

МИНОБРНАУКИ РОССИИ
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

**РАБОЧАЯ ПРОГРАММА
УЧЕБНОЙ ДИСЦИПЛИНЫ**

ОП.О4 Основы алгоритмизации и программирования

программы подготовки специалистов среднего звена /
квалифицированных рабочих и служащих

09.02.07 Информационные системы и программирование

Форма обучения: *очная*

Находка 2026

Рабочая программа учебной дисциплины *ОП.04 Основы алгоритмизации и программирования* разработана в соответствии с требованиями Федерального государственного образовательного стандарта среднего профессионального образования по специальности / профессии *09.02.07 Информационные системы и программирование*, утвержденного приказом Минобрнауки России от «09» декабря 2016 г. № 1547, примерной образовательной программой.

Разработчик(и): *Лихацкий М.Б. преподаватель ОСПО филиала ФГБОУ ВО «ВВГУ» в г. Находке*

Рассмотрено и одобрено на заседании цикловой методической комиссии

Протокол № 10 от «29» июня 2026 г.

Председатель ЦМК _____ *В.В Плахотина*
подпись

Содержание

- 1 ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ
УЧЕБНОЙ ДИСЦИПЛИНЫ**
- 2 СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ**
- 3 УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ**
- 4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ
УЧЕБНОЙ ДИСЦИПЛИНЫ**

1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

1.1 Место дисциплины в структуре основной образовательной программы

Учебная дисциплина *ОП.04 Основы алгоритмизации и программирования* является частью учебного цикла основной образовательной программы (далее ООП) в соответствии с ФГОС СПО по специальности/профессии *09.02.07 Информационные системы и программирование*.

1.2 Цель и планируемые результаты освоения дисциплины

По итогам освоения дисциплины, обучающиеся должны продемонстрировать результаты обучения, соотнесённые с результатами освоения ООП СПО, приведенные в таблице.

Код компетенции	Умения	Знания
ОК 1 ОК 2	Разрабатывать алгоритмы решения задач с использованием структурного и модульного подходов. Осуществлять ввод, редактирование, компиляцию и отладку программ на языке высокого уровня.	Понятие алгоритма, свойства алгоритмов, способы описания (блок-схемы, псевдокод). Основные алгоритмические конструкции (линейные, ветвления, циклы).
ОК 4 ОК 5	Использовать базовые структуры данных (массивы, строки, списки, стеки, очереди).	Структуры данных и методы их реализации. Синтаксис и семантику языка программирования
ОК 9 ОК 10	Использовать инструментальные среды разработки для написания и отладки кода.	Методы разработки и отладки программ.
ПК 5.1 ПК 5.2 ПК 5.6 ПК 6.3	Разрабатывать модули программного кода для информационных систем. Использовать стандартные алгоритмы и структуры данных при решении прикладных задач. Проводить тестирование и отладку разработанных модулей. Документировать разработанные алгоритмы и программы.	Основные алгоритмы обработки массивов, строк, файлов. Принципы модульного программирования и повторного использования кода. Методы и средства тестирования и отладки программ. Правила оформления программной документации.

2 Структура и содержание учебной дисциплины

2.1 Объем учебной дисциплины и виды учебной работы

Таблица 2 - Распределение трудоемкости дисциплины по видам учебной работы

Вид учебной работы	Объем часов
Объем образовательной программы учебной дисциплины	142
в том числе:	
– теоретическое обучение	66
– практические занятия <i>(если предусмотрено)</i>	66
– самостоятельная работа	
– консультации	
– промежуточная аттестация – <i>(дифференцируемый зачет, экзамен)</i>	10

2.2 Тематический план и содержание учебной дисциплины «ОП.04 Основы алгоритмизации и программирования»

Таблица 3 - Тематический план дисциплины

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем часов	Коды компетенций, формированию которых способствует элемент программы
1	2	3	4
Тема 1.1 Основы алгоритмизации	Содержание учебного материала: Введение в дисциплину. Основные понятия. Определения. Алгоритм. Свойства алгоритма. Способы представления алгоритма.	16	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 5.1, 5.2, 5.6, ПК 6.3
	Практическое занятие №1: Разработка линейных алгоритмов. Практическое занятие №2: Разработка разветвляющихся алгоритмов Практическое занятие №3: Разработка циклических алгоритмов	16	
	Самостоятельная работа обучающихся	Не предусмотрено	
	Лабораторное занятие	Не предусмотрено	
Тема 1.2. Основы программирования	Содержание учебного материала: Развитие языков программирования. Области применения языков программирования. Классификация языков программирования. Понятие интегрированной среды разработки. Структура редактора VBA. Структура программы VBA. Структура модуля, назначение разделов. Типы данных VBA. Константы. Переменные. Способы объявления. Область видимости переменной. Время жизни переменной.	16	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 5.1, 5.2, 5.6, ПК 6.3
	Практическое занятие №4: Изучение среды разработки Практическое занятие №5: Изучение режимов и инструментов отладки программы Практическое занятие №6: Разработка линейной программы Практическое занятие №7: Изучение свойств диалоговых окон	16	
	Самостоятельная работа обучающихся	Не предусмотрено	
	Лабораторное занятие	Не предусмотрено	
Промежуточная аттестация - дифференцированный зачет		2	

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем часов	Коды компетенций, формированию которых способствует элемент программы
1	2	3	4
Тема 1.3. Основы процедурного программирования	Содержание учебного материала: Стандартные функции и процедуры. Разработка линейной программы. Понятие функции и подпрограммы. Объявление и вызов процедур. Типы параметров. Организация ветвления в программе. Инструкции цикла. Разработка циклической программы. Одномерный, многомерный и динамический массивы. Алгоритмы поиска и сортировки массивов	17	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 5.1, 5.2, 5.6, ПК 6.3
	Практическое занятие №8: Разработка процедур пользователя Практическое занятие №9: Разработка разветвляющейся программы Практическое занятие №10: Разработка циклической программы Практическое занятие №11: Ввод, отладка и решение задач с массивом данных Практическое занятие №12: Ввод, отладка и решение задач поиска и сортировки в массиве	17	
	Самостоятельная работа обучающихся	Не предусмотрено	
	Лабораторное занятие	Не предусмотрено	
Тема 1.4. Основы объектноориентированного программирования	Содержание учебного материала: VBA как объектно-ориентированный язык программирования. Пользовательская форма. Свойства, методы, события формы. Элементы управления формы. Свойства, методы и события элементов управления. Создание формы на добавление, просмотр, редактирование данных.	17	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 5.1, 5.2, 5.6, ПК 6.3
	Практическое занятие №13: Разработка пользовательской формы Практическое занятие №14: Изучение свойств, методов и событий элементов управления формы Практическое занятие №15: Разработка формы на просмотр, редактирования и добавления данных	17	
	Самостоятельная работа обучающихся	Не предусмотрено	
	Лабораторное занятие	Не предусмотрено	
Промежуточная аттестация – экзамен, дифференцированный зачет		8	

ИТОГО часов		142	
Теоретические занятия		66	
Практические занятия		66	
Самостоятельная работа		0	
Дифференцированный зачет		10	

3 Условия реализации учебной дисциплины

3.1 Для реализации программы учебной дисциплины «ОП.04 Основы алгоритмизации и программирования» образовательной организации, предусмотрено наличие следующих специальных помещений:

Для реализации программы учебной дисциплины предусмотрено наличие следующих специальных помещений:

Компьютерный класс информационных технологий: количество посадочных мест - 28 шт., доска меловая - 1 шт., стол для преподавателя - 1 шт., стул для преподавателя - 1 шт.; учебно-методические пособия, дидактические и презентационные материалы

Компьютерный класс имеет следующее оборудование:

Учебно-лабораторное оборудование: Процессор Intel Количество ядер - 2 . Объем кэш-памяти - 8Мб • Базовая частота процессора - 2,0 ГГц. Объем установленной памяти - 4Гб Накопитель твердотельный объемом 512 видео карта - встроенная. Максимальное разрешение - 1920x1080 пикс. Диагональ монитора 23" количество мониторов 1 шт. Разрешение: FHD (1920x1080 пикс.) клавиатура мышь ОС Microsoft Windows 7 Pro Мультимедийный комплект в составе: проектор Sanyo, крепление Kromax, экран Lumien Eco Picture: Технология проецирования изображения - микрозеркала на полупроводниковом чипе. Базовое разрешение проектора: 1280x800 пикс. Входы - VGA, HDMI 2 шт., Порты - RS-232C, USB 2 шт., LAN Размер полотна: ширина 200 см, высота 127 см. ПО: Microsoft Office 2010, Пакет Adobe CC 2019, Pixologic Zbrush Academic Volume License, Adobe Substance Painter, Autodesk 3Ds max, Maya, Blender, Unity, Unreal Engine

3.2 Информационное обеспечение реализации программы

Для реализации программы учебной дисциплины библиотечный фонд ВВГУ укомплектован печатными и электронными изданиями.

Обучающиеся из числа инвалидов и лиц с ограниченными возможностями здоровья обеспечены печатными и (или) электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

Основная литература:

1. Бизяев, А.А. Информационные технологии: практикум: [Электронный ресурс]/А.А. Бизяев, К.А. Куратов ; Новосибирский государственный технический университет. - Новосибирск: Новосибирский государственный технический университет, 2022. - 96 с.: ил., табл. - Режим доступа: <http://biblioclub.ru/index.php?page=book&id=575330>

2. Информационные технологии [Электронный ресурс] : метод. рекомендации по самостоят. изучению дисциплины / Г.И. Попов, С.И. Бажинов, В.Г. Конюхов, Е.Н. Яшкина, В.А. Пирогов .- М. : РГУФКСМиТ, 2022 .-167 с. : ил. - Режим доступа: <https://lib.rucont.ru/efd/671375>

3. Информационные технологии в профессиональной деятельности [Электронный ресурс] / Т.В. Марзаева .— Улан-Удэ : Бурятский государственный университет, 2025 .- 71 с. -Режим доступа: <https://lib.rucont.ru/efd/348195>

4. Информационные технологии: учеб. пособие [Электронный ресурс]/ Л.Г. Гагарина, Я.О. Теплова, Е.Л. Румянцева, А.М. Баин / под ред. Л.Г. Гагариной. — Москва: ИД «ФОРУМ»: ИНФРА- М, 2021. — 320 с. -Режим доступа: <https://znanium.com/catalog/product/471464>.

5. Филимонова, Е.В. Информационные технологии в профессиональной деятельности : учебник / Филимонова Е.В. — Москва : КноРус, 2022. — 482 с. — (СПО). — ISBN 978-5-406-04887-0. — URL: <https://book.ru/book/922139> (дата обращения:

06.08.2024). — Текст : электронный.

6. Хныкина, А.Г. Информационные технологии [Электронный ресурс] : учеб. пособие / Т.В. Минкина, А.Г. Хныкина .— Ставрополь : изд-во СКФУ, 2023 .— 126 с. : ил. — Режим доступа: <https://lib.rucont.ru/efd/671178>

Дополнительная литература:

1. Богданова, С.В. Информационные технологии [Электронный ресурс]: учебное пособие / С.В. Богданова, А.Н. Ермакова. - Ставрополь: Сервисшкола, 2024. - 211 с. - Режим доступа: <https://znanium.com/catalog/product/514867>

2. Самойленко, А.П. Информационные технологии статистической обработки данных [Электронный ресурс] : учеб. пособие / О.А. Усенко, Южный федеральный ун-т, А.П. Самойленко .— Ростов н/Д. : Изд-во ЮФУ, 2024 .— 128 с. : ил. — ISBN 978-5-9275-2521-8 .— Режим доступа: <https://lib.rucont.ru/efd/692393>

3. Суханова, О.Н. Информационные технологии [Электронный ресурс] : лаб. практикум / О.В.

4. Ментюкова, О.Н. Суханова .— Пенза : РИО ПГСХА, 2022 .- 116 с. : ил. - Режим доступа: <https://lib.rucont.ru/efd/323727>.

Интернет-ресурсы:

1. www.informatics.mccme.ru/moodle/ - дистанционная подготовка по информатике.
2. www.ixbt.com - сайт о высоких технологиях, новости индустрии из мира компьютерного «железа», тестовые испытания и обзоры оборудования. Конференция. Платформа ПК. Цифровой звук. Цифровое фото. Мобильные устройства. Форум.
3. www.lib.ru/INTERMET/termwww.txt - толковый словарь терминов.
4. www.computerra.ru - новости про компьютеры, железо, новые технологии, информационные технологии. Рубрики: В поле зрения; Компьютерралаб; Дюжина полезных программ; Реклама на сайте; Блоги; Наука и жизнь. Архив номеров журнала.
5. www.novtex.ru/IT/arhiv.htm
6. <https://www.planttext.com/>
7. <https://kroki.io/#>
8. <https://www.canva.com/ru ru/>
9. <https://prezi.com/>
10. www.jir.ru/ - электронный научный журнал.

4. Контроль результатов освоения учебной дисциплины

Результаты обучения	Критерии оценки	Методы оценки
- актуальный профессиональный и социальный контекст, в котором приходится работать и жить; - основные источники информации и ресурсы для решения задач и проблем в профессиональном и/или социальном контексте; - алгоритмы выполнения работ в профессиональной и смежных областях; - методы работы в профессиональной и смежных сферах; - структуру плана для решения задач; - порядок оценки результатов решения задач профессиональной деятельности - основные методы и средства сбора, обработки, хранения, передачи и накопления информации; - технологию поиска информации в сети Интернет; - номенклатура информационных источников применяемых в профессиональной деятельности; приемы структурирования информации; - формат оформления результатов поиска информации - содержание актуальной нормативно-правовой документации;	Уровень правильных ответов при письменном и устном контроле, четкость изложения материала. Быстрота ориентации в представляемом материале, быстрота реакции на вопросы. Правильность, полнота выполнения заданий, точность формулировок, точность расчетов, соответствие требованиям.	устный опрос, практические работы, тестирование

- современная научная и профессиональная терминология; - возможные траектории профессионального развития и самообразования - психологические основы деятельности коллектива, - психологические особенности личности; - основы проектной деятельности - особенности социального и культурного контекста; - правила оформления документов и построения устных сообщений - назначение, состав, основные характеристики организационной и компьютерной техники;		
- основные компоненты компьютерных сетей, принципы пакетной передачи данных, организацию межсетевого взаимодействия; - назначение и принципы использования системного и прикладного программного обеспечения; - принципы защиты информации от несанкционированного доступа; - правовые аспекты использования информационных технологий и программного обеспечения; - основные понятия автоматизированной	Оптимальность выбора способов действий, методов, последовательностей действий. Соответствие требованиям инструкций, регламентов, рациональность действий. Точность распознавания проблемных ситуаций в различных контекстах; Адекватность анализа сложных ситуаций при решении задач профессиональной деятельности, эффективность поиска. Точность определения источников нужных ресурсов. Разработка детального плана действий, правильность оценки рисков на каждом шагу; Точность оценки плюсов и минусов полученного результата, своего плана и его реализации, предложение критериев оценки и рекомендаций по улучшению плана	Текущий контроль: экспертное наблюдение и оценка в процессе выполнения заданий для практических занятий; Промежуточная аттестация: тестирование

Для оценки достижения запланированных результатов обучения по дисциплине разработаны контрольно-оценочные средства для проведения текущего контроля и промежуточной аттестации, которые прилагаются к рабочей программе дисциплины.

Для оценки достижения запланированных результатов обучения по дисциплине разработаны контрольно-оценочные средства для проведения текущего контроля и промежуточной аттестации, которые прилагаются к рабочей программе дисциплины.

Оценка результатов обучения производится при помощи бально-рейтинговой системы.

Основные критерии оценки:

- конспектирование 0,5 баллов;
- активность на уроке 0,5 баллов;
- посещаемость занятий 0,5 баллов;
- выполнение индивидуальных домашних заданий 1 задание- 1 балл;
- выполнение самостоятельной работы 1 балл;
- лабораторная работа 3 балла;
- контрольная работа 1 задание -1 балл;
- участие в научно-исследовательской работе 15 баллов;
- дифференцированный зачет 20 баллов.

МИНОБРНАУКИ РОССИИ
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

для проведения текущего контроля и промежуточной аттестации
по учебной дисциплине

ОП.О4 Основы алгоритмизации и программирования

программы подготовки специалистов среднего звена /
квалифицированных рабочих и служащих

09.02.07 Информационные системы и программирование

Форма обучения: очная

Оценочные средства для проведения текущего контроля и промежуточной аттестации по учебной дисциплине ОП.О4 «Основы алгоритмизации и программирования» разработаны в соответствии с требованиями ФГОС СПО по специальности / профессии 09.02.07 Информационные системы и программирование, утвержденного приказом Минобрнауки России от «09» декабря 2016 г. № 1547, примерной образовательной программой.

Разработчик(и): *Лихацкий М.Б. преподаватель ОСПО филиала ФГБОУ ВО «ВВГУ» в г. Находке*

Рассмотрено и одобрено на заседании цикловой методической комиссии

Протокол № 10 от «29» июня 2026 г.

Председатель ЦМК _____ *В.В Плахотина*
подпись

1 Общие сведения

Фонд оценочных средств (далее – ФОС) предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины ОП.О4 «Основы алгоритмизации и программирования».

ФОС включают в себя контрольные материалы для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине, которая проводится в форме дифференцированного зачёта (с использованием оценочного средства - *тестирование*)

2 Планируемые результаты обучения по дисциплине, обеспечивающие результаты освоения образовательной программы

Код ОК, ПК	Код результата обучения	Наименование результата обучения ¹
ОК 1 ОК 2	31	Понятие алгоритма, свойства алгоритмов, способы описания (блок-схемы, псевдокод)
ОК 4 ОК 5	32	Основные алгоритмические конструкции (линейные, ветвления, циклы)
ОК 9 ОК 10	33	Базовые структуры данных (массивы, строки, списки, стеки, очереди)
ПК 5.1 ПК 5.2	34	Синтаксис и семантику языка программирования, инструментальные среды разработки
ПК 5.6 ПК 6.3	35	Методы тестирования и отладки, правила оформления программной документации
	У1	Разрабатывать алгоритмы решения задач с использованием структурного подхода
	У2	Писать, отлаживать и тестировать программы на языке высокого уровня
	У3	Использовать структуры данных и стандартные алгоритмы (поиск, сортировка)
	У4	Разрабатывать программные модули, проводить тестирование и документировать результаты

3 Соответствие оценочных средств контролируемым результатам обучения

Краткое наименование раздела (модуля) / темы дисциплины	Код результата обучения	Показатель ² овладения результатами обучения	Наименование оценочного средства и представление его в КОС	
			Текущий контроль	Промежуточная аттестация
1. Введение в ИТ				
Тема 1.1. Основы алгоритмизации	31	Способность понимать назначение и виды информационных технологий, технологий сбора, накопления, обработки, передачи и распространения информации	Устный опрос (п 5.1 вопросы 1-5) Практическая работа №1-3 (п 5.2)	Устный опрос (п 6.1 вопросы 1-15) Тест вариант №1-2 (п 6.2)
	32	Способность изучать состав, структуру, принципы реализации и функционирования информационных технологий		
	У1	Способность программировать базовые и прикладные информационные технологии		
Тема №. 1.2. Основы программирования	31	Способность понимать назначение и виды информационных технологий, технологий сбора, накопления, обработки, передачи и распространения информации	Устный опрос (п 5.1 вопросы 6-10) Практическая работа №4-7 (п 5.2)	Устный опрос (п 6.1 вопросы 1-15) Тест вариант №1-2 (п 6.2)
	32	Способность изучать состав, структуру, принципы реализации и функционирования информационных технологий		
	У1	Способность программировать базовые и прикладные информационные технологии		
Тема №. 1.3 Основы процедурного программирования	У2	Способность обрабатывать текстовую и числовую информацию	Устный опрос (п 5.1 вопросы 11-14) Практическая работа №8-12 (п 5.2)	Устный опрос (п 6.1 вопросы 1-15) Тест вариант №1-2 (п 6.2)
	У3	Способность применять мультимедийные технологии обработки и представления информации		
Тема №. 1.4 Основы объектноориентированного программирования	У2	Способность обрабатывать текстовую и числовую информацию	Устный опрос (п 5.1 вопросы 15-18) Практическая работа №13-15 (п 5.2)	Устный опрос (п 6.1 вопросы 1-15) Тест вариант №1-2 (п 6.2)
	У3	Способность применять мультимедийные технологии обработки и представления информации		

	У1	Способность программировать базовые и прикладные информационные технологии		
--	----	--	--	--

4. Описание процедуры оценивания

Результаты обучения по дисциплине, уровень сформированности компетенций оцениваются по четырёх бальной шкале оценками: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Текущая аттестация по дисциплине проводится с целью систематической проверки достижений обучающихся. Объектами оценивания являются: степень усвоения теоретических знаний, уровень овладения практическими умениями и навыками по всем видам учебной работы, качество выполнения самостоятельной работы, учебная дисциплина (активность на занятиях, своевременность выполнения различных видов заданий, посещаемость всех видов занятий по аттестуемой дисциплине).

При проведении промежуточной аттестации оценивается достижение студентом запланированных по дисциплине результатов обучения, обеспечивающих результаты освоения образовательной программы в целом. *Оценка на зачете выставляется с учетом оценок, полученных при прохождении текущей аттестации.*

Критерии оценивания устного ответа

(оценочные средства: *собеседование, устное сообщение*)

5 баллов - ответ показывает прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа; умение приводить примеры современных проблем изучаемой области.

4 балла - ответ, обнаруживающий прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа. Однако допускается одна - две неточности в ответе.

3 балла – ответ, свидетельствующий в основном о знании процессов изучаемой предметной области, отличающийся недостаточной глубиной и полнотой раскрытия темы; знанием основных вопросов теории; слабо сформированными навыками анализа явлений, процессов, недостаточным умением давать аргументированные ответы и приводить примеры; недостаточно свободным владением монологической речью, логичностью и последовательностью ответа. Допускается несколько ошибок в содержании ответа; неумение привести пример развития ситуации, провести связь с другими аспектами изучаемой области.

2 балла – ответ, обнаруживающий незнание процессов изучаемой предметной области, отличающийся неглубоким раскрытием темы; незнанием основных вопросов теории, несформированными навыками анализа явлений, процессов; неумением давать аргументированные ответы, слабым владением монологической речью, отсутствием логичности и последовательности. Допускаются серьезные ошибки в содержании ответа; незнание современной проблематики изучаемой области.

Критерии оценивания письменной работы

(оценочные средства: *конспект, тестирование*).

5 баллов - студент выразил своё мнение по сформулированной проблеме, аргументировал его, точно определив ее содержание и составляющие. Проблема раскрыта полностью, выводы обоснованы. Приведены данные отечественной и зарубежной литературы, статистические сведения, информация нормативно-правового характера. Студент владеет навыком самостоятельной работы по заданной теме;

методами и приемами анализа теоретических и/или практических аспектов изучаемой области. Фактических ошибок, связанных с пониманием проблемы, нет; графически работа оформлена правильно.

4 балла - работа характеризуется смысловой цельностью, связностью и последовательностью изложения; допущено не более 1 ошибки при объяснении смысла или содержания проблемы. Проблема раскрыта. Не все выводы сделаны и/или обоснованы. Для аргументации приводятся данные отечественных и зарубежных авторов. Продемонстрированы исследовательские умения и навыки. Фактических ошибок, связанных с пониманием проблемы, нет. Допущены одна-две ошибки в оформлении работы.

3 балла – студент проводит достаточно самостоятельный анализ основных этапов и смысловых составляющих проблемы; понимает базовые основы и теоретическое обоснование выбранной темы. Проблема раскрыта не полностью. Выводы не сделаны и/или выводы не обоснованы. Проведен анализ проблемы без привлечения дополнительной литературы. Допущено не более 2 ошибок в смысле или содержании проблемы, оформлении работы.

2 балла - работа представляет собой пересказанный или полностью переписанный исходный текст без каких бы то ни было комментариев, анализа. Не раскрыта структура и теоретическая составляющая темы. Проблема не раскрыта. Выводы отсутствуют. Допущено три или более трех ошибок в смысловом содержании раскрываемой проблемы, в оформлении работы.

Критерии оценивания тестового задания

Оценка	<i>Отлично</i>	<i>Хорошо</i>	<i>Удовлетворительно</i>	<i>Неудовлетворительно</i>
Количество правильных ответов	91 % и $\geq$	от 81% до 90,9 %	не менее 70%	менее 70%

Критерии выставления оценки студенту на зачете

(оценочные средства: *выполнение письменных разноуровневых задач и заданий.*)

Оценка по промежуточной аттестации	Характеристика качества сформированности компетенций
«зачтено» / «отлично»	Студент демонстрирует сформированность дисциплинарных компетенций на продвинутом уровне: обнаруживает всестороннее, систематическое и глубокое знание учебного материала, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.
«зачтено» / «хорошо»	Студент демонстрирует сформированность дисциплинарных компетенций на базовом уровне: основные знания, умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
«зачтено» / «удовлетворительно»	Студент демонстрирует сформированность дисциплинарных компетенций на пороговом уровне: имеет знания только основного материала, но не усвоил его деталей, в ходе контрольных мероприятий допускаются значительные ошибки, недостаточно правильные формулировки, нарушения

	логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ, при оперировании знаниями и умениями при их переносе на новые ситуации.
«не зачтено» / «неудовлетворительно»	Студент демонстрирует сформированность дисциплинарных компетенций на уровне ниже порогового: выявляется полное или практически полное отсутствие знаний значительной части программного материала, студент допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы, умения и навыки не сформированы.

5. Примеры оценочных средств для проведения текущей аттестации

5.1 Вопросы для устного опроса

- 1.Алгоритмические языки и интегрированные системы программирования. Способы трансляция.
- 2.Этапы решения задачи на ЭВМ.
- 3.Базовые алгоритмические структуры. Линейная структура. Оператор присваивания.
- 4.Примеры линейной структуры.
- 5.Базовые алгоритмические структуры. Структура ветвления. Примеры структуры ветвления.
- 6.Базовые алгоритмические структуры. Циклическая структура. Типы циклов. Цикл с параметром. Примеры
- 7.Базовые алгоритмические структуры. Циклическая структура. Типы циклов. Цикл с постусловием. Примеры
- 8.Базовые алгоритмические структуры. Циклическая структура. Типы циклов. Цикл с предусловием. Примеры
- 9.Базовые алгоритмические структуры. Циклическая структура. Типы циклов. Вложенные циклы. Примеры
10. Структура программы VBA. Структура модуля, назначение разделов.
11. Типы данных VBA. Константы. Переменные. Способы объявления. Область видимости переменной. Время жизни переменной.
12. Встроенные диалоговые окна. Окно ввода InputBox. Синтаксис. Примеры применения.
13. Встроенные диалоговые окна. Окно сообщения MsgBox. Синтаксис. Примеры применения.
14. Процедуры. Типы процедур. Разделы процедуры. Область видимости процедур.
15. Процедура – подпрограмма. Объявление. Способы объявления формальных параметров. Вызов процедуры.
16. Процедура – функция. Объявление. Способы объявления формальных параметров. Вызов процедуры.
17. Рекурсивные процедуры. Рекурсия явная и косвенная. Примеры рекурсивных процедур.
18. Инструкции управления. Условные операторы. Одностороннее и двухстороннее ветвление

5.2 Примеры практических работ

Тема 1.1: Основы алгоритмизации

Практическое занятие №1: Разработка линейных алгоритмов

Задание: Составить линейный алгоритм и программу для вычисления площади и периметра прямоугольника. (Входные данные: стороны a и b).

Решение (VBA):

```
vba
Sub PZ1_Linear()
    Dim a As Double, b As Double
    Dim S As Double, P As Double
    a = InputBox("Введите сторону a")
    b = InputBox("Введите сторону b")
    S = a * b
    P = 2 * (a + b)
    MsgBox "Площадь: " & S & ", Периметр: " & P
End Sub
```

Практическое занятие №2: Разработка разветвляющихся алгоритмов

Задание: Написать программу, которая запрашивает три числа и выводит на экран наибольшее из них.

Решение:

```
vba
Sub PZ2_Branching()
    Dim a As Double, b As Double, c As Double, max As Double
    a = InputBox("Введите число 1")
    b = InputBox("Введите число 2")
    c = InputBox("Введите число 3")
    max = a
    If b > max Then max = b
    If c > max Then max = c
    MsgBox "Максимальное число: " & max
End Sub
```

Практическое занятие №3: Разработка циклических алгоритмов

Задание: Найти сумму всех целых чисел от 1 до N (N вводится с клавиатуры).

Решение:

```
vba
Sub PZ3_Cyclic()
    Dim N As Integer, i As Integer, sum As Long
    N = InputBox("Введите N")
    sum = 0
    For i = 1 To N
        sum = sum + i
    Next i
    MsgBox "Сумма чисел от 1 до " & N & " равна " & sum
End Sub
```

Тема 1.2: Основы программирования

Практическое занятие №4: Изучение среды разработки

Задание: Открыть редактор VBA (Alt+F11). Добавить новый модуль. Написать простейшую процедуру, которая записывает в ячейку A1 листа Excel текст "Привет, мир!".

Решение:

```
vba
Sub PZ4_Environment()
```

```
Worksheets("Лист1").Range("A1").Value = "Привет, мир!"  
End Sub
```

Практическое занятие №5: Изучение приемов и инструментов отладки программы

Задание: Специально допустить ошибку (например, попытаться разделить число на текстовую переменную) и с помощью пошагового выполнения (клавиша F8) и точек останова (F9) найти и исправить ее.

Решение (Пример ошибки):

```
vba  
Sub PZ5_Debugging()  
    Dim x As Integer, y As Integer  
    x = 10  
    y = "Строка" ' Ошибка типа данных  
    MsgBox x / y ' Ошибка деления на ноль или несовпадение типов  
    ' Обратите внимание на окно Locals во время отладки.  
End Sub
```

Практическое занятие №6: Разработка линейной программы

Задание: Перевести температуру из градусов Цельсия в градусы Фаренгейта по формуле $F = C * 9 / 5 + 32$.

Решение:

```
vba  
Sub PZ6_Linear()  
    Dim C As Double, F As Double  
    C = InputBox("Введите температуру в Цельсиях")  
    F = C * 9 / 5 + 32  
    MsgBox "Температура в Фаренгейтах: " & F  
End Sub
```

Практическое занятие №7: Изучение свойств диалоговых окон

Задание: Запросить у пользователя его имя через InputBox, а затем вывести приветствие с помощью MsgBox с кнопками "Да"/"Нет". Если пользователь нажал "Да" — вывести "Отлично!", если "Нет" — "Жаль".

Решение:

```
vba  
Sub PZ7_Dialogs()  
    Dim name As String  
    Dim answer As VbMsgBoxResult  
    name = InputBox("Как вас зовут?", "Знакомство")  
    answer = MsgBox("Привет, " & name & "! Вам нравится программирование?", vbYesNo + vbQuestion)  
    If answer = vbYes Then  
        MsgBox "Отлично!"  
    Else  
        MsgBox "Жаль!"  
    End If  
End Sub
```

Тема 1.3: Основы процедурного программирования

Практическое занятие №8: Разработка процедур пользователя

Задание: Создать процедуру Sub (для вывода суммы двух чисел) и процедуру Function (для вычисления квадрата числа). Продемонстрировать их вызов.

Решение:

```
vba  
' Процедура Sub
```

```
Sub SumPrint(a As Double, b As Double)
    MsgBox "Сумма: " & (a + b)
End Sub
```

' Функция

```
Function Square(x As Double) As Double
    Square = x * x
End Function
```

' Основная процедура вызова

```
Sub PZ8_Procedures()
    Dim n As Double
    n = InputBox("Введите число")
    MsgBox "Квадрат: " & Square(n) ' Вызов функции
    Call SumPrint(n, 10) ' Вызов подпрограммы
End Sub
```

Практическое занятие №9: Разработка разветвляющейся программы

Задание: Разработать программу, которая по введенной оценке (1-5) выводит текстовую характеристику, используя оператор Select Case.

Решение:

```
vba
Sub PZ9_Branching()
    Dim grade As Integer
    grade = InputBox("Введите оценку (1-5)")
    Select Case grade
        Case 5: MsgBox "Отлично"
        Case 4: MsgBox "Хорошо"
        Case 3: MsgBox "Удовлетворительно"
        Case Else: MsgBox "Неудовлетворительно"
    End Select
End Sub
```

Практическое занятие №10: Разработка циклической программы

Задание: Написать программу для нахождения среднего арифметического вводимых пользователем чисел. Ввод останавливается, когда пользователь вводит 0. (Использовать цикл Do While).

Решение:

```
vba
Sub PZ10_Cyclic()
    Dim x As Double, sum As Double
    Dim count As Integer
    x = InputBox("Введите число (0 для выхода)")
    Do While x <> 0
        sum = sum + x
        count = count + 1
        x = InputBox("Введите число (0 для выхода)")
    Loop
    If count > 0 Then MsgBox "Среднее: " & (sum / count)
End Sub
```

Практическое занятие №11: Ввод, отладка и решение задач с массивом данных

Задание: Заполнить одномерный массив из 5 элементов случайными числами, вывести их сумму и сами числа в окно Immediate (Ctrl+G).

Решение:

```

vba
Sub PZ11_Array()
    Dim arr(1 To 5) As Integer
    Dim i As Integer, sum As Integer
    Randomize
    For i = 1 To 5
        arr(i) = Int(Rnd * 100) ' Случайное число 0-99
        Debug.Print "Элемент " & i & ": " & arr(i)
        sum = sum + arr(i)
    Next i
    Debug.Print "Сумма: " & sum
End Sub

```

Практическое занятие №12: Ввод, отладка и решение задач поиска и сортировки в массиве

Задание: Реализовать алгоритм сортировки "пузырьком" по возрастанию для массива из 10 случайных чисел.

Решение:

```

vba
Sub PZ12_Sorting()
    Dim arr(1 To 10) As Integer
    Dim i As Integer, j As Integer, temp As Integer
    Randomize
    For i = 1 To 10: arr(i) = Int(Rnd * 100): Next i
    ' Сортировка пузырьком
    For i = 1 To 9
        For j = 1 To 10 - i
            If arr(j) > arr(j + 1) Then
                temp = arr(j): arr(j) = arr(j + 1): arr(j + 1) = temp
            End If
        Next j
    Next i
    ' Вывод
    For i = 1 To 10
        Debug.Print arr(i);
    Next i
End Sub

```

Тема 1.4: Основы объектно-ориентированного программирования

Практическое занятие №13: Разработка пользовательской формы

Задание: Создать форму (UserForm). Добавить на нее текстовое поле (TextBox1), кнопку (CommandButton1) и подпись (Label1). По нажатию кнопки, текст из текстового поля должен отображаться в подписи.

Решение:

В коде формы (UserForm1):

```

vba
Private Sub CommandButton1_Click()
    Label1.Caption = "Вы ввели: " & TextBox1.Text
End Sub

```

Практическое занятие №14: Изучение свойств, методов и событий элементов управления формы

Задание: Изучить свойства (Caption, BackColor, Enabled) и события (Click, Enter, Change). Создать кнопку, которая при наведении мыши (событие MouseMove) меняет свой цвет, а при клике очищает все

поля.

Решение:

vba

```
Private Sub CommandButton1_MouseMove(ByVal Button As Integer, ByVal Shift As Integer, ByVal X As Single, ByVal Y As Single)
```

```
    CommandButton1.BackColor = vbYellow ' Событие наведения мыши  
End Sub
```

```
Private Sub CommandButton1_Click()
```

```
    TextBox1.Text = "" ' Метод очистки
```

```
    CommandButton1.BackColor = vbGreen ' Событие клика  
End Sub
```

Практическое занятие №15: Разработка формы на просмотр, редактирование и добавление данных

Задание: Создать форму для добавления данных (Имя и Возраст) в таблицу Excel. При нажатии кнопки "Добавить" данные из TextBox'ов записываются в следующие свободные строки листа, а поля очищаются.

Решение (Код формы):

```
vba  
Private Sub CommandButton1_Click()  
    Dim nextRow As Long  
    ' Ищем первую пустую строку в колонке A  
    nextRow = ThisWorkbook.Worksheets("Лист1").Cells(Rows.Count, 1).End(xlUp).Row + 1  
  
    ' Записываем данные  
    Worksheets("Лист1").Cells(nextRow, 1).Value = TextBox1.Text  
    Worksheets("Лист1").Cells(nextRow, 2).Value = TextBox2.Text  
  
    ' Очищаем поля (метод)  
    TextBox1.Text = ""  
    TextBox2.Text = ""  
  
    MsgBox "Данные успешно добавлены в строку " & nextRow  
End Sub
```

6. Оценочные средства для проведения промежуточной аттестации

6.1 Вопросы для собеседования при проведении дифференцированного зачета

Тема 1.1 Основы алгоритмизации

1. Что такое алгоритм? Перечислите основные свойства алгоритма (дискретность, массовость, конечность и др.) и раскройте суть каждого из них.
2. Опишите базовые алгоритмические структуры (линейная, ветвление, цикл). Как они изображаются на блок-схемах? Приведите примеры из жизни для каждой структуры.

Тема 1.2 Основы программирования (VBA)

3. Опишите структуру программы в VBA и структуру модуля. Из каких разделов он состоит? Каково назначение раздела Declarations?
4. Что такое переменные и константы в VBA? Перечислите основные типы данных. Какие способы объявления переменных существуют (Dim, Private, Public)? В чем разница между областью видимости и временем жизни переменной?

5. Для чего нужны встроенные диалоговые окна? Объясните синтаксис и приведите примеры практического применения InputBox и MsgBox. Как обработать ответ пользователя в MsgBox (например, нажатие «Да» или «Нет»)?
6. Что такое отладка программы? Какие инструменты отладки доступны в среде VBA (точки останова, пошаговое выполнение, окно Locals)?

Тема 1.3 Основы процедурного программирования

7. В чем принципиальная разница между процедурой-подпрограммой (Sub) и процедурой-функцией (Function)? Как правильно вызвать каждую из них?
8. Объясните механизм передачи параметров в процедуры. В чем разница между ByVal и ByRef? Приведите пример, когда изменение параметра внутри процедуры повлияет на внешнюю переменную, а когда нет.
9. Опишите инструкции управления. Что такое одностороннее и двухстороннее ветвление (If...Then...Else)? В каких случаях удобнее применять конструкцию Select Case?
10. Расскажите о циклических конструкциях VBA (For...Next, Do While...Loop, Do...Loop Until). В чем главное отличие цикла с предусловием от цикла с постусловием? Что такое вложенные циклы?
11. Что такое массивы? Дайте характеристику одномерным, многомерным и динамическим массивам. Как они объявляются и как осуществляется доступ к элементам?
12. Опишите алгоритмы поиска и сортировки массивов. Например, как работает линейный поиск, и в чем суть метода сортировки "пузырьком"?

Тема 1.4 Основы объектно-ориентированного программирования

13. Раскройте понятие объектно-ориентированного программирования применительно к VBA. Что такое объект, его свойства, методы и события? Приведите примеры.
14. Что такое пользовательская форма (UserForm)? Перечислите основные элементы управления (кнопки, текстовые поля, списки). Какие ключевые свойства и события есть у этих элементов? Как создать форму для добавления и редактирования данных?
15. Опишите объектную модель MS Excel. Перечислите иерархию основных объектов (Application, Workbook, Worksheet, Range). Как обратиться к конкретной ячейке листа? Расскажите об этапах разработки небольшого приложения (от постановки задачи и создания формы до отладки и тестирования).

6.2 Тестовые материалы

Вариант 1

Часть 1. Выбор одного правильного ответа (вопросы 1–3)

1. Какое из перечисленных свойств алгоритма означает, что процесс решения задачи разбит на отдельные простые шаги (команды)?
А) Массовость
Б) Дискретность
В) Конечность
Г) Детерминированность
2. Какой тип данных в VBA используется для хранения целых чисел от -32768 до 32767?
А) Single
Б) String
В) Integer
Г) Boolean
3. В объектно-ориентированном программировании (VBA) нажатие кнопки пользователем вызывает...
А) Метод
Б) Свойство
В) Событие
Г) Класс

Часть 2. Установление соответствия (вопросы 4–6)

4. Установите соответствие между базовой алгоритмической структурой и её характеристикой.
Структура: 1. Линейная, 2. Ветвление, 3. Цикл
Характеристика: А. Выполнение действий многократно, пока условие истинно, Б. Действия выполняются строго последовательно друг за другом, В. Выбор одного из двух вариантов действий в зависимости от условия.
(Ответ: 1-Б, 2-В, 3-А)
5. Установите соответствие между встроенным объектом/функцией VBA и его назначением.
Объект: 1. InputBox, 2. MsgBox, 3. Cells
Назначение: А. Окно для вывода сообщения пользователю, Б. Окно для ввода данных пользователем, В. Обращение к конкретной ячейке листа Excel.
(Ответ: 1-Б, 2-А, 3-В)
6. Установите соответствие между понятиями ООП и их сутью.
Понятие: 1. Объект, 2. Свойство, 3. Метод
Суть: А. Действие, которое может выполнять объект, Б. Конкретный экземпляр класса, В. Параметр, характеризующий объект (например, цвет, размер).
(Ответ: 1-Б, 2-В, 3-А)

Часть 3. Установление последовательности (вопросы 7–9)

7. Расставьте в правильной последовательности этапы решения задачи на ЭВМ:
1. Написание кода на языке программирования
2. Постановка задачи
3. Отладка и тестирование
4. Построение математической модели и выбор метода
5. Разработка алгоритма
(Ответ: 2, 4, 5, 1, 3)
8. Расставьте в правильном порядке действия при объявлении и использовании переменной в VBA:
1. Использование переменной в выражении
2. Объявление переменной с помощью оператора Dim
3. Присваивание переменной значения с помощью оператора =

(Ответ: 2, 3, 1)

9. Расставьте в правильной последовательности работу цикла с параметром (For...Next):
1. Выполнение тела цикла
 2. Установка начального значения счетчика
 3. Проверка условия достижения конечного значения счетчика
 4. Увеличение счетчика на шаг (Next)

(Ответ: 2, 3, 1, 4)

Часть 4. Открытые вопросы (вопросы 10–12)

10. Перечислите не менее трёх типов циклов в VBA. Опишите их основное отличие друг от друга.
 11. В чем заключается принципиальная разница между процедурой-подпрограммой (Sub) и процедурой-функцией (Function)? Приведите пример вызова функции.
 12. Что такое рекурсия? Опишите разницу между явной и косвенной рекурсией.
-

Вариант 2

Часть 1. Выбор одного правильного ответа (вопросы 1–3)

1. Какой способ представления алгоритма представляет собой набор условных графических обозначений (блоков), соединенных стрелками?
А) Словесный
Б) Программный
В) Графический (блок-схема)
Г) Табличный
2. Какой оператор VBA используется для организации одностороннего и двухстороннего ветвления?
А) For...Next
Б) If...Then...Else
В) Do...Loop
Г) Select...Case (хотя это тоже ветвление, If является самым базовым)
3. Как называется массив, в котором для обращения к элементу требуется два индекса?
А) Одномерный
Б) Многомерный (двумерный)
В) Динамический
Г) Строковый

Часть 2. Установление соответствия (вопросы 4–6)

4. Установите соответствие между способом представления алгоритма и его описанием.
Способ: 1. Словесный, 2. Блок-схема, 3. Программный код
Описание: А. Запись на языке программирования, Б. Описание шагов на естественном языке, В. Изображение шагов с помощью специальных геометрических фигур.
(Ответ: 1-Б, 2-В, 3-А)
5. Установите соответствие между конструкцией VBA и её назначением.
Конструкция: 1. Sub, 2. For...Next, 3. Dim
Назначение: А. Объявление переменной, Б. Объявление подпрограммы, В. Организация цикла с параметром.
(Ответ: 1-Б, 2-В, 3-А)
6. Установите соответствие между компонентами ООП и их примерами в VBA.
Компонент: 1. Класс, 2. Событие, 3. Метод
Пример: А. Click (нажатие), Б. UserForm (форма как шаблон), В. AddItem (добавление элемента в список).
(Ответ: 1-Б, 2-А, 3-В)

Часть 3. Установление последовательности (вопросы 7–9)

7. Расставьте в правильной последовательности процесс создания и отладки программы в среде

VBA:

1. Запуск программы (F5) и поиск ошибок
2. Открытие редактора Visual Basic (VBE)
3. Написание кода в окне кода модуля
4. Проверка синтаксиса и тестирование результата
(Ответ: 2, 3, 4, 1)

8. Расставьте в правильной последовательности этапы вызова подпрограммы с параметрами:

1. Выполнение кода внутри подпрограммы
2. Передача фактических параметров
3. Вызов подпрограммы (Call)
4. Возврат к месту вызова после End Sub
(Ответ: 3, 2, 1, 4)

9. Расставьте в правильном порядке этапы работы с одномерным массивом:

1. Обработка элементов (например, суммирование)
2. Описание (объявление) массива
3. Вывод полученных результатов
4. Ввод (заполнение) элементов массива
(Ответ: 2, 4, 1, 3)

Часть 4. Открытые вопросы (вопросы 10–12)

10. Опишите подробно отличие цикла с предусловием (Do While) от цикла с постусловием (Do Until). В каких случаях применяется каждый из них?
11. Что такое область видимости переменной? Объясните разницу между переменными, объявленными с помощью Dim внутри процедуры, Private в модуле и Public.
12. Как реализуется объектная модель в VBA на примере создания пользовательской формы (UserForm)? Опишите процесс добавления элементов управления и настройки их свойств.

Ответ может быть сформулирован в другой близкой по смыслу формулировке

6.3 Ключи к тестовым заданиям для проведения промежуточной аттестации

Вариант 1

Часть 1 (Выбор одного правильного ответа)

1. Б (Дискретность)
2. В (Integer)
3. В (Событие)

Часть 2 (Установление соответствия)

4. 1-Б, 2-В, 3-А (Линейная-Б, Ветвление-В, Цикл-А)
5. 1-Б, 2-А, 3-В (InputBox-Б, MsgBox-А, Cells-В)
6. 1-Б, 2-В, 3-А (Объект-Б, Свойство-В, Метод-А)

Часть 3 (Установление последовательности)

7. 2, 4, 5, 1, 3 (Постановка -> Модель -> Алгоритм -> Код -> Отладка)
8. 2, 3, 1 (Объявление Dim -> Присваивание = -> Использование)
9. 2, 3, 1, 4 (Начальное значение -> Проверка условия -> Тело цикла -> Next)

Часть 4 (Открытые вопросы — модель правильного ответа)

10. **Типы циклов в VBA:** Цикл с параметром (For...Next), цикл с предусловием (Do While...Loop / While...Wend), цикл с постусловием (Do...Loop Until). Отличие: For используется, когда количество повторений известно заранее; Do While проверяет условие перед выполнением (может не выполниться ни разу); Do Until проверяет условие после выполнения (выполнится минимум один раз).

11. **Sub и Function:** Подпрограмма (Sub) выполняет действия, но не возвращает значение. Функция (Function) выполняет действия и обязательно возвращает результат в виде значения (присваивается имени функции). Пример вызова функции: Результат = Сумма(5, 10)
12. **Рекурсия:** Это вызов процедурой или функцией самой себя. Явная рекурсия — когда функция вызывает напрямую саму себя. Косвенная рекурсия — когда функция А вызывает функцию Б, а функция Б вызывает функцию А.

Вариант 2

Часть 1 (Выбор одного правильного ответа)

1. В (Графический - блок-схема)
2. Б (If...Then...Else)
3. Б (Многомерный/двумерный)

Часть 2 (Установление соответствия)

4. 1-Б, 2-В, 3-А (Словесный-Б, Блок-схема-В, Код-А)
5. 1-Б, 2-В, 3-А (Sub-Б, For...Next-В, Dim-А)
6. 1-Б, 2-А, 3-В (Класс-Б, Событие-А, Метод-В)

Часть 3 (Установление последовательности)

7. 2, 3, 4, 1 (Открыть VBE -> Написать код -> Проверить и протестировать -> Запустить и отладить ошибки)
8. 3, 2, 1, 4 (Вызов Call -> Передача параметров -> Выполнение кода -> Возврат)
9. 2, 4, 1, 3 (Объявление -> Заполнение -> Обработка -> Вывод)

Часть 4 (Открытые вопросы — модель правильного ответа)

10. **Циклы с предусловием и постусловием:** Цикл с предусловием (Do While) сначала проверяет условие, и если оно ложно, тело цикла не выполняется ни разу. Цикл с постусловием (Do Until) сначала выполняет тело цикла, а потом проверяет условие, поэтому тело гарантированно выполнится хотя бы один раз. Применяются для работы с данными, количество которых заранее неизвестно.
11. **Область видимости переменной:** Dim внутри процедуры — локальная переменная, видна только внутри этой процедуры и уничтожается после её завершения. Private в начале модуля — доступна всем процедурам данного модуля. Public — глобальная переменная, доступна во всех модулях проекта (время жизни — до закрытия приложения).
12. **Объектная модель UserForm:** В VBA пользовательская форма (UserForm) — это класс. В конструкторе форм на неё перетаскиваются элементы управления (CommandButton, TextBox и т.д.). У каждого элемента можно настроить свойства (например, Caption, Name, BackColor) в окне свойств, а также запрограммировать методы и события (например, событие Click для кнопки).

Ответ может быть сформулирован в другой близкой по смыслу формулировке

6.4 Ключи к вопросам для собеседования при проведении дифференцированного зачета

Тема 1.1 Основы алгоритмизации

Вопрос 1. Алгоритм и его свойства.

Ответ: Алгоритм — это точное, понятное предписание, определяющее последовательность действий для решения задачи. Обязательные свойства: **Дискретность** (разбит на шаги), **Детерминированность** (однозначность, нет двойных толкований), **Конечность** (завершается за конечное число шагов), **Массовость** (применим к целому классу задач), **Результативность** (приводит к конкретному результату).

Вопрос 2. Базовые структуры и блок-схемы.

Ответ: **Линейная** (действия идут строго друг за другом), **Ветвление** (выбор одного из путей по условию: ромб с вопросом, стрелки «Да»/«Нет»), **Цикл** (многократное повторение действий). Примеры из жизни: линейная — заварить чай (насыпать, залить, размешать); ветвление — если идет дождь, взять зонт; цикл — перебирать все страницы книги, пока не найдется нужная.

Тема 1.2 Основы программирования (VBA)

Вопрос 3. Структура VBA и модуля.

Ответ: Код хранится в **модулях**. Модуль состоит из двух разделов: **Раздел деклараций** (в самом верху, где пишутся Option Explicit, глобальные переменные, константы) и **Раздел процедур** (основной код: Sub и Function). Назначение разделов: разделение настроек проекта/глобальных данных и логики выполнения.

Вопрос 4. Переменные, типы данных, область видимости.

Ответ: Переменные — ячейки памяти. Типы: Integer, Long, Single, Double, String, Boolean. Объявление: Dim (локальная, видна только внутри процедуры), Private (видна в пределах модуля), Public (глобальная, видна во всем проекте). Время жизни: локальные живут, пока работает процедура; глобальные — до закрытия приложения. Константы объявляются через Const.

Вопрос 5. Диалоговые окна InputBox и MsgBox.

Ответ: **InputBox** — окно ввода. Синтаксис: InputBox("Подсказка", "Заголовок", "Значение по умолчанию"). Возвращает строку. **MsgBox** — окно вывода/сообщения. Синтаксис: MsgBox("Текст", vbYesNo + vbQuestion, "Заголовок"). Возвращает число (код нажатой кнопки), например, vbYes или vbNo, чтобы программа могла среагировать на выбор пользователя.

Вопрос 6. Отладка программы.

Ответ: Отладка — поиск и исправление ошибок. Инструменты: **Точка останова (F9)** — останавливает код в нужном месте; **Пошаговое выполнение (F8)** — построчное выполнение; **Окно Locals / Watch** — просмотр текущих значений переменных; **Окно Immediate (Ctrl+G)** — моментальное выполнение команд без запуска всей программы.

Тема 1.3 Основы процедурного программирования

Вопрос 7. Разница Sub и Function.

Ответ: **Sub** (подпрограмма) — выполняет действие, но **не возвращает** значение. Вызывается командой Call Имя или просто по имени. **Function** (функция) — выполняет действие и **обязательно возвращает** результат (присваивается имени функции). Вызывается присваиванием: Переменная = ИмяФункции(аргументы).

Вопрос 8. ByVal и ByRef.

Ответ: **ByVal** — передача **по значению**: в процедуру уходит копия переменной. Изменения внутри процедуры не повлияют на исходную переменную. **ByRef** (по умолчанию) — передача **по ссылке**: передается адрес памяти, и процедура может изменить исходную переменную снаружи.

Вопрос 9. Условные операторы (ветвление).

Ответ: **Одностороннее** ветвление (If Условие Then Действие) — если условие истинно, делаем действие. **Двухстороннее** (If Условие Then Действие1 Else Действие2) — если да, делаем одно, иначе — другое. Select Case удобнее применять, когда нужно проверить переменную на множество конкретных значений (например, оценка 5, 4, 3, 2), чтобы не писать длинную цепочку ElseIf.

Вопрос 10. Циклы.

Ответ: **For...Next** — цикл с параметром, количество повторений известно заранее. **Do While...Loop** — цикл с предусловием: проверяет условие **до** выполнения, может не выполниться ни разу. **Do...Loop Until** — цикл с постусловием: проверяет условие **после** выполнения, тело цикла выполнится минимум один раз. **Вложенные циклы** — когда один цикл находится внутри другого (например, таблица: строки x столбцы).

Вопрос 11. Массивы.

Ответ: Массив — набор однотипных данных. **Одномерный** (Dim arr(1 To 10) As Integer) — список. **Многомерный** (Dim arr(1 To 5, 1 To 5)) — таблица (матрица). **Динамический** (Dim arr()) — размер объявляется позже через ReDim Preserve, который позволяет менять размер, сохраняя данные.

Вопрос 12. Поиск и сортировка.

Ответ: **Линейный поиск** — перебор всех элементов массива по порядку до нахождения нужного. **Сортировка пузырьком** — попарное сравнение соседних элементов и их обмен местами (большие значения "всплывают" вверх). Каждый проход по массиву ставит один максимальный элемент на свое место.

Тема 1.4 Основы объектно-ориентированного программирования

Вопрос 13. ООП в VBA.

Ответ: Это подход, где программа строится из объектов. **Объект** (например, Кнопка, Форма, Лист Excel). **Свойства** (цвет, название, размер). **Методы** (действия, которые объект умеет делать, например, Clear для текстового поля). **События** (реакции на действия пользователя, например, Click — клик мышью).

Вопрос 14. Пользовательская форма (UserForm).

Ответ: Для создания формы в VBE выбираем Insert -> UserForm. Элементы управления: TextBox (поле ввода), Label (подпись), CommandButton (кнопка), ComboBox (выпадающий список). **Свойства** (например, Name, Caption) задаются в окне Properties. **События** (например, Click для кнопки) программируются в окне кода. Разработка формы позволяет создавать удобный интерфейс для просмотра, добавления и редактирования данных без ввода в ячейки напрямую.

Вопрос 15. Объектная модель Excel и этапы разработки.

Ответ: Иерархия: Application (сам Excel) -> Workbook (книга) -> Worksheet (лист) -> Range (ячейка). Обращение: Worksheets("Лист1").Range("A1").Value = "Текст".

Этапы разработки приложения: 1) Постановка задачи (что делаем). 2) Построение алгоритма (как делаем). 3) Написание кода (создание формы и процедур). 4) Отладка (поиск ошибок). 5) Тестирование на реальных данных. 6) Сопровождение и исправление недочетов.

Все ответы могут быть сформулированы в другой близкой по смыслу формулировке

6.5 Ключи на вопросы для устного опроса

1. Алгоритмические языки и интегрированные системы программирования. Способы трансляции.

Алгоритмические языки — это формализованные языки для описания алгоритмов и данных (например, Pascal, C++, Python, VBA).

Интегрированные системы программирования (IDE) — это программные комплексы (например,

Visual Studio, Visual Basic Editor для VBA), включающие редактор кода, отладчик, компилятор/интерпретатор и средства автоматизации.

Способы трансляции (перевода языка программирования в машинный код):

- **Компиляция** — полный перевод всей программы в машинный код до её запуска (создается .exe файл).
- **Интерпретация** — построчный перевод и выполнение кода во время запуска (VBA работает именно так).
- **Гибридная (трансляция в промежуточный байт-код)** — компиляция в промежуточный код, который затем интерпретируется виртуальной машиной (Java, C#).

2. Этапы решения задачи на ЭВМ

1. **Постановка задачи** (формулировка цели, условий и ограничений).
2. **Построение математической модели** (описание задачи на языке формул).
3. **Выбор метода решения и разработка алгоритма** (порядок действий, блок-схема).
4. **Запись алгоритма на языке программирования** (написание кода).
5. **Отладка и тестирование** (поиск и исправление ошибок).
6. **Анализ результатов и сопровождение программы** (проверка корректности вывода, документация).

3. Базовые алгоритмические структуры. Линейная структура. Оператор присваивания

Линейная структура — алгоритм, в котором все действия выполняются строго последовательно, одно за другим (сверху вниз).

Оператор присваивания — команда, записывающая значение в переменную. В VBA обозначается знаком =.

Синтаксис: Переменная = Выражение.

Пример: $X = 5$, $Y = X + 3$.

4. Примеры линейной структуры

- Вычислить площадь прямоугольника: Запросить стороны A и B $\rightarrow S = A * B \rightarrow$ Вывести S.
- Перевод метров в километры: $M \rightarrow Km = M / 1000 \rightarrow$ Вывести Km.
- Среднее арифметическое трёх чисел.

5. Базовые алгоритмические структуры. Структура ветвления. Примеры структуры ветвления

Структура ветвления (условие) — алгоритм, в котором в зависимости от истинности условия выполняется одна или другая последовательность действий. Используется If...Then...Else или Select Case.

Примеры:

- Если число положительное, вывести "Плюс", иначе "Минус".
- Если температура выше 30 градусов, включить кондиционер.

6. Базовые алгоритмические структуры. Циклическая структура. Типы циклов. Цикл с параметром. Примеры

Циклы — многократное повторение блока команд. Типы: цикл с параметром (счетчиком), цикл с предусловием, цикл с постусловием.

Цикл с параметром (For...Next) — количество повторений известно заранее.

Пример на VBA: Вывести числа от 1 до 10.

```
vba
For i = 1 To 10
    Debug.Print i
Next i
```

7. Базовые алгоритмические структуры. Циклическая структура. Типы циклов. Цикл с постусловием. Примеры

Цикл с постусловием (Do...Loop Until / While) — тело цикла выполняется как минимум один раз, условие проверяется **после** выполнения тела цикла.

Пример на VBA: Запрашивать у пользователя пароль, пока он не введет верный.

```
vba
```

```
Do
    pass = InputBox("Введите пароль")
Loop Until pass = "123"
```

8. Базовые алгоритмические структуры. Циклическая структура. Типы циклов. Цикл с предусловием. Примеры

Цикл с предусловием (While...Wend или Do While...Loop) — условие проверяется до выполнения тела цикла. Если условие изначально ложно, тело цикла не выполнится ни разу.

Пример на VBA: Суммировать вводимые числа, пока не будет введен 0.

```
vba
Sum = 0
x = InputBox("Введите число")
Do While x <> 0
    Sum = Sum + x
    x = InputBox("Введите число")
Loop
```

9. Базовые алгоритмические структуры. Циклическая структура. Типы циклов. Вложенные циклы. Примеры

Вложенные циклы — один цикл находится внутри другого. Внутренний цикл полностью выполняется при каждом проходе внешнего.

Пример на VBA: Вывести таблицу умножения (матрицу).

```
vba
For i = 1 To 3
    For j = 1 To 3
        Cells(i, j).Value = i * j
    Next j
Next i
```

10. Структура программы VBA. Структура модуля, назначение разделов

Код в VBA хранится в **модулях** (Module1, Module2 и т.д.).

- **Раздел деклараций (Declarations):** находится в самом верху модуля. Содержит опции (Option Explicit), глобальные переменные и константы (Public).
- **Раздел процедур:** здесь располагаются основные блоки кода — процедуры Sub и функции Function.
- **Назначение разделов:** Разделение кода на логические блоки, ограничение области видимости переменных, удобство отладки и повторного использования кода.

11. Типы данных VBA. Константы. Переменные. Способы объявления. Область видимости переменной. Время жизни переменной

- **Типы данных:** Integer (целые), Long (длинные целые), Single/Double (вещественные), String (строки), Boolean (Логический Истина/Ложь), Date (дата), Variant (универсальный тип).
- **Константы:** Const Pi As Double = 3.14
- **Переменные:** Dim x As Integer.
- **Способы объявления:**
 - Dim (локальная переменная внутри процедуры),
 - Private (на уровне модуля),
 - Public (глобальная, доступна во всех модулях).
- **Область видимости:** где переменную можно использовать (процедура, модуль, вся программа).
- **Время жизни:**
 - Локальные (Dim) — живут, пока выполняется процедура, затем исчезают.
 - Static — сохраняют значение между вызовами процедуры.
 - Глобальные — живут, пока открыта программа/документ.

12. Встроенные диалоговые окна. Окно ввода InputBox. Синтаксис. Примеры применения

InputBox — стандартное окно VBA для ввода данных пользователем.

Синтаксис: InputBox(Prompt, [Title], [Default], [Xpos], [Ypos])

- Prompt — текст-подсказка (обязательно).
- Title — заголовок окна.
- Default — значение по умолчанию в поле ввода.

Пример:

```
vba
```

```
Dim name As String
```

```
name = InputBox("Введите ваше имя:", "Регистрация", "Гость")
```

13. Встроенные диалоговые окна. Окно сообщения MsgBox. Синтаксис. Примеры применения

MsgBox — окно для вывода информации или получения ответа (Да/Нет/ОК).

Синтаксис: MsgBox(Prompt, [Buttons], [Title])

- Buttons — определяет, какие кнопки и иконки показать (например, vbYesNo, vbInformation, vbCritical).

Пример:

```
vba
```

```
Dim answer As Integer
```

```
answer = MsgBox("Продолжить выполнение?", vbYesNo + vbQuestion, "Вопрос")
```

```
If answer = vbYes Then
```

```
    ' Действие
```

```
End If
```

14. Процедуры. Типы процедур. Разделы процедуры. Область видимости процедур

Типы процедур:

1. **Sub** (подпрограмма) — выполняет действия, но **не возвращает** значение.
2. **Function** (функция) — выполняет действия и **возвращает** результат.

Разделы процедуры: Заголовок (имя, параметры), тело (код), конец (End Sub / End Function).

Область видимости:

- Public (по умолчанию) — доступна из всех модулей.
- Private — доступна только внутри текущего модуля.

15. Процедура – подпрограмма. Объявление. Способы объявления формальных параметров. Вызов процедуры

Объявление: Sub ИмяПроцедуры (Параметр1 As Тип, Параметр2 As Тип) ... End Sub.

Способы объявления формальных параметров (передачи данных):

- ByVal (по значению) — в процедуру передается копия значения. Изменения внутри процедуры **не влияют** на исходную переменную.
- ByRef (по ссылке, по умолчанию) — передается адрес ячейки памяти. Процедура может **изменить** исходное значение переменной.

Вызов: Call ИмяПроцедуры(аргумент1, аргумент2) или просто ИмяПроцедуры аргумент1, аргумент2.

16. Процедура – функция. Объявление. Способы объявления формальных параметров. Вызов процедуры

Объявление: Function ИмяФункции (Параметры) As ТипВозвращаемогоЗначения ... End Function.

Внутри функции обязательно нужно присвоить результат **имени функции:** ИмяФункции = Результат.

Способы объявления параметров: Аналогично Sub (ByVal и ByRef).

Вызов: Результат присваивается переменной: Переменная = ИмяФункции(аргументы). Например: y = Sum(2, 3).

17. Рекурсивные процедуры. Рекурсия явная и косвенная. Примеры рекурсивных процедур

Рекурсия — вызов процедуры или функции самой себя.

- **Явная рекурсия** — функция вызывает саму себя напрямую.
- **Косвенная рекурсия** — функция А вызывает функцию Б, которая вызывает функцию А.

Пример явной рекурсии (факториал):

```
vba
Function Fact(n As Integer) As Long
    If n <= 1 Then
        Fact = 1 ' Базовый случай (условие выхода)
    Else
        Fact = n * Fact(n - 1) ' Рекурсивный вызов
    End If
End Function
```

18. Инструкции управления. Условные операторы. Одностороннее и двухстороннее ветвление

- **Одностороннее ветвление** (без Else): действие выполняется только при истинности условия, иначе пропускается.

```
vba
If x > 0 Then Debug.Print "Число положительное"
```

- **Двухстороннее ветвление** (с Else): при истинности выполняется одна ветка, при ложности — другая.

```
vba
If x > 0 Then
    Debug.Print "Положительное"
Else
    Debug.Print "Отрицательное или ноль"
End If
```

(Для множественного ветвления используется конструкция Select Case).

Все ответы могут быть сформулированы в другой близкой по смыслу формулировке

6.6 Экзаменационные билеты

Билеты к дифференцированному зачету

Билет №1

1. Понятие алгоритма. Основные свойства алгоритмов (дискретность, массовость, конечность, детерминированность).
2. Способы представления алгоритмов (словесный, графический, программный).
Практическое задание: Составить блок-схему и написать код линейного алгоритма для вычисления суммы двух чисел.

Билет №2

1. Линейная алгоритмическая структура. Оператор присваивания в VBA.
2. Понятие переменной. Типы данных в VBA (Integer, Double, String, Boolean).
Практическое задание: Написать программу для вычисления площади и периметра прямоугольника.

Билет №3

1. Разветвляющаяся структура. Условный оператор If...Then...Else.
2. Одностороннее и двухстороннее ветвление.
Практическое задание: Написать программу, определяющую, является ли введенное число положительным, отрицательным или нулем.

Билет №4

1. Вложенные условия. Множественный выбор Select Case.
2. Структура модуля в VBA. Назначение разделов.
Практическое задание: Написать программу, которая по введенной оценке (1-5) выводит её текстовую характеристику («отлично», «хорошо» и т.д.) с помощью Select Case.

Билет №5

1. Циклическая структура. Цикл с параметром For...Next.

2. Понятие счетчика цикла. Шаг цикла (Step).

Практическое задание: Вывести на экран таблицу умножения для числа 7.

Билет №6

1. Цикл с предусловием Do While...Loop.
2. Отличие цикла с предусловием от цикла с постусловием.

Практическое задание: Написать программу, которая суммирует вводимые числа до тех пор, пока не будет введен 0.

Билет №7

1. Цикл с постусловием Do...Loop Until.
2. Правила использования циклов при обработке данных.

Практическое задание: Написать программу, которая запрашивает пароль, пока пользователь не введет правильный (например, "123").

Билет №8

1. Вложенные циклы.
2. Понятие сложности алгоритма на примере вложенных циклов.

Практическое задание: Вывести на экран матрицу (таблицу) 3x3, заполненную числами от 1 до 9.

Билет №9

1. Процедуры-подпрограммы (Sub). Объявление, вызов.
2. Передача параметров по значению (ByVal) и по ссылке (ByRef).

Практическое задание: Создать подпрограмму Sub, которая принимает два числа и выводит их сумму через MsgBox.

Билет №10

1. Процедуры-функции (Function). Отличие от Sub.
2. Способы возврата значения из функции (присваивание имени функции).

Практическое задание: Создать функцию для вычисления квадрата введенного числа.

Билет №11

1. Рекурсия. Понятие явной и косвенной рекурсии.
2. Базовый случай в рекурсии. Условие выхода.

Практическое задание: Написать рекурсивную функцию для вычисления факториала числа N.

Билет №12

1. Область видимости переменных (Dim, Private, Public).
2. Время жизни переменной (локальные и статические переменные).

Практическое задание: Написать процедуру, демонстрирующую область видимости переменной (локальная внутри одной процедуры).

Билет №13

1. Встроенные диалоговые окна VBA: InputBox. Синтаксис и назначение.
2. Встроенные диалоговые окна VBA: MsgBox. Синтаксис и обработка нажатых кнопок (vbYesNo).

Практическое задание: Запросить у пользователя имя, а затем вывести сообщение «Привет, [Имя]!» с кнопками «Да» и «Нет».

Билет №14

1. Одномерные массивы. Понятие, объявление и заполнение.
2. Индексация элементов массива. Обращение к элементам.

Практическое задание: Создать массив из 5 случайных чисел и вычислить их сумму.

Билет №15

1. Двумерные массивы (матрицы). Объявление и особенности.
2. Ввод и вывод двумерного массива.

Практическое задание: Заполнить матрицу 2x2 с клавиатуры и вывести её на экран.

Билет №16

1. Динамические массивы. Оператор ReDim и Preserve.

2. Зачем нужны динамические массивы? Отличие от статических.

Практическое задание: Запросить у пользователя размер массива, создать его динамически и заполнить введенными данными.

Билет №17

1. Алгоритмы поиска. Линейный поиск элемента в массиве.
2. Нахождение максимального (минимального) элемента.

Практическое задание: Написать программу для поиска максимального элемента в массиве из 10 случайных чисел.

Билет №18

1. Алгоритмы сортировки. Метод «пузырька».
2. Пошаговый принцип работы сортировки пузырьком.

Практическое задание: Отсортировать массив из 5 чисел по возрастанию.

Билет №19

1. Основные понятия ООП в VBA: объект, свойства, методы, события.
2. Примеры объектов в VBA (UserForm, CommandButton, TextBox).

Практическое задание: Создать форму UserForm с кнопкой, при нажатии на которую в текстовое поле выводится слово «Привет».

Билет №20

1. Пользовательская форма (UserForm). Элементы управления: Label, TextBox, CommandButton.
2. Свойства элементов управления (Caption, Name, Text, Visible).

Практическое задание: Создать форму, где по нажатию кнопки текст из TextBox отображается в Label.

Билет №21

1. События элементов управления (Click, Change, Enter, MouseMove).
2. Написание обработчиков событий в коде VBA.

Практическое задание: Создать кнопку, которая меняет свой цвет при наведении на нее мыши (событие MouseMove) и возвращает цвет при клике.

Билет №22

1. Введение в дисциплину. История развития языков программирования.
2. Понятие интегрированной среды разработки (IDE).

Практическое задание: Открыть редактор VBA, вставить новый модуль и написать простейшую процедуру записи текста в ячейку Excel.

Билет №23

1. Строковые функции VBA: Len, Left, Right, Mid. Назначение.
2. Оператор конкатенации строк (&).

Практическое задание: Написать программу, которая выводит первые 3 символа введенной строки.

Билет №24

1. Математические функции VBA (Abs, Int, Rnd, Sqr). Назначение.
2. Операторы арифметики в VBA (+, -, *, /, %, Mod, ^).

Практическое задание: Вычислить среднее арифметическое трех введенных чисел.

Билет №25

1. Основные принципы структурного программирования.
2. Граф программы, вершины и пути графа.

Практическое задание: Составить блок-схему и код программы для вычисления суммы чисел от 1 до N (с помощью цикла).

Билет №26

1. Понятие константы в VBA. Объявление констант (Const).
2. Понятие стандартных процедур и функций.

Практическое задание: Объявить константу числа Пи и использовать ее для вычисления площади круга.

Билет №27

1. Понятие «массив данных». Зачем нужны массивы?
2. Классификация массивов (одномерные, многомерные, динамические).

Практическое задание: Создать массив из 4 имен и вывести их в окно сообщения по одному.

Билет №28

1. Организация работы с базой данных в Excel через VBA (обращение к ячейкам: Cells, Range).
2. Понятие «Объект Excel» (Workbook, Worksheet, Range).

Практическое задание: Написать процедуру, которая записывает в ячейку A1 число 10, а в A2 сумму 10 и 5.

Билет №29

1. Обработка исключительных ситуаций. Оператор On Error GoTo.
2. Методы отладки программ (точки останова, пошаговое выполнение).

Практическое задание: Написать программу с проверкой ввода числа (защита от ввода букв через Val и On Error).

Билет №30

1. Комплексное задание: Понятие пользовательского интерфейса.
2. Требования к хорошему интерфейсу пользователя.

Практическое задание: Разработать форму для добавления данных (Имя и Возраст) в список (ListBox) с кнопкой «Добавить» и «Очистить».

6.7 Ключи к экзаменационным билетам

Билет №1

- **T1:** Алгоритм — это последовательность действий, ведущая к решению задачи. Свойства: дискретность, массовость, конечность, детерминированность.
- **T2:** Словесный (описание на естественном языке), графический (блок-схема), программный (код).
- **Практика:**

vba

```
Sub Sum()
```

```
Dim a, b, s
```

```
a = Val(InputBox("Введите a"))
```

```
b = Val(InputBox("Введите b"))
```

```
s = a + b
```

```
MsgBox "Сумма: " & s
```

```
End Sub
```

Билет №2

- **T1:** Линейный — выполняется последовательно, оператор присваивания: $x = 10$.
- **T2:** Integer, Long, Single, Double, String, Boolean.
- **Практика:** $S = a * b$, $P = 2 * (a + b)$.

Билет №3

- **T1:** Ветвление — если условие истинно, выполняется одна ветка, иначе другая.
- **T2:** Одностороннее (без Else, просто пропускает действие), двухстороннее (с Else).
- **Практика:**

vba

```
If x > 0 Then MsgBox "Положительное" ElseIf x < 0 Then MsgBox "Отрицательное" Else MsgBox "Ноль"
```

Билет №4

- **T1:** Select Case — для выбора из нескольких вариантов. Вложенные If — для сложных условий.
- **T2:** Верхняя часть модуля — Declarations (объявления), далее — процедуры.
- **Практика:** Select Case grade: Case 5: MsgBox "Отлично" ... End Select

Билет №5

- **T1:** For i = 1 To N — цикл с известным количеством повторений.
- **T2:** Счетчик — переменная i, шаг Step (например, Step 2).
- **Практика:** For i = 1 To 10: Debug.Print 7 & " x " & i & " = " & 7 * i: Next i

Билет №6

- **T1:** Do While ... Loop — проверяет условие перед телом цикла.
- **T2:** Предусловие проверяется до, а постусловие — после. При While тело может не выполниться ни разу.
- **Практика:** Do While x <> 0: sum = sum + x: x = InputBox("Введите число"): Loop

Билет №7

- **T1:** Do ... Loop Until — тело цикла выполнится минимум один раз.
- **T2:** Осторожность с бесконечными циклами (наличие условия выхода).
- **Практика:** Do: pass = InputBox("Пароль"): Loop Until pass = "123"

Билет №8

- **T1:** Вложенные — один цикл внутри другого.
- **T2:** Сложность $O(n^2)$.
- **Практика:** For i = 1 To 3: For j = 1 To 3: Cells(i, j).Value = i * j: Next j: Next i

Билет №9

- **T1:** Sub Имя(параметры) ... End Sub. Вызов: Call Имя или просто Имя.
- **T2:** ByVal — передается копия (изменения не влияют на оригинал). ByVal — передается ссылка (изменяется оригинал).
- **Практика:** Sub SumPrint(a, b): MsgBox a + b: End Sub

Билет №10

- **T1:** Function возвращает результат, Sub просто выполняет действия.
- **T2:** Результат присваивается имени функции внутри тела: Square = x * x.
- **Практика:** Function Square(x) As Double: Square = x * x: End Function

Билет №11

- **T1:** Рекурсия — вызов функции самой себя. Явная — вызывает саму себя напрямую. Косвенная — А вызывает Б, Б вызывает А.
- **T2:** Базовый случай (условие выхода), без него — бесконечная рекурсия.
- **Практика:** Function Fact(n) If n <= 1 Then Fact = 1 Else Fact = n * Fact(n - 1)

Билет №12

- **T1:** Dim — локальная (внутри процедуры), Private — в модуле, Public — во всех модулях.
- **T2:** Локальные живут, пока работает процедура. Static сохраняет значение между вызовами.
- **Практика:** Sub Test() Dim x As Integer: x = 10: End Sub (x исчезнет).

Билет №13

- **T1:** InputBox — окно ввода, возвращает строку.
- **T2:** MsgBox — окно вывода, может возвращать код нажатой кнопки (vbYes, vbNo).
- **Практика:** name = InputBox("Имя"): answer = MsgBox("Привет, " & name, vbYesNo): If answer = vbYes Then ...

Билет №14

- **T1:** Массив — набор однотипных данных с общим именем.
- **T2:** Нумерация элементов (индексы) Dim arr(1 To 5).
- **Практика:**

vba

Dim arr(1 To 5) As Integer

Dim sum As Integer: sum = 0

Randomize

For i = 1 To 5: arr(i) = Int(Rnd * 100): sum = sum + arr(i): Next i

MsgBox sum

Билет №15

- **T1:** Двумерный — Dim arr(1 To 2, 1 To 2).
- **T2:** Двойные циклы ввода/вывода.
- **Практика:** For i = 1 To 2: For j = 1 To 2: arr(i,j) = InputBox(...): Next j: Next i

Билет №16

- **T1:** Dim arr() As Integer + ReDim arr(n).
- **T2:** Динамический можно менять размер, статический — нет. Preserve сохраняет содержимое.
- **Практика:** n = Val(InputBox("Размер")): ReDim arr(n)

Билет №17

- **T1:** Линейный поиск — перебор всех элементов по порядку до совпадения.
- **T2:** max = arr(1), затем сравнение с остальными.
- **Практика:** max = arr(1): For i = 2 To 10: If arr(i) > max Then max = arr(i): Next i

Билет №18

- **T1:** «Пузырек» — попарное сравнение соседних элементов и их обмен местами.
- **T2:** Большие элементы «всплывают» вверх за каждый проход.
- **Практика:** For i = 1 To 4: For j = 1 To 5 - i: If arr(j) > arr(j+1) Then temp = arr(j): arr(j) = arr(j+1): arr(j+1) = temp: Next j: Next i

Билет №19

- **T1:** Объект (например, Кнопка), Свойства (Caption), Методы (AddItem), События (Click).
- **T2:** UserForm (форма), CommandButton (кнопка), TextBox (поле ввода).
- **Практика:** Private Sub CommandButton1_Click() TextBox1.Text = "Привет" End Sub

Билет №20

- **T1:** Label (подпись), TextBox (ввод), CommandButton (кнопка).
- **T2:** Caption (текст надписи), Name (имя объекта), Text (содержимое), Visible (видимость).
- **Практика:** Private Sub CommandButton1_Click() Label1.Caption = TextBox1.Text End Sub

Билет №21

- **T1:** Click (клик), Change (изменение текста), Enter (вход в поле), MouseMove (наведение мыши).
- **T2:** Обработчики событий пишутся в окне кода UserForm.
- **Практика:**

vba

```
Private Sub CommandButton1_MouseMove(...)
```

```
    CommandButton1.BackColor = vbYellow
```

```
End Sub
```

Билет №22

- **T1:** История от машинных кодов до объектно-ориентированных языков.
- **T2:** IDE (например, Visual Studio, VBE) — редактор кода, отладчик, компилятор.
- **Практика:** Alt+F11, Insert -> Module, Sub Test(): Range("A1").Value = "Привет": End Sub

Билет №23

- **T1:** Len (длина), Left (слева), Right (справа), Mid (с середины).
- **T2:** & — склеивание строк.
- **Практика:** Left(str, 3)

Билет №24

- **T1:** Abs (модуль), Int (целая часть), Rnd (случайное число), Sqr (корень).
- **T2:** +, -, *, /, \ (целочисленное), Mod (остаток), ^ (степень).
- **Практика:** (a + b + c) / 3

Билет №25

- **T1:** Разбиение на модули, отказ от безусловных переходов, использование циклов и ветвлений.
- **T2:** Граф — вершины (операторы), дуги (переходы).
- **Практика:** Sum = 0: For i = 1 To N: Sum = Sum + i: Next: MsgBox Sum

Билет №26

- **T1:** Const Pi = 3.14 — значение не меняется.

- **T2:** Встроенные функции, например, MsgBox, InputBox, математические функции.
- **Практика:** Const Pi As Double = 3.14: S = Pi * r ^ 2

Билет №27

- **T1:** Массив — это структура для хранения нескольких однотипных значений под одним именем.
- **T2:** Одномерные (векторы), многомерные (матрицы), динамические (изменяемый размер).
- **Практика:** Dim names(1 To 4) As String: For i = 1 To 4: MsgBox names(i): Next i

Билет №28

- **T1:** Cells(row, col) и Range("A1") — обращение к ячейкам.
- **T2:** Workbook (книга), Worksheet (лист), Range (ячейки).
- **Практика:** Range("A1").Value = 10: Range("A2").Value = Range("A1").Value + 5

Билет №29

- **T1:** On Error GoTo ErrHandler — перехват ошибок.
- **T2:** Точки останова (F9), пошаговое выполнение (F8), окно Locals.
- **Практика:**

vba

On Error GoTo ErrHandler

x = Val(InputBox("Число"))

Exit Sub

ErrHandler:

MsgBox "Ошибка ввода!"

End Sub

Билет №30

- **T1:** Пользовательский интерфейс — способ взаимодействия пользователя и системы.
- **T2:** Интуитивность, удобство, понятность, стандартизация.
- **Практика:** ListBox1.AddItem TextBox1.Text (для кнопки «Добавить»), ListBox1.Clear (для кнопки «Очистить»).