии с Положением о промежуточной аттестации обучающихся по программам высшего образования.

Контрольно-измерительные материалы промежуточной аттестации включают в себя теоретические вопросы, позволяющие оценить уровень полученных знаний и/или практическое(ие) задание(я), позволяющее(ие) оценить степень сформированности умений и(или) навыков, и(или) опыт деятельности (указывает реальную структуру). При оценивании используются количественные или качественные шкалы оценок (нужное выбрать). Критерии оценивания приведены выше.