

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Луганский государственный университет имени Владимира Даля»

Экономический факультет  
Кафедра экономической кибернетики и прикладной статистики

УТВЕРЖДАЮ:  
Декан экономического факультета  
Тхор Е.С.  
« 24 » \_\_\_\_\_ 2023 года



**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

**«БИЗНЕС-ИНФОРМАТИКА 2»**

По направлению подготовки 38.03.05 Бизнес-информатика  
Профили: «Информационная бизнес-аналитика», «Экономическая аналитика  
и бизнес-статистика»

Луганск – 2023

Лист согласования РПУД


Рабочая программа учебной дисциплины «Бизнес-информатика 2» по направлению подготовки 38.03.05 Бизнес-информатика. – 35 с.

Рабочая программа учебной дисциплины «Бизнес-информатика 2» составлена с учетом Федерального государственного образовательного стандарта высшего образования по направлению подготовки 38.03.05 Бизнес-информатика, утвержденного приказом Министерства науки и высшего образования Российской Федерации от 29 июня 2020 года № 838.

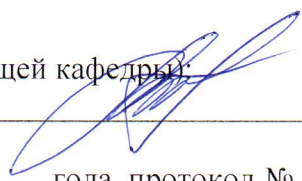
СОСТАВИТЕЛЬ (СОСТАВИТЕЛИ):

к.т.н., доц. Велигура А.В.

Рабочая программа дисциплины утверждена на заседании кафедры экономической кибернетики и прикладной статистики «18» 04 2023 г., протокол № 26

Заведующий кафедрой экономической кибернетики  
и прикладной статистики  А.В. Велигура

Переутверждена: « \_\_\_ » \_\_\_\_\_ 20\_\_ г., протокол № \_\_\_\_\_

Согласована (для обеспечивающей кафедры)   
Декан экономического факультета \_\_\_\_\_ Тхор Е.С.

Переутверждена: « \_\_\_ » \_\_\_\_\_ 20\_\_ года, протокол № \_\_\_\_\_

Рекомендована на заседании учебно-методической комиссии экономического факультета «21» апреля 2023 г., протокол № 4.

Председатель учебно-методической  
комиссии экономического факультета  Е.Н. Шаповалова

© Велигура А.В., 2023 год

© ФГБОУ ВО «ЛГУ им. В. Даля», 2023 год

## Структура и содержание дисциплины

### 1. Цели и задачи дисциплины, ее место в учебном процессе

Цель изучения дисциплины – изучение и освоение базовых понятий и приемов программирования, применяемых на всех основных этапах разработки программ; изучение методов программирования для подготовки к овладению знаниями в области технологии программирования; подготовка к осознанному использованию как языков программирования, так и методов программирования. Воспитательной целью дисциплины является формирование у студентов научного, творческого подхода к освоению технологий, методов и средств производства программного обеспечения.

Задачи:

формирование целостного представления о предмете;

освоение теоретических знаний и практических навыков, позволяющих ориентироваться в области разработки алгоритмов решения задач и написания программных кодов на языках программирования высокого уровня;

освоение теоретических основ структурного и объектно-ориентированного программирования;

изучение конкретных языков программирования;

использование языков программирования для решения практических задач.

### 2. Место дисциплины в структуре ОПОП ВО

Дисциплина «Бизнес-информатика 2» относится к вариативной части (часть, формируемая участниками образовательных отношений) дисциплин.

Содержание дисциплины является логическим продолжением содержания дисциплин «Бизнес-информатика», «Дискретный анализ в экономике», «Логические основы систем управления в экономике», «Математика».

Содержание дисциплины служит основой для освоения дисциплин «Вычислительные системы, сети, телекоммуникации», «Системы, основанные на знаниях», «Информационные системы и технологии в управленческой деятельности», «Интеллектуальный анализ данных», «Технологии разработки программных продуктов», «Технология проектирования и администрирования баз данных», «Информационно-коммуникационные технологии в бизнесе» и учебной практики.

### 3. Требования к результатам освоения содержания дисциплины

Код и наименование компетенции	Индикаторы достижений компетенции (по реализуемой дисциплине)	Перечень планируемых результатов
ПК-3 Способен управлять ИТ-инфраструктурой предприятия	ПК-3.3 Способен реализовывать алгоритмы обработки данных на языках высокого уровня	<b>Знать:</b> Методы разработки и реализации алгоритмов обработки данных на языках высокого уровня
		<b>Уметь:</b> разрабатывать программы на языках высокого уровня
		<b>Владеть:</b> навыками написания и отладки программ, реализующих алгоритмы обработки данных на языках высокого уровня

### 4. Структура и содержание дисциплины

#### 4.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов (зач. ед.)		
	Очная форма	Очно-заочная форма	Заочная форма
<b>Общая учебная нагрузка (всего)</b>	<b>144</b> (4 зач. ед)	<b>144</b> (4 зач. ед)	<b>144</b> (4 зач. ед)
<b>Обязательная аудиторная учебная нагрузка (всего)</b>	<b>68</b>	<b>55</b>	<b>6</b>
<b>в том числе:</b>			
Лекции	17	14	2
Семинарские занятия			
Практические занятия	17	13	2
Лабораторные работы	34	28	2
Курсовая работа (курсовой проект)			
Другие формы и методы организации образовательного процесса ( <i>расчетно-графические работы, групповые дискуссии, ролевые игры, тренинг, компьютерные симуляции, интерактивные лекции, семинары, анализ деловых ситуаций и т.п.</i> )			
<b>Самостоятельная работа студента (всего)</b>	<b>76</b>	<b>89</b>	<b>138</b>
Форма аттестации	экзамен	экзамен	экзамен

#### 4.2. Содержание разделов дисциплины

##### **Тема 1. ОСНОВЫ АЛГОРИТМИЗАЦИИ**

Алгоритмы. Свойства алгоритмов. Способы описания алгоритмов. Основные конструкции алгоритмического языка: линейный алгоритм, ветвление, цикл. Этапы решения задач с помощью ЭВМ: постановка задачи, создание модели, алгоритм, кодирование алгоритма, анализ результатов. Правила постановки задачи. Модель: входные и выходные параметры, соотношение между ними.

## ***Тема 2. ЯЗЫКИ ПРОГРАММИРОВАНИЯ***

Развитие языков программирования. Обзор языков программирования. Области применения языков программирования. Стандарты языков программирования. Среда проектирования. Компиляторы и интерпретаторы.

## ***Тема 3. ТИПЫ ДАННЫХ***

Переменные и константы. Объявление объектов данных. Внутренне представление данных в памяти компьютера

## ***Тема 4. ОПЕРАТОРЫ ЯЗЫКА ПРОГРАММИРОВАНИЯ***

Основные понятия языка Python: алфавит, служебные слова, константы, переменные, имена. Встроенные функции языка. Понятие выражений. Типы выражений. Структура программы на Python. Операции и выражения. Правила формирования и вычисления выражений. Структура программы. Ввод и вывод данных. Виды операторов языка Python: простые, сложные (структурные), составные. Элементарный ввод и вывод на Python. Операторы присваивания, управления ходом выполнения программы. Идентичность и совместимость типов данных при присваивании. Условный оператор. Использование вложенных условных операторов при составлении программ. Логические функции в условных операторах. Операторы цикла. Цикл с постусловием. Цикл с предусловием. Цикл с параметром. Вложенные циклы.

## ***Тема 5. ПРОЦЕДУРЫ И ФУНКЦИИ***

Общие сведения о подпрограммах. Понятие подпрограмм, подпрограммы – процедуры и подпрограммы - функции. Определение и вызов подпрограмм. Описание процедур. Выполнение процедур. Описание функций, их выполнение. Виды параметров в подпрограммах. Область видимости и время жизни переменной. Механизм передачи параметров. Составление библиотек подпрограмм.

## ***Тема 6. СТРУКТУРИЗАЦИЯ В ПРОГРАММИРОВАНИИ***

Основы структурного программирования. Методы структурного программирования.

## ***Тема 7. МОДУЛЬНОЕ ПРОГРАММИРОВАНИЕ***

Модульное программирование. Понятие модуля. Структура модуля. Компиляция и компоновка программы. Библиотеки подпрограмм: понятие и виды. Схемы вызова библиотек. Статическое и динамическое связывание.

## ***Тема 8. СПИСКИ***

Списочные типы данных. Инициализация. Действия над списками. Заполнение списка данными. Ввод и вывод одномерных и двумерных списков. Стандартные функции для списков целых и вещественных чисел.

Использование стандартных функций для работы со списками. Обработка списка. Удаление и вставка элементов в список.

### ***Тема 9. СТРОКИ***

Символьный и строковый типы. Объявление типов. Поиск, удаление, замена и добавление символов в строке. Операции со строками

### ***Тема 10. ФАЙЛЫ***

Типы файлов. Организация доступа к файлам. Файлы последовательного доступа. Открытие и закрытие файла последовательного доступа. Запись в файл и чтение из файла последовательного доступа. Файлы произвольного доступа. Порядок работы с файлами произвольного доступа. Создание структуры записи. Открытие и закрытие файла произвольного доступа. Запись и считывание из файла произвольного доступа. Использование файла произвольного доступа. Стандартные процедуры и функции для файлов разного типа. Использование стандартных процедур и функций для работы с файлами.

### ***Тема 11. ОСНОВНЫЕ ПРИНЦИПЫ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ (ООП)***

История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс. Основные принципы ООП: инкапсуляция, наследование, полиморфизм. Событийно-управляемая модель программирования. Компонентно-ориентированный подход. Классы объектов. Компоненты и их свойства.

### ***Тема 12. ЭТАПЫ РАЗРАБОТКИ ПРОГРАММЫ***

Программирование приложения. Тестирование, отладка приложения. Создание документации. Понятие интерфейса программы, дружественный интерфейс. Организация интерфейса программы. Создание базовых элементов интерфейса.

### ***Тема 13. ИЕРАРХИЯ КЛАССОВ***

Классы объектно-ориентированного языка программирования: виды, назначение, свойства, методы, события. Объявление класса, свойств и методов экземпляра класса. Конструкторы и деструкторы. Имя, описание и определение конструктора. Имя, описание и определение деструктора. Создание класса с использованием конструкторов и деструкторов. Создание класса с конструктором и деструктором

### ***Тема 14. РАЗРАБОТКА ОКОННОГО ПРИЛОЖЕНИЯ***

Разработка функционального интерфейса приложения. Создание интерфейса приложения. Разработка функциональной схемы работы приложения. Создание процедур обработки событий.

### 4.3. Лекции

№ п/п	Название темы	Объем часов		
		Очная форма	Очно-заочная форма	Заочная форма
1	Основы алгоритмизации	1	1	1
2	Языки программирования	1	1	
3	Типы данных	1	1	
4	Операторы языка программирования	1	1	
5	Процедуры и функции	1	1	
6	Структуризация в программировании	1	1	1
7	Модульное программирование	1	1	
8	Списки	2	1	
9	Строки	2	1	
10	Файлы	1	1	
11	Основные принципы объектно-ориентированного программирования (ООП)	1	1	
12	Этапы разработки программы	1	1	
13	Иерархия классов	2	1	
14	Разработка оконного приложения	1	1	
<b>Итого:</b>		<b>17</b>	<b>14</b>	<b>2</b>

### 4.4. Практические (семинарские) занятия

№ п/п	Название темы	Объем часов		
		Очная форма	Очно-заочная форма	Заочная форма
1	Основы алгоритмизации	1	1	1
2	Языки программирования	1	1	
3	Типы данных	1	1	
4	Операторы языка программирования	1	1	
5	Процедуры и функции	1	1	
6	Структуризация в программировании	1	1	1
7	Модульное программирование	1	1	
8	Списки	2	1	
9	Строки	2	1	
10	Файлы	1	1	
11	Основные принципы объектно-ориентированного программирования (ООП)	1	1	
12	Этапы разработки программы	1	1	
13	Иерархия классов	2	0,5	
14	Разработка оконного приложения	1	0,5	
<b>Итого:</b>		<b>17</b>	<b>13</b>	<b>2</b>

### 4.5. Лабораторные работы

№ п/п	Название темы	Объем часов		
		Очная форма	Очно-заочная форма	Заочная форма
1	Основы алгоритмизации	2	2	1
2	Языки программирования	2	2	
3	Типы данных	2	2	



4	Операторы языка программирования	2	2	
5	Процедуры и функции	2	2	
6	Структуризация в программировании	2	2	
7	Модульное программирование	2	2	
8	Списки	4	2	
9	Строки	4	2	
10	Файлы	2	2	
11	Основные принципы объектно-ориентированного программирования (ООП)	2	2	1
12	Этапы разработки программы	2	2	
13	Иерархия классов	4	2	
14	Разработка оконного приложения	2	2	
<b>Итого:</b>		<b>34</b>	<b>28</b>	<b>2</b>

#### 4.6. Самостоятельная работа студентов

№ п/п	Название темы	Вид СРС	Объем часов		
			Очная форма	Очно-заочная форма	Заочная форма
1	Основы алгоритмизации	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	5	7	10
2	Языки программирования	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	5	7	10
3	Типы данных	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	5	7	10
4	Операторы языка программирования	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	5	7	10
5	Процедуры и функции	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	5	7	10
6	Структуризация в программировании	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	5	6	10
7	Модульное программирование	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	5	6	10
8	Списки	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	6	6	10



9	Строки	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	6	6	10
10	Файлы	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	6	6	10
11	Основные принципы объектно-ориентированного программирования (ООП)	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	6	6	10
12	Этапы разработки программы	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	6	6	10
13	Иерархия классов	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	6	6	9
14	Разработка оконного приложения	Подготовка к лабораторным работам, к текущему и промежуточному контролю знаний и умений.	5	6	9
<b>Итого:</b>			<b>76</b>	<b>89</b>	<b>138</b>

#### **4.7. Курсовые работы/проекты.**

Курсовая работа рабочим учебным планом не предусмотрена.

#### **5. Образовательные технологии**

Преподавание дисциплины ведется с применением следующих видов образовательных технологий:

*Информационные технологии:* использование электронных образовательных ресурсов (электронный конспект, размещенный на сайте кафедры [eprs.gnomio.com](http://eprs.gnomio.com)) при подготовке к лекциям и практическим занятиям.

*Мультимедийные технологии:* презентации к лекциям.

#### **6. Формы контроля освоения дисциплины**

Текущая аттестация студентов производится в дискретные временные интервалы лектором и преподавателем(ями), ведущими и практические занятия по дисциплине в следующих формах:

собеседование (устный или письменный опрос);

тестирование;

контрольные работы.

Фонды оценочных средств, включающие типовые задания, контрольные работы, тесты и методы контроля, позволяющие оценить результаты текущей и промежуточной аттестации обучающихся по данной дисциплине, помещаются в приложении к рабочей программе в соответствии с «Положением о фонде оценочных средств».

Форма аттестации по результатам освоения дисциплины проходит в форме письменного/устного экзамена (включает в себя ответ на теоретические вопросы и решение задач) либо в сочетании различных форм (компьютерного тестирования, решения задач и пр.). Студенты, выполнившие 75% текущих и контрольных мероприятий на «отлично», а остальные 25 % на «хорошо», имеют право на получение итоговой отличной оценки.

В экзаменационную ведомость и зачетную книжку выставляются оценки по шкале, приведенной в таблице.

Шкала оценивания	Характеристика знания предмета и ответов
отлично (5)	Студент глубоко и в полном объеме владеет программным материалом. Грамотно, исчерпывающе и логично его излагает в устной или письменной форме. При этом знает рекомендованную литературу, проявляет творческий подход в ответах на вопросы и правильно обосновывает принятые решения, хорошо владеет умениями и навыками при выполнении практических задач.
хорошо (4)	Студент знает программный материал, грамотно и по сути излагает его в устной или письменной форме, допуская незначительные неточности в утверждениях, трактовках, определениях и категориях или незначительное количество ошибок. При этом владеет необходимыми умениями и навыками при выполнении практических задач.
удовлетворительно (3)	Студент знает только основной программный материал, допускает неточности, недостаточно чёткие формулировки, непоследовательность в ответах, излагаемых в устной или письменной форме. При этом недостаточно владеет умениями и навыками при выполнении практических задач. Допускает до 30% ошибок в излагаемых ответах.
неудовлетворительно (2)	Студент не знает значительной части программного материала. При этом допускает принципиальные ошибки в доказательствах, в трактовке понятий и категорий, проявляет низкую культуру знаний, не владеет основными умениями и навыками при выполнении практических задач. Студент отказывается от ответов на дополнительные вопросы.

## 7. Учебно-методическое и информационное обеспечение дисциплины:

### а) основная литература:

1. Златопольский Д.М., Основы программирования на языке Python: учебник / Златопольский Д. М. - М.: ДМК Пресс, 2017. - 284 с. - ISBN 978-5-97060-552-3 - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <http://www.studentlibrary.ru/book/ISBN9785970605523.html>

2. Шелудько В.М., Основы программирования на языке высокого уровня Python: учебное пособие / Шелудько В. М. - Ростов н/Д: Изд-во ЮФУ, 2017. - 146 с. - ISBN 978-5-9275-2649-9 - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <http://www.studentlibrary.ru/book/ISBN9785927526499.html>

3. Рик Гаско, Простой учебник программирования / Рик Гаско - М.: СОЛОН-ПРЕСС, 2018. - 320 с. (Серия "Программирование") - ISBN 978-5-91359-281-1 - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <http://www.studentlibrary.ru/book/ISBN9785913592811.html>

4. Комлев Н.Ю., Объектно Ориентированное Программирование. Настольная книга программиста / Комлев Н. Ю. - М.: СОЛОН-ПРЕСС, 2018. - 298 с. - ISBN 978-5-91359-276-7 - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <http://www.studentlibrary.ru/book/ISBN9785913592767.html>

#### **б) дополнительная литература:**

1. Варфоломеева Т.Н., Лабораторный практикум по объектно-ориентированному программированию / Т.Н. Варфоломеева, И.Ю. Ефимова. - 3-е изд., стер. - М.: ФЛИНТА, 2019. - 74 с. - ISBN 978-5-9765-2042-4 - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <http://www.studentlibrary.ru/book/ISBN9785976520424.html>

2. Варфоломеева Т.Н., Лабораторный практикум по структурному программированию / Т.Н. Варфоломеева, И.Ю. Ефимова. - 3-е изд., стер. - М.: ФЛИНТА, 2019. - 112 с. - ISBN 978-5-9765-2041-7 - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <http://www.studentlibrary.ru/book/ISBN9785976520417.html>

3. Карпович Е.Е., Языки программирования интеллектуальных систем: учеб. / Е.Е. Карпович. - М.: МИСиС, 2018. - ISBN 978-5-906953-51-3 - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <http://www.studentlibrary.ru/book/ISBN9785906953513.html>

4. Златопольский Д.М., Подготовка к ЕГЭ по информатике. Решение задач по программированию / Златопольский Д. М. - М.: ДМК Пресс, 2017. - 252 с. - ISBN 978-5-97060-598-1 - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <http://www.studentlibrary.ru/book/ISBN9785970605981.html>

5. Шелудько В.М., Язык программирования высокого уровня Python. Функции, структуры данных, дополнительные модули: учебное пособие / Шелудько В. М. - Ростов н/Д: Изд-во ЮФУ, 2017. - 107 с. - ISBN 978-5-9275-2648-2 - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <http://www.studentlibrary.ru/book/ISBN9785927526482.html>

6. Рутковская А.Э., Офисное программирование. Лабораторный практикум: учеб. пособие / А.Э. Рутковская - Минск: РИПО, 2017. - 146 с. - ISBN 978-985-503-705-8 - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <http://www.studentlibrary.ru/book/ISBN9789855037058.html>

7. Зубкова Т.М., Технология разработки программного обеспечения: учебное пособие / Зубкова Т.М. - Оренбург: ОГУ, 2017. - 468 с. - ISBN 978-5-7410-1785-2 - Текст: электронный // ЭБС "Консультант студента": [сайт]. - URL: <http://www.studentlibrary.ru/book/ISBN9785741017852.html>

**в) методические рекомендации:**

1. Методические указания к лабораторным работам по дисциплине «Бизнес-информатика 2» для студентов направления подготовки 38.03.05 – Бизнес-информатика [Электронный ресурс] / сост. А.В. Велигура. – Луганск: ЛНУ им. В. Даля, 2019. – 69 с.

2. Методические указания к самостоятельной работе по дисциплине «Бизнес-информатика 2» для студентов направления подготовки 38.03.05 – Бизнес-информатика [Электронный ресурс] / сост. А.В. Велигура. – Луганск: ЛНУ им. В. Даля, 2019. – 29 с.

**г) Интернет-ресурсы:**

1. Министерство образования и науки Российской Федерации – <http://минобрнауки.рф/>

2. Федеральная служба по надзору в сфере образования и науки – <http://obrnadzor.gov.ru/>

3. Министерство образования и науки Луганской Народной Республики – <https://minobr.su>

4. Министерство промышленности и торговли Луганской Народной Республики – <https://www.minpromlnr.su/main.php/>

5. Министерство экономического развития Луганской Народной Республики – <https://merlnr.su/>

6. Министерство финансов Луганской Народной Республики – <https://minfinlnr.su/>

7. Народный совет Луганской Народной Республики – <https://nslnr.su>

8. Государственный комитет статистики Луганской Народной Республики – <https://www.gkslnr.su/>

9. Портал Федеральных государственных образовательных стандартов высшего образования – <http://fgosvo.ru/>

10. Федеральный портал «Российское образование» – <http://www.edu.ru/>

11. Информационная система «Единое окно доступа к образовательным ресурсам» – <http://window.edu.ru/>

12. Федеральный центр информационно-образовательных ресурсов – <http://fcior.edu.ru/>

**Электронные библиотечные системы и ресурсы**

13. Электронно-библиотечная система «Консультант студента» – <http://www.studentlibrary.ru/>

14. Электронно-библиотечная система «StudMed.ru» – <https://www.studmed.ru/>

8. Руководство по своду знаний по бизнес-анализу (BABOK 2.0.) (на рус.яз.). URL: <http://iiba.ru/chapter-1-introduction/>

15. Бизнес-информатика. Концепция ведения бизнеса нового поколения URL: [http://donntu.org/sites/default/files/documents/prezentaciya\\_napravleniya\\_podgotovki.pdf](http://donntu.org/sites/default/files/documents/prezentaciya_napravleniya_podgotovki.pdf)

16. Системы дистанционного обучения кафедры экономической кибернетики и прикладной статистики Луганского национального университета имени Владимира Даля в среде Moodle. URL: <https://ecps.gnomio.com/>

17. Методическая работа в вузе: методические указания / сост . Н. П. Пучков. – Тамбов: Изд-во ГОУ ВПО ТГТУ, 2010. – 32 с. URL: <http://window.edu.ru/resource/184/73184/files/puchkov.pdf>

**Информационный ресурс библиотеки образовательной организации**

18. Научная библиотека имени А. Н. Коняева – <http://biblio.dahluniver.ru/>

## **8. Материально-техническое и программное обеспечение дисциплины**

Для проведения лекционных занятий требуется аудитория на курс, оборудованная мультимедийным проектором с экраном.

Для проведения лабораторных занятий требуется компьютерный класс, подключенный к Интернет.

Программное обеспечение:

<b>Функциональное назначение</b>	<b>Бесплатное программное обеспечение</b>	<b>Ссылки</b>
Офисный пакет	OpenOffice 4.3.7	<a href="https://www.openoffice.org/">https://www.openoffice.org/</a>
Операционная система	UBUNTU 19.04	<a href="https://ubuntu.com/">https://ubuntu.com/</a>
Браузер	Firefox Mozilla	<a href="http://www.mozilla.org/ru/firefox/fx">http://www.mozilla.org/ru/firefox/fx</a>
Графический редактор	GIMP (GNU Image Manipulation Program)	<a href="http://www.gimp.org/">http://www.gimp.org/</a> <a href="http://gimp.ru/viewpage.php?page_id=8">http://gimp.ru/viewpage.php?page_id=8</a> <a href="http://ru.wikipedia.org/wiki/GIMP">http://ru.wikipedia.org/wiki/GIMP</a>
Редактор PDF	Adobe Acrobat Reader	<a href="https://get.adobe.com/ru/reader/">https://get.adobe.com/ru/reader/</a>
Аудиоплеер	VLC	<a href="http://www.videolan.org/vlc/">http://www.videolan.org/vlc/</a>
Язык программирования	Python	<a href="https://www.python.org/downloads/">https://www.python.org/downloads/</a>
Среда разработки	PyCharm Community	<a href="https://www.jetbrains.com/ru-ru/pycharm/download/">https://www.jetbrains.com/ru-ru/pycharm/download/</a>

## 9. Оценочные средства по дисциплине

### Паспорт оценочных средств по учебной дисциплине «Бизнес-информатика»

Перечень компетенций (элементов компетенций), формируемых в результате освоения учебной дисциплины (модуля) или практики

№ п/п	Код контролируемой компетенции	Формулировка контролируемой компетенции	Индикаторы достижений компетенции и (по реализуемой дисциплине)	Контролируемые темы учебной дисциплины, практики	Этапы формирования (семестр изучения)
5	ПК-3	Способен управлять ИТ-инфраструктурой предприятия	ПК-3.3	Тема 1 Тема 2 Тема 3 Тема 4 Тема 5 Тема 6 Тема 7 Тема 8 Тема 9 Тема 10 Тема 11 Тема 12 Тема 13 Тема 14	2

### Показатели и критерии оценивания компетенций, описание шкал оценивания

№ п/п	Код контролируемой компетенции	Индикаторы достижений компетенции (по реализуемой дисциплине)	Перечень планируемых результатов	Контролируемые разделы (темы) учебной дисциплины (модуля), практики	Наименование оценочного средства <sup>2</sup>
3	ПК-3	ПК-3.3	Знать: Методы разработки и реализации алгоритмов обработки данных на языках высокого уровня Уметь: разрабатывать	Тема 1 Тема 2 Тема 3 Тема 4 Тема 5 Тема 6 Тема 7 Тема 8 Тема 9 Тема 10 Тема 11	Собеседование (устный или письменный опрос), контрольная работа, тесты

			программы на языках высокого уровня Владеть: навыками написания и отладки программ, реализующих алгоритмы обработки данных на языках высокого уровня	Тема 12 Тема 13 Тема 14	
--	--	--	---	-------------------------------	--

**Оценочные средства по дисциплине «Бизнес-информатика»**  
**Перечень вопросов (для проведения собеседования (устный или**  
**письменный опрос))**

1. Назовите основные подсистемы системы программирования, с которыми вы работали в ходе выполнения лабораторной работы.
2. Что такое проект приложения, какие файлы входят в состав проекта консольного приложения той системы программирования, которую вы использовали при выполнении лабораторной работы?
3. Опишите порядок создания приложения в выбранной системе программирования.
4. Чем отличаются конфигурации Debug и Release сборки проекта приложения?
5. Какими средствами отладки приложений располагает система программирования, использованная вами при выполнении лабораторной работы?
6. Чем отличаются команды сборки приложений Build и Make?
7. Объясните проблему отображения кириллических символов в консольных приложениях в ОС Windows. Как решается эта проблема?
8. Какие простые типы данных вы знаете?
9. Как определить и инициализировать переменную в программе на языке Python?
10. Какие операторы для реализации разветвляющихся вычислительных процессов в языке Python вы знаете?
11. Расскажите, как работает оператор if.
12. Какие операторы цикла языка Python вам известны?
13. Чем отличаются операторы цикла с пред- и постусловием?
14. Чем оператор цикла for отличается от оператора while?
15. Как сформулировать несколько условий продолжения/выхода из цикла?
16. Как принудительно завершить работу циклического оператора?



17. Как можно организовать итерационный вычислительный процесс, не используя операторов цикла?
18. Дайте определение списка, множества, словаря?
19. В каких случаях в программах необходимо использовать списки?
20. Как определяется список в языке Python?
21. Чем отличаются изменяемые и неизменяемые типы данных?
22. В чем состоит сортировка массива? Опишите принцип работы одного из алгоритмов сортировки (по заданию преподавателя).
23. Каковы критерии эффективности работы алгоритмов сортировки? Что можно сказать об эффективности работы алгоритма сортировки «пузырьком»?
24. Что такое динамические структуры данных? В чем их преимущество по сравнению со статическим? Как выделяется память под динамические типы данных?
25. Приведите примеры основных операций над односвязным списком структур (вставка, удаление, просмотр элементов).
26. Двусвязный список: определение элемента, алгоритмы создания списка, просмотра в прямом и обратном направлении.
27. Назовите достоинства и недостатки двусвязного списка по сравнению с односвязным.
28. Каковы общие принципы работы с динамическими структурами типа стек и очередь.
29. Что такое двоичное дерево? Как определяется вершина двоичного дерева в программе? Для каких целей может быть использована подобная динамическая структура?
30. Что такое поток? Какие преимущества дает потоковый подход к вводу-выводу информации по внешние устройства?
31. Какие стандартные потоки вам известны? С какими устройствами они связаны по умолчанию?
32. Как создать собственный поток в программе? Какие режимы доступа к потокам вам известны?
33. Как осуществляется символьный ввод-вывод?
34. Что такое текущий указатель потока? Какую роль он играет? Как узнать/изменить позицию указателя?
35. Каким образом можно идентифицировать ошибки при работе с потоком?
36. Как выглядит обобщенное определение функции на языке Python?
37. Что такое формальный параметр функции, фактический параметр функции, прототип функции, сигнатура функции, описание функции?
38. Для чего дается описание функции в программе? Как оно выглядит?
39. Какова роль умалчиваемых значений параметров функции? Какие ограничения на использование умалчиваемых значений налагает синтаксис языка?
40. Что такое передача параметров в функцию по значению? По адресу? По ссылке?

41. Как передавать в функцию одномерные массивы? Двумерные массивы?
42. Перечислите основные преимущества использования функций в программе.
43. Как определяются и для чего используются указатели на функции?
44. Что такое рекурсия?
45. Приведите примеры рекурсии в различных отраслях знаний.
46. Что общего и в чем разница между циклическим и рекурсивным способами организации вычислений?
47. Объясните термины «база рекурсии» и «шаг рекурсии». Определите базу рекурсии и шаг рекурсии для своей задачи.
48. Что такое «рекурсивное заикливание»? К каким последствиям оно приводит?
49. Каково главное ограничение при использовании рекурсии?
50. Что такое явная и косвенная рекурсии?
51. Оцените, от чего зависит глубина рекурсии в алгоритме решения вашей задачи.
52. Как создать многомодульное приложение в используемой вами среде программирования?
53. Как организовать взаимодействие функций из разных модулей программы?
54. Объясните принцип модульной декомпозиции, примененный вами в вашем приложении
55. Дайте определение алгоритма. В каких сферах человеческой деятельности применимы алгоритмы?
56. Какие свойства алгоритмов вам известны? Объясните на примере разработанных вами алгоритмов суть этих принципов.
57. Какие существуют формы записи алгоритмов? Опишите их достоинства и недостатки. В каких случаях они применяются?
58. Перечислите основные правила составления алгоритмов.
59. Какие виды алгоритмических процессов вы знаете? Выделите их в собственных алгоритмах.
60. Что такое трассировка алгоритма? Для чего она применяется?

Критерии и шкала оценивания по оценочному средству собеседование (устный или письменный опрос)

Шкала оценивания (интервал баллов)	Критерий оценивания
5	собеседование (устный или письменный опрос) прошел на высоком уровне (студент в полном объеме осветил рассматриваемый вопрос, владеет профильным понятийным (категориальным) аппаратом и т.п.)
4	собеседование (устный или письменный опрос) прошел на среднем уровне (студент в целом осветил рассматриваемый вопрос, владеет профильным понятийным (категориальным) аппаратом и т.п.)
3	собеседование (устный или письменный опрос) на низком уровне (студент допустил существенные неточности, изложил материал с ошибками, не владеет в достаточной степени профильным категориальным аппаратом и т.п.)
2	собеседование (устный или письменный опрос) прошел на неудовлетворительном уровне или не представлен (студент не готов, не выполнил задание и т.п.)

**Комплект контрольных заданий**

Задание 1.

Цепочка из трех бусин, помеченных латинскими буквами, формируется по следующим правилам. В конце цепочки стоит одна из бусин А, В, С. На первом месте одна из бусин В, D, С, которой нет на третьем месте. В середине — одна из бусин А, С, Е, В, не стоящая на первом месте. Какая из перечисленных цепочек создана по этим правилам?

1)СВВ 2)ЕАС 3)BCD 4)BCB

Задание 2.

Определить значения переменных а и в после исполнения данного алгоритма, построив трассировочную таблицу.

$$a=11$$

$$a=112-a*9$$

$$в=51-33+a*2$$

$$a=a-2$$

$$a=a-в/a+39$$

где / - операция целочисленного деления.

Задание 3.

Сколько раз выполнится тело цикла, чему равны значения переменных а и в после выполнения фрагмента алгоритма?

Задание 4.

У исполнителя Утроитель две команды, которым присвоены номера:

1. Вычти 1

2. Умножь на3

Первая из них уменьшает число на экране на 1, вторая - утраивает его. Запишите порядок команд в алгоритме получения из числа 5 числа 26, содержащем не более 5 команд, указывая лишь номера команд.

Задание 5.

Алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала записывается буква, следующая в алфавите после первой буквы исходной цепочки, затем две последние буквы исходной цепочки в обратном порядке и, наконец, первая буква исходной цепочки. Получившаяся цепочка является результатом работы алгоритма. Например, если исходная цепочка символов была СОН, то результатом алгоритма будет цепочка ТНОС.

Дана цепочка символов КОШ. Какая цепочка символов получится, если к данной цепочке применить алгоритм трижды (т.е. алгоритм применяется к данной цепочке, затем к результату его работы, а затем ко второму результату работы алгоритма)?

Русский алфавит:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

Задание 6.

В формировании цепочки из четырех бусин используются некоторые правила. В конце цепочки стоит одна из бусин Р, N, Т, О. На первом — одна из бусин Р, R, Т, О; которой нет на третьем месте. На третьем месте — одна из бусин О, Р, Т, не стоящая в цепочке последней. Какая из перечисленных цепочек могла быть создана с учетом этих правил?

1)PORT 2)TTTO 3)TTOO 4)OORO

Задание 7.

Определить значения переменных а и в после исполнения данного алгоритма, построив трассировочную таблицу.

$a=24$

$b=18$

$a=a+b/3+16$

$b=a+2*b$

$a=b/a+b/2$

где / - операция целочисленного деления.

Задание 8.

Сколько раз выполнится тело цикла, чему равны значения переменных а и в после выполнения фрагмента алгоритма?

Задание 9.

У исполнителя Утроитель две команды, которым присвоены номера:

1. Вычти 2

2. Умножь на 3

Первая из НИХ уменьшает число на экране на 2, вторая - утраивает его. Запишите порядок команд в программе получения из 11 числа 13, содержащей не более 5 команд, указывая лишь номера команд.

Задание 10.

Алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала записываются первые две буквы исходной цепочки в обратном порядке, затем буква, следующая в алфавите за последней буквой исходной цепочки, и, наконец, исходная цепочка символов, записанная в обратном порядке. Получившаяся цепочка является результатом работы алгоритма. Например, если исходная цепочка символов была СОН, то результатом алгоритма будет цепочка ОСОНОС.

Дана цепочка символов НИЛ. Какая цепочка символов получится, если к данной цепочке применить алгоритм дважды (т.е. алгоритм применяется к данной цепочке, затем к результату его работы)?

Русский алфавит:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

Задание 11.

Для составления цепочек используются бусины, помеченные буквами А, В, С, D, Е. На первом месте в цепочке стоит одна из бусин А, С, Е. На втором — любая гласная, если первая буква согласная, и любая согласная, если первая гласная. На третьем месте — одна из бусин С, D, Е, не стоящая в цепочке на первом месте. Какая из перечисленных цепочек создана по этим правилам?

1)СВЕ 2)ADD 3)ECE 4)EAD

Задание 12.

Определить значения переменных а и в после выполнения данного алгоритма, построив трассировочную таблицу.

$a=15$

$b=2*a+24$

$a= b/2+a$

$b=2*a+2*b+9$

$a=b/a+2*a$

где / - операция целочисленного деления.

Задание 13.

Сколько раз выполнится тело цикла, чему равны значения переменных I и A после выполнения фрагмента алгоритма?

Задание 14.

У исполнителя Калькулятор две команды, которым присвоены номера:

1. Прибавь 2

2. Умножь на 3

Выполняя первую из них, Калькулятор прибавляет к числу на экране 2, а выполняя вторую, утраивает его. Запишите порядок команд в программе получения из 0 числа 28, содержащей не более 6 команд, указывая лишь номера команд.

Задание 15.

Алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала записывается буква, предшествующая в алфавите первой букве исходной цепочки, затем исходная цепочка символов, записанная в обратном порядке. Получившаяся цепочка является результатом

работы алгоритма. Например, если исходная цепочка символов была СОН, то результатом алгоритма будет цепочка РНОС.

Дана цепочка символов РОК. Какая цепочка символов получится, если к данной цепочке применить алгоритм трижды (т.е. алгоритм применяется к данной цепочке, затем к результату его работы, а затем ко второму результату работы)?

Русский алфавит:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

Задание 16.

Цепочка из трех бусин формируется по следующим правилам. На первом месте в цепочке стоит одна из бусин А, Б, В. На втором — одна из бусин Б, В, Г. На третьем месте — одна из бусин А, В, Г, не стоящая в цепочке на первом или втором месте. Какая из следующих цепочек создана по этим правилам?

1)АГБ 2)ВАГ 3)БГГ 4)ББГ

Задание 17.

Определить значения переменных а и в после исполнения данного алгоритма, построив трассировочную таблицу.

$a=48$

$v=a/4-5$

$a= a-v$

$v=2*a+37$

$a=v/a+2*a$

где / - операция целочисленного деления.

Задание 18.

Сколько раз выполнится тело цикла, чему равны значения переменных I, A и R после выполнения фрагмента алгоритма?

Задание 19.

Исполнитель Калькулятор имеет только две команды, которым присвоены номера:

1. Вычти 3

2. Умножь на 2

Выполняя команду номер 1, Калькулятор вычитает из числа на экране 3, а выполняя команду номер 2, умножает число на экране на 2. Напишите программу, содержащую не более 5 команд, которая из числа 5 получает число 25. Укажите лишь номера команд.

Задание 20.

Алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала записывается буква, предшествующая в алфавите последней букве исходной цепочки, затем первые две буквы исходной цепочки, и наконец, буква, следующая в алфавите за первой буквой исходной цепочки символов. Получившаяся цепочка является результатом работы алгоритма. Например, если исходная цепочка символов была СОН, то результатом алгоритма будет цепочка МСОТ.

Дана цепочка символов АРТ. Какая цепочка символов получится, если к данной цепочке применить алгоритм дважды (т.е. алгоритм применяется к данной цепочке, затем к результату его работы)?

Русский алфавит:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

Задание 21.

Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} 4x^2 + 1, & \text{если } x < 5, \\ x - 5, & \\ 3x^2 - 2, & \text{если } x \geq 5. \end{cases}$$

Задание 22.

Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} 5x^2 + 2, & \text{если } x > -4, \\ x + 4, & \\ 3x^2 + 7, & \text{если } x \leq -4 \end{cases}$$

Задание 23.

Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} 7x^2 - 1, & \text{если } x < -3, \\ 2x + 6, & \\ 4x^2 - 5, & \text{если } x \geq -3 \end{cases}$$

Задание 24.

Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} 9x^2 + 5, & \text{если } x < -4, \\ 3x + 12, & \\ 4x^2 - 7, & \text{если } x \geq -4 \end{cases}$$

Критерии и шкала оценивания по оценочному средству «Контрольная работа»

Шкала оценивания (интервал баллов)	Критерий оценивания
5	Контрольная работа выполнена на высоком уровне (правильные ответы даны на 90-100% вопросов/задач)
4	Контрольная работа выполнена на среднем уровне (правильные ответы даны на 75-89% вопросов/задач)
3	Контрольная работа выполнена на низком уровне (правильные ответы даны на 50-74% вопросов/задач)
2	Контрольная работа выполнена на неудовлетворительном уровне (правильные ответы даны менее чем на 50%)



1. Понятие "алгоритм" произошло...
  - a. от имени ученого-математика
  - b. от латинского слова, в переводе означающего "исполнитель"
  - c. от греческого слова, в переводе означающего "по шагам"
  - d. от латинского слова, в переводе означающего "программирование"
2. Какое свойство алгоритма означает, что процесс решения задачи должен представлять последовательность простых шагов?
  - a. результативность
  - b. определенность
  - c. дискретность
  - d. массовость
3. Какое свойство алгоритма означает, что алгоритм должен приводить к решению за конечное число шагов?
  - a. результативность
  - b. определенность
  - c. дискретность
  - d. массовость
4. Какое свойство алгоритма означает, что каждое правило алгоритма должно быть четким и однозначным?
  - a. результативность
  - b. определенность
  - c. дискретность
  - d. массовость
5. Какое свойство алгоритма означает, что он должен разрабатываться в общем виде, для решения целого класса однотипных задач?
  - a. результативность
  - b. определенность
  - c. дискретность
  - d. массовость
6. Какого способа записи алгоритма не существует?
  - a. словесного
  - b. графического
  - c. естественного
  - d. программного
7. Алгоритм, в котором та или иная команда выполняется в зависимости от истинности условия...
  - a. линейный
  - b. разветвляющийся
  - c. циклический
8. Графическое представление алгоритма
  - a. таблица
  - b. блок-схема
  - c. чертеж

- d. иерархия
9. Какой тип блоков не используется в линейном алгоритме?
- блок условия
  - блок действия
  - блок ввода
  - блок вывода
10. Выберите условие принадлежности точки  $x$  отрезку  $[a;b]$
- $(x \leq b) \text{ OR } (x \geq a)$
  - $(x \leq b) \text{ AND } (x \geq a)$
  - $(x \leq a) \text{ OR } (x \geq b)$
  - $(x \leq a) \text{ AND } (x \geq b)$
11. Выберите условие  $x$  лежит вне отрезка  $[a;b]$
- $(x < b) \text{ OR } (x > a)$
- $(x < b) \text{ AND } (x > a)$
  - $(x < a) \text{ OR } (x > b)$
  - $(x \leq a) \text{ AND } (x \geq b)$
12. Какой логической операции в алгебре множеств соответствует операция объединения множеств?
- дизъюнкции
  - конъюнкции
  - импликации
  - инверсии
13. Какой логической операции в алгебре множеств соответствует операция пересечения множеств?
- дизъюнкции
  - конъюнкции
  - импликации
  - инверсии
14. Заголовок цикла определяет...
- количество повторений цикла
  - набор команд для повторения
  - условие печати результатов
  - параметры цикла
15. Составное высказывание ложно тогда и только тогда, когда оба элементарных высказывания ложны. О какой логической операции идет речь?
- конъюнкция
  - дизъюнкция
  - импликация
  - эквиваленция
16. Составное высказывание истинно тогда и только тогда, когда оба элементарных высказывания истинны. О какой логической операции идет речь?
- конъюнкция
  - дизъюнкция

- c. импликация
  - d. эквиваленция
17. Что такое тело цикла?
- a. параметр цикла
  - b. условие выхода из цикла
  - c. первая строка циклической структуры
  - d. набор команд для повторения
18. Какое свойство алгоритма не выполняется, когда входными данными являются константы?
- a. результативность
  - b. определенность
  - c. дискретность
  - d. массовость
19. Какого вида ошибок при отладке программы не существует?
- a. грамматические
  - b. синтаксические
  - c. логические
20. Алгоритмическая конструкция выбор используется для...
- a. определения условия выхода из цикла
  - b. перечня констант и переменных
  - c. организации множественного ветвления
  - d. определения тела цикла
21. В каком виде циклической структуры тело цикла выполнится хотя бы один раз?
- a. цикл с параметром
  - b. цикл с предусловием
  - c. цикл с постусловием
22. Для какого из указанных значений числа  $X$  истинно выражение  $\text{NOT}(X < 2) \text{ AND } (X < 3)$ ?
- a. Ответ: 1
  - b. Ответ: 2
  - c. Ответ: 3
  - d. Ответ: 4
23. Для какого из указанных значений числа  $X$  истинно выражение  $\text{NOT}(X > 2) \text{ OR } (X > 6)$ ?
- a. Ответ: 2
  - b. Ответ: 3
  - c. Ответ: 4
  - d. Ответ: 5
- Языки программирования - это...
- a. искусственные языки
  - b. естественные языки
  - c. смешанные языки
24. Как называется средство для преобразования текстов из языков программирования высокого уровня в машинный код?

- d. маршрутизатор
  - e. транслятор
  - f. автокод
  - a. контроллер
25. Какой из языков программирования является машинно-зависимым?
- b. бейсик
  - a. паскаль
  - b. ассемблер
  - c. си
26. Какой язык программирования относится к языкам низкого уровня?
- a. ассемблер
  - b. бейсик
  - c. фортран
  - d. кобол
27. Какой язык программирования относится к языкам высокого уровня?
- a. ассемблер
  - b. автокод
  - c. си
28. Что такое алфавит языка программирования?
- a. фиксированный для данного языка набор символов
  - b. буквы, используемые в данном языке
  - c. правила записи объектов языка
29. Что такое идентификатор?
- a. фиксированный для данного языка набор символов
  - b. последовательность символов, применяющаяся для имен констант, переменных, функций
  - c. средство трансляции в машинный код
  - d. средство, определяющее тип данных
30. К какому типу данных относится число 23.56?
- a. string
  - b. integer
  - c. longint
  - d. real
31. Выберите пример правильного имени переменной:
- a. FFG11
  - b. 6PO
  - c. АПРЛ
  - d. FF+HH
32. Выберите пример неправильного имени переменной:
- a. DFR56
  - b. ИВЛ23
  - c. \_VFA
  - d. DH55DF
33. Выберите идентификатор целого типа данных

- a. word
  - b. double
  - c. extended
  - d. char
34. Выберите идентификатор логического типа данных
- a. word
  - b. boolean
  - c. extended
  - d. char
35. Выберите пример описания интервального типа данных
- a. type season = (winter, spring, summer, autumn);
  - b. var color: (red, yellow, green);
  - c. var fad: = 100..200;
  - d. var cad: integer;
36. Чему равен результат операции  $(47 \bmod 15)$ ?
- a. 7.
  - b. 2.
  - c. 4.
  - d. 10.
37. Величина, не изменяющая свое значение в процессе выполнения программы, называется...
- a. константа
  - b. переменная
  - c. функция
  - d. выражение
38. Чему равен результат операции  $(67 \operatorname{div} 10)$ ?
- a. 7.
  - b. 6.
  - c. 4.
  - d. 10.
39. Чему равно значение выражения  $\operatorname{Frac}(45.56)$ ?
- a. 46.
  - b. 45.
  - c. 0.45
  - d. 0.56
40. Чему равно значение выражения  $\operatorname{Inc}(5,5)$ ?
- a. 5.
  - b. 10.
  - c. 0.
  - d. 55.
41. Чему равно значение выражения  $\operatorname{Dec}(15,10)$ ?
- a. 25.
  - b. 10.
  - c. 5.
  - d. 15.

42. Чему равно значение выражения  $\text{Inc}(20)$ ?
- 19.
  - 10.
  - 2.
  - 21.
43. Чему равно значение выражения  $\text{Dec}(15)$ ?
- 15.
  - 14.
  - 16.
  - 25.
44. Функция SQR предназначена для:
- возведения аргумента в третью степень
  - вычисления квадратного корня из аргумента
  - возведения аргумента во вторую степень
  - вычисления корня третьей степени из аргумента
45. С понятием переменная не связано следующее понятие:
- тип
  - идентификатор
  - объект
  - значение
46. Внутреннюю форму представления данных в ЭВМ определяет:
- идентификатор
  - транслятор
  - тип данных
  - операция
47. Раздел описания переменных задается служебным словом:
- const
  - uses
  - type
  - var
48. Тело программы начинается после служебного слова:
- begin
  - var
  - uses
  - type

#### Критерии и шкала оценивания по оценочному средству тесты

Шкала оценивания (интервал баллов)	Критерий оценивания
5	Тесты выполнены на высоком уровне (правильные ответы даны на 90-100% тестов)
4	Тесты выполнены на среднем уровне (правильные ответы даны на 75-89% тестов)
3	Тесты выполнены на низком уровне (правильные ответы даны на 50-74% тестов)
2	Тесты выполнены на неудовлетворительном уровне (правильные ответы даны менее чем на 50% тестов)

## Оценочные средства для промежуточной аттестации

### Вопросы к экзамену:

1. Алгоритмы: определение, свойства.
2. Жизненный цикл программного обеспечения.
3. Обзор языков программирования.
4. Способы описания алгоритмов.
5. Структура программы на языке Python.
6. Типы данных в языке Python. Определение простых переменных в программе.
7. Классы памяти объектов языка Python.
8. Арифметические операции языка Python.
9. Логические операции языка Python
10. Операции присваивания, сложного присваивания, понятие леводопустимого выражения.
11. Операции языка Python: приведения типа, условная, перечисления, взятия размера.
12. Приоритеты операций языка Python.
13. Операторы языка Python: выражение, составной, пустой.
14. Операторы языка Python: условный, оператор-переключатель.
15. Операторы циклов в языке Python.
16. Оператор продолжения, оператор завершения.
17. Одномерные массивы в Python: определение, способы инициализации.
18. Типовые операции с одномерными массивами: поиск минимума, сумма элементов
19. Двумерные массивы в языке Python: определение, способы инициализации.
20. Сортировка одномерного массива методом «пузырька».
21. Сортировка одномерного массива методом выбора.
22. Сортировка одномерного массива методом вставки.
23. Указатели: назначение, определение, инициализация.
24. Динамическое выделение памяти.
25. Арифметические операции с указателями.
26. Функции ввода-вывода в.
27. Связь массивов и указателей.
28. Динамическое выделение памяти под одномерные массивы.
29. Динамическое выделение памяти под двумерные массивы.
30. Универсальный метод выделения динамической памяти под двумерные массивы.
31. Строки в языке Python: определение, инициализация, кодирование символов.
32. Терминатор строки, его роль при работе со строками.
33. Функции для работы со строками языка Python: взятие размера, объединения и копирования строк.



34. Функции для работы со строками языка Python: сравнение строк, поиск подстроки.
35. Структурный тип данных: определение типа и переменных, инициализация структур.
36. Указатели на структуры. Динамическое выделение памяти под структуры.
37. Вложенные структуры, массивы структур. Динамическое выделение памяти под массивы структур.
38. Битовые поля структур: определения, преимущества, ограничения при работе.
39. Объединения: определение, отличие от структур, назначение.
40. Вариативные структуры: обобщенное определение, преимущества, примеры использования.
41. Перечислимый тип данных: определение, назначение, пример использования.
42. Вариативные структуры: обобщенное определение, преимущества, примеры использования.
43. Перечислимый тип данных: определение, назначение, пример использования.
44. Динамические списки: назначение, основные принципы организации.
45. Односвязный список: определение, особенности использования, типовые алгоритмы обработки.
46. Сравнительный анализ односвязного списка и массива структур.
47. Двусвязный список: определение, особенности использования, типовые алгоритмы обработки.
48. Поточковый ввод-вывод. Определение потока, стандартные потоки, типы потоков.
49. Открытие потоков: режимы открытия, различия текстовых и двоичных потоков.
50. Основные функция контроля состояния потоков.
51. Основные функции ввода строковой информации в поток и вывода информации из потока.
52. Форматированный потоковый ввод-вывод.
53. Ввод-вывод с использованием буфера. Отличия от форматированного ввода-вывода.
54. Функции: определение, прототип, вызов.
55. Основные преимущества структурного стиля программирования.
56. Формальные и фактические параметры. Связь формальных и фактических параметров функций.
57. Порядок вызова функции и возврата из нее. Передача параметров по значению и по адресу.
58. Функции и массивы: массивы как параметры функций.
59. Функции с умалчиваемыми значениями параметров.
60. Передача параметров функции main.

61. Ссылки: назначение, определение, использование.
62. Рекурсивные вызовы функций: назначение, примеры рекурсивных задач, прямая и косвенная рекурсия.
63. Разворачивание и свертывание рекурсии в памяти, глубина рекурсии (на практическом примере).
64. Алгоритм быстрой обменной сортировки, его рекурсивная реализация
65. Указатели на функции: определение, назначение, примеры использования.
66. Указатели на функции как параметры функций, сортировка массивов с использованием qsort.
67. Перегрузка функций: определение, достоинства, ограничения. Примеры реализации.
68. Препроцессор: назначение, основные этапы препроцессорной обработки.
69. Препроцессор: директивы включения файлов и замены идентификаторов.
70. Препроцессор: макроопределения с параметрами.
71. Препроцессор: сравнение функций и макросов.
72. Препроцессор: директивы условной компиляции.
73. Основы объектно-ориентированной парадигмы программирования: инкапсуляция, наследование, полиморфизм.
74. Определение классов и объектов.
75. Область видимости компонент класса.
76. Правила наследования классов.

### **Примеры экзаменационных задач**

#### **ВАРИАНТ 1**

1. Составьте программу вычисления количества букв «А» в предложении.
2. Составьте программу, определяющую, на какую букву начинается второе слово в предложении.
- 3\*. Составьте программу, определяющую длину первого слова в предложении.

#### **ВАРИАНТ 2**

1. Составьте программу вычисления количества букв «Е» в предложении.
2. Составьте программу, определяющую, на какую букву начинается третье слово в предложении.
- 3\*. Составьте программу, определяющую длину последнего слова в предложении.

#### **ВАРИАНТ 3**

1. Составьте программу вычисления количества букв «К» в предложении.

2. Составьте программу, определяющую, на какую букву заканчивается второе слово в предложении.

3\*. Составьте программу, определяющую длину самого длинного слова в предложении.

#### ВАРИАНТ 4

1. Составьте программу вычисления количества букв «М» в предложении.

2. Составьте программу, определяющую, на какую букву заканчивается третье слово в предложении.

3\*. Составьте программу, определяющую длину самого короткого слова в предложении.

#### ВАРИАНТ 5

1. Составьте программу, заменяющую букву «А» на «О» в предложении.

2. Составьте программу, определяющую, какая из букв, «Е» или «О» встречается чаще в предложении.

3\*. Составьте программу, определяющую, является ли данное предложение вопросительным.

#### ВАРИАНТ 6

1. Составьте программу, заменяющую букву «Е» на «А» в предложении.

2. Составьте программу, определяющую, какая из букв, «А» или «О» встречается реже в предложении.

3\*. Составьте программу, определяющую, является ли данное предложение восклицательным.

#### ВАРИАНТ 7

1. Составьте программу, заменяющую слог «ма» на «ко» в предложении.

2. Составьте программу, определяющую, какая из букв, «М» или «К» встречается чаще в предложении.

3\*. Составьте программу, определяющую, с гласной ли буквы начинается предложение.

#### ВАРИАНТ 8

1. Составьте программу, заменяющую слог «ка» на «ме» в предложении.

2. Составьте программу, определяющую, какая из букв, «Т» или «В» встречается реже в предложении.

3\*. Составьте программу, определяющую, на гласную ли букву заканчивается предложение.

#### ВАРИАНТ 9

1. Составьте программу, удаляющую букву «Е» из предложения.

2. Составьте программу, вставляющую после буквы «Н» еще букву «Н».

3\*. Составьте программу, печатающую слово в обратном порядке.

#### ВАРИАНТ 10

1. Составьте программу, удаляющую букву «А» из предложения.
2. Составьте программу, вставляющую после буквы «К» еще букву «К».
- 3\*. Составьте программу, печатающую число в обратном порядке.

#### ВАРИАНТ 11

1. Составьте программу, удаляющую слог «КА» из предложения.
2. Составьте программу, вставляющую после буквы «К» слог «ВА».
- 3\*. Составьте программу, печатающую первое слово в предложении в обратном порядке.

#### ВАРИАНТ 12

1. Составьте программу, удаляющую слог «МА» из предложения.
2. Составьте программу, вставляющую после буквы «О» слог «КО».
- 3\*. Составьте программу, печатающую последнее слово в предложении в обратном порядке.

#### ВАРИАНТ 13

1. Составьте программу, определяющую сумму цифр числа.
2. Составьте программу, печатающую строку: «AABB...ZZ».
- 3\*. Составьте программу, распечатывающую слова предложения в алфавитном порядке.

#### ВАРИАНТ 14

1. Составьте программу, определяющую произведение цифр числа.
2. Составьте программу, печатающую строку: «ZZYY...AA».
- 3\*. Составьте программу, распечатывающую слова предложения в порядке, обратном алфавитному.

#### ВАРИАНТ 15

1. Составьте программу, определяющую сумму четных цифр числа.
2. Составьте программу, печатающую строку: «ABBCDD...Z».
- 3\*. Составьте программу, распечатывающую слова предложения, которые начинаются и заканчиваются на одну букву.

#### ВАРИАНТ 16

1. Составьте программу, определяющую сумму нечетных цифр числа.
2. Составьте программу, печатающую строку: «ZZYXX...A».
- 3\*. Составьте программу, распечатывающую слова предложения, которые содержат одинаковое количество букв.

## Критерии и шкала оценивания к промежуточной аттестации «экзамен»

Шкала оценивания	Характеристика знания предмета и ответов
отлично (5)	Студент глубоко и в полном объеме владеет программным материалом. Грамотно, исчерпывающе и логично его излагает в устной или письменной форме. При этом знает рекомендованную литературу, проявляет творческий подход в ответах на вопросы и правильно обосновывает принятые решения, хорошо владеет умениями и навыками при выполнении практических задач.
хорошо (4)	Студент знает программный материал, грамотно и по сути излагает его в устной или письменной форме, допуская незначительные неточности в утверждениях, трактовках, определениях и категориях или незначительное количество ошибок. При этом владеет необходимыми умениями и навыками при выполнении практических задач.
удовлетворительно (3)	Студент знает только основной программный материал, допускает неточности, недостаточно четкие формулировки, непоследовательность в ответах, излагаемых в устной или письменной форме. При этом недостаточно владеет умениями и навыками при выполнении практических задач. Допускает до 30% ошибок в излагаемых ответах.
неудовлетворительно (2)	Студент не знает значительной части программного материала. При этом допускает принципиальные ошибки в доказательствах, в трактовке понятий и категорий, проявляет низкую культуру знаний, не владеет основными умениями и навыками при выполнении практических задач. Студент отказывается от ответов на дополнительные вопросы.

### Лист изменений и дополнений

№ п/п	Виды дополнений и изменений	Дата и номер протокола заседания кафедры (кафедр), на котором были рассмотрены и одобрены изменения и дополнения	Подпись (с расшифровкой) заведующего кафедрой (заведующих кафедрами)