

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Погребная Ярослава Адольфовна
Должность: Директор
Дата подписания: 27.01.2025 14:32:12
Уникальный программный ключ:
df3b41101d3b2b77a07bf7ecfceb4c437367e6f2

**Частное образовательное учреждение
профессионального образования
«Налоговый колледж»**

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
учебной дисциплины**

ОП.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

Специальность 09.02.07 Информационные системы и программирование

Квалификация выпускника: программист

Образовательная программа на базе среднего общего образования
Образовательная программа на базе основного общего образования

Формы обучения: очная

Фонд оценочных средств учебной дисциплины составлен на основании Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование, утвержденного приказом Минобрнауки России от 09.12.2016 № 1547 (с изменениями и дополнениями)

Фонд оценочных средств обсужден на заседании ПЦК общепрофессиональных и профессиональных дисциплин

Протокол № 5 от 24.05.2024

СОГЛАСОВАНО
Заместитель директора по УМР
Ложникова Т.В.
24.05.2024

СОДЕРЖАНИЕ

1.	ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ	4
2.	РЕЗУЛЬТАТЫ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ	5
3.	ОЦЕНКА ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	10
3.1	Контроль и оценка освоения учебной дисциплины по темам	10
3.2	Оценочные средства для текущего контроля	11
3.3	Оценочные средства для промежуточной аттестации	14
4.	КРИТЕРИИ ОЦЕНКИ	17

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств (ФОС) является неотъемлемой частью рабочей программы учебной дисциплины ОП.04 Основы алгоритмизации и программирования и предназначен для контроля и оценки образовательных достижений, обучающихся (в т.ч. по самостоятельной работе студентов, далее – СРС), освоивших программу данной дисциплины.

Целью фонда оценочных средств является установление соответствия уровня подготовки обучающихся требованиям ФГОС СПО по специальности 09.02.07 Информационные системы и программирование (квалификация - программист).

Рабочей программой учебной дисциплины предусмотрено формирование следующих компетенций:

- 1) ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам;
- 2) ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности;
- 3) ОК 04. Эффективно взаимодействовать и работать в коллективе и команде;
- 4) ОК 05. Эффективно взаимодействовать и работать в коллективе и команде;
- 5) ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках;
- 6) ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием;
- 7) ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием;
- 8) ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств;
- 9) ПК 1.4. Выполнять тестирование программных модулей;
- 10) ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода;
- 11) ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения;
- 12) ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

Формой промежуточной аттестации по учебной дисциплине является экзамен.

2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ

В результате аттестации по учебной дисциплине осуществляется комплексная проверка следующих знаний, умений, практического опыта, а также динамика формирования общих и профессиональных компетенций.

Результаты обучения: знания, умения, практический опыт	Формируемые компетенции
Знать:	ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
31 – актуальный профессиональный и социальный контекст, в котором приходится работать и жить;	
32 – основные источники информации и ресурсы для решения задач и проблем в профессиональном и/или социальном контексте;	
33 – алгоритмы выполнения работ в профессиональной и смежных областях;	
34 – методы работы в профессиональной и смежных сферах;	
35 – структуру плана для решения задач;	
36 – порядок оценки результатов решения задач профессиональной деятельности.	
Уметь:	
У1 – распознавать задачу и/или проблему в профессиональном и/или социальном контексте;	
У2 – анализировать задачу и/или проблему и выделять её составные части;	
У3 – определять этапы решения задачи;	
У4 – выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы;	
У5 – составлять план действия;	
У6 – определять необходимые ресурсы;	
У7 – владеть актуальными методами работы в профессиональной и смежных сферах;	
У8 – реализовывать составленный план;	
У9 – оценивать результат и последствия своих действий (самостоятельно или с помощью наставника).	
Знать:	ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности
31 – номенклатуру информационных источников, применяемых в профессиональной деятельности;	
32 – приемы структурирования информации;	
33 – формат оформления результатов поиска информации, современные средства и устройства информатизации;	
34 – порядок их применения и программное обеспечение в профессиональной деятельности в том числе с использованием цифровых средств.	
Уметь:	
У1 – определять задачи для поиска информации;	
У2 – определять необходимые источники информации;	
У3 – планировать процесс поиска;	
У4 – структурировать получаемую информацию;	

У5 – выделять наиболее значимое в перечне информации;	
У6 – оценивать практическую значимость результатов поиска;	
У7 – оформлять результаты поиска, применять средства информационных технологий для решения профессиональных задач;	
У8 – использовать современное программное обеспечение;	
У9 – использовать различные цифровые средства для решения профессиональных задач.	
Знать:	
З1 – психологические основы деятельности коллектива, психологические особенности личности;	
З2 – основы проектной деятельности.	
Уметь:	
У1 – организовывать работу коллектива и команды;	
У2 – взаимодействовать с коллегами, руководством, клиентами в ходе профессиональной деятельности.	
Знать:	
З1 – особенности социального и культурного контекста;	
З2 – правила оформления документов и построения устных сообщений.	
Уметь:	
У1 – грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке, проявлять толерантность в рабочем коллективе.	
Знать:	
З1 – правила построения простых и сложных предложений на профессиональные темы;	
З2 – основные общеупотребительные глаголы (бытовая и профессиональная лексика);	
З3 – лексический минимум, относящийся к описанию предметов, средств и процессов профессиональной деятельности;	
З4 – особенности произношения;	
З5 – правила чтения текстов профессиональной направленности.	
Уметь:	
У1 – понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые), понимать тексты на базовые профессиональные темы;	
У2 – участвовать в диалогах на знакомые общие и профессиональные темы;	
У3 – строить простые высказывания о себе и о своей профессиональной деятельности;	
У4 – кратко обосновывать и объяснять свои действия (текущие и планируемые);	
У5 – писать простые связные сообщения на знакомые или интересующие профессиональные темы.	
Знать:	
	ОК 04. Эффективно взаимодействовать и работать в коллективе и команде
	ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста
	ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках
	ПК 1.1. Формировать

31 – основные этапы разработки программного обеспечения;	алгоритмы разработки программных модулей в соответствии с техническим заданием
32 – основные принципы технологии структурного и объектно-ориентированного программирования;	
33 – актуальную нормативно-правовую базу в области документирования алгоритмов.	
Уметь:	
У1 – формировать алгоритмы разработки программных модулей в соответствии с техническим заданием;	
У2 – оформлять документацию на программные средства;	
У3 – оценивать сложность алгоритма.	
Иметь практический опыт в:	
П1 – разработке алгоритма решения поставленной задачи и реализации его средствами автоматизированного проектирования.	
Знать:	
31 – основные этапы разработки программного обеспечения;	ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием
32 – основные принципы технологии структурного и объектно-ориентированного программирования;	
33 – API современных мобильных операционных систем.	
Уметь:	
У1 – создавать программу по разработанному алгоритму как отдельный модуль;	
У2 – оформлять документацию на программные средства;	
У3 – осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.	
Иметь практический опыт в:	
П1 – разработке кода программного продукта на основе готовой спецификации на уровне модуля;	
П2 – разработке мобильных приложений.	
Знать:	ПК.1.3. Выполнять отладку программных модулей с использованием специализированных программных средств
31 – основные принципы отладки и тестирования программных продуктов;	
32 – инструментарий отладки программных продуктов.	
Уметь:	
У1 – выполнять отладку и тестирование программы на уровне модуля;	
У2 – оформлять документацию на программные средства;	
У3 – применять инструментальные средства отладки программного обеспечения.	
Иметь практический опыт в:	
П1 – использовании инструментальных средств на этапе отладки программного продукта;	
П2 – проведении тестирования программного модуля по определенному сценарию.	
Знать:	ПК 1.4. Выполнять тестирование программных модулей
31 – основные виды и принципы тестирования программных продуктов.	
Уметь:	
У1 – выполнять отладку и тестирование программы на	

уровне модуля;	
У2 – оформлять документацию на программные средства.	
Иметь практический опыт в:	
П1 – проведении тестирования программного модуля по определенному сценарию;	
П2 – использования инструментальных средства на этапе тестирования программного продукта.	
Знать:	
З1 – способы оптимизации и приемы рефакторинга;	
З2 – инструментальные средства анализа алгоритма;	
З3 – методы организации рефакторинга и оптимизации кода;	
З4 – принципы работы с системой контроля версий.	
Уметь:	
У1 – выполнять оптимизацию и рефакторинг программного кода;	
У2 – работать с системой контроля версий.	
Иметь практический опыт в:	
П1 – анализе алгоритмов, в том числе с применением инструментальных средств;	
П2 – осуществлении рефакторинга и оптимизации программного кода.	
Знать:	
З1 – модели процесса разработки программного обеспечения;	
З2 – основные принципы процесса разработки программного обеспечения;	
З3 – основные подходы к интегрированию программных модулей;	
З4 – основы верификации и аттестации программного обеспечения;	
З5 – методы и способы идентификации сбоев и ошибок при интеграции приложений;	
З6 – методы и схемы обработки исключительных ситуаций;	
З7 – основные методы и виды тестирования программных продуктов;	
З8 – приемы работы с инструментальными средствами тестирования и отладки;	
З9 – стандарты качества программной документации;	
З10 – основы организации инспектирования и верификации;	
З11 – встроенные и основные специализированные инструменты анализа качества программных продуктов;	
З12 – методы организации работы в команде разработчиков.	
Уметь:	
У1 – использовать выбранную систему контроля версий;	
У2 – анализировать проектную и техническую документацию;	
У3 – выполнять тестирование интеграции;	
У4 – организовывать постобработку данных;	
	ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.
	ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения

У5 – использовать приемы работы в системах контроля версий;		
У6 – оценивать размер минимального набора тестов;		
У7 – разрабатывать тестовые пакеты и тестовые сценарии;		
У8 – выполнять ручное и автоматизированное тестирование программного модуля;		
У9 – выявлять ошибки в системных компонентах на основе спецификаций.		
Иметь практический опыт в:		
П1 – разработке тестовых наборов (пакетов) для программного модуля;		
П2 – разработке тестовых сценариев программного средства;		
П3 – инспектировании разработанных программных модулей на предмет соответствия стандартам кодирования.		
Знать:		ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования
З1 – модели процесса разработки программного обеспечения;		
З2 – основные принципы процесса разработки программного обеспечения;		
З3 – основные подходы к интегрированию программных модулей;		
З4 – основы верификации и аттестации программного обеспечения;		
З5 – стандарты качества программной документации;		
З6 – основы организации инспектирования и верификации;		
З7 – встроенные и основные специализированные инструменты анализа качества программных продуктов;		
З8 – методы организации работы в команде разработчиков.		
Уметь:		
У1 – использовать выбранную систему контроля версий;		
У2 – использовать методы для получения кода с заданной функциональностью и степенью качества;		
У3 – анализировать проектную и техническую документацию;		
У4 – организовывать постобработку данных;		
У5 – использовать приемы работы в системах контроля версий;		
У6 – выявлять ошибки в системных компонентах на основе спецификаций.		
Иметь практический опыт в:		
П1 – инспектировании разработанных программных модулей на предмет соответствия стандартам кодирования.		

3. ОЦЕНКА ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Контроль и оценка освоения учебной дисциплины по темам

Предметом оценки служат знания, умения и практический опыт, предусмотренные ФГОС СПО, направленные на формирование общих и профессиональных компетенций.

Контроль и оценка освоения учебной дисциплины по темам

Элемент учебной дисциплины	Формы и методы контроля			
	Текущий контроль		Промежуточная аттестация	
	Форма контроля	Проверяемые компетенции/ знания/ умения/ практический опыт	Форма контроля	Проверяемые компетенции/ знания/ умения/ практический опыт
Тема 1.1. Языки программирования	Практические работы; Устный опрос; Тестовые задания	ОК 01: 31-36, У1-У9; ОК 02: 31-34, У1-У9; ОК 04: 31, 32, У1, У2; ОК 05: 31, 32, У1; ОК 09: 31-3-5, У1-У5; ПК 1.1: 31-33, У1-У3, П1; ПК 1.2: 31-33, У1-У3, П1, П2; ПК 1.3: 31, 32, У1-У3, П1, П2; ПК 1.4: 31, У1, У2, П1, П2; ПК 1.5: 31-34, У1, У2, П1, П2; ПК 2.4: 31-312, У1-У9, П1-П3; ПК 2.5: 31-38, У1-У6, П1.	Контрольная работа / Экзамен	ОК 01: 31-36, У1-У9; ОК 02: 31-34, У1-У9; ОК 04: 31, 32, У1, У2; ОК 05: 31, 32, У1; ОК 09: 31-3-5, У1-У5; ПК 1.1: 31-33, У1-У3, П1; ПК 1.2: 31-33, У1-У3, П1, П2; ПК 1.3: 31, 32, У1-У3, П1, П2; ПК 1.4: 31, У1, У2, П1, П2; ПК 1.5: 31-34, У1, У2, П1, П2; ПК 2.4: 31-312, У1-У9, П1-П3; ПК 2.5: 31-38, У1-У6, П1.
Тема 1.2. Типы данных				
Тема 2.1. Операторы языка программирования				
Тема 3.1. Процедуры и функции				
Тема 3.2. Структуризация в программировании				
Тема 3.3. Модульное программирование				
Тема 4.1. Указатели				
Тема 5.1. Основные принципы объектно-ориентированного программирования (ООП)				
Тема 5.2. Интегрированная среда разработчика				
Тема 5.3. Визуальное событийно-управляемое программирование				
Тема 5.4. Разработка оконного приложения				
Тема 5.5. Этапы разработки приложений				
Тема 5.6. Иерархия классов				

3.2. Оценочные средства для текущего контроля

Тематика практических работ

1. Знакомство со средой программирования.
2. Составление программ линейной структуры.
3. Составление программ разветвляющейся структуры.
4. Составление программ циклической структуры
5. Обработка одномерных массивов.
6. Обработка двумерных массивов.
7. Работа со строками.
8. Работа с данными типа множество.
9. Файлы последовательного доступа.
10. Типизированные файлы.
11. Нетипизированные файлы.
12. Организация процедур.
13. Организация функций.
14. Применение рекурсивных функций.
15. Программирование модуля.
16. Создание библиотеки подпрограмм.
17. Использование указателей для организации связанных списков.
18. Изучение интегрированной среды разработчика.
19. Создание проекта с использованием компонентов для работы с текстом.
20. Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени.
21. События компонентов (элементов управления), их сущность и назначение.
22. Создание процедур на основе событий.
23. Создание проекта с использованием кнопочных компонентов.
24. Создание проекта с использованием компонентов стандартных диалогов и системы меню.
25. Разработка функциональной схемы работы приложения.
26. Разработка оконного приложения с несколькими формами.
27. Разработка игрового приложения.
28. Создание процедур обработки событий. Компиляция и запуск приложения.
29. Разработка интерфейса приложения.
30. Тестирование, отладка приложения.
31. Классы ООП: виды, назначение, свойства, методы, события.
32. Объявления класса.
33. Создание наследованного класса.
34. Программирование приложений.
35. Перегрузка методов.

Задания для устного опроса

1. Определение алгоритма. Свойства алгоритма. Формы записи алгоритмов. Примеры.
2. Запись алгоритмов блок-схемами. Основные элементы блок-схем.
3. Алгоритмы с ветвлением. Пример алгоритма.
4. Алгоритм цикла с предусловием. Пример алгоритма.
5. Алгоритм цикла с постусловием. Пример алгоритма.
6. Алгоритм цикла с управляющей переменной. Пример алгоритма.
7. Основные типы данных
8. Целый и вещественный типы данных. Операции с переменными этого типа.
9. Логический тип данных. Символьный тип данных. Операции с переменными этого типа.
10. Назовите поколения языков программирования и их характеристики.
11. Дайте определение алфавита и лексики языка программирования. Приведите пример.

12. Дайте определение синтаксиса и семантики программирования. Приведите пример.
13. Из каких частей состоит исходная программа.
14. Что такое система программирования. Назовите классы систем программирования.
15. Объясните суть процессов трансляции и компиляции.
16. Что такое библиотеки подпрограмм и для чего их используют.
17. Файл. Типы файлов.
18. Общие принципы разработки ПО.
19. Частотный принцип разработки ПО и принцип модульности.
20. Принцип функциональной избирательности при разработке ПО и принцип генерируемости.
21. Принцип функциональной избыточности при разработке ПО и принцип «по умолчанию».
22. Общесистемные принципы разработки ПО.

Тестовые задания

1. Алгоритм называется циклическим, если:
 - а) он составлен так, что его выполнение предполагает многократное повторение одних и тех же действий;
 - б) последовательность выполнения его команд зависит от истинности тех или иных условий;
 - в) он представим в табличной форме
 - г) его команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий;
2. Какое из свойств алгоритма описывает возможность применения алгоритма к целому классу задач?
 - а) определенность
 - б) дискретность
 - в) массовость
 - г) конечность
3. Примером разветвленного алгоритма является:
 - а) переход улицы по сигналу светофора
 - б) жизнь растения
 - в) заваривание чая
 - г) круговорот воды в природе
4. Какую структуру имеет алгоритм, в котором каждое действие выполняется ровно один раз?
 - а) ветвление
 - б) цикл
 - в) выбор
 - г) линейную
5. Программа, которая обеспечивает последовательный "перевод" команд программы на машинный язык с одновременным их выполнением, называется -?
 - а) компрессор
 - б) компилятор

- в) интерпретатор
- г) декомпрессор

6. Определите значение переменной "с" после выполнения фрагмента программы.

a: = - 2;

b: = - 3;

a: =b + a * 3;

Если a < b то c: =a - b иначе c: = b - a;

- а) 6
- б) - 12
- в) - 6
- г) 12

7. Определите значение переменной "s" после выполнения фрагмента программы.

x: = 4;

y: = 3;

Если x < y то s: = x + y иначе s: = x - y;

- а) 5
- б) 1
- в) -1
- г) -5

8. Какое из предложенных выражений может быть использовано в качестве условий?

- а) (x<-3) или (x<>5)
- б) t*4-3
- в) x-y
- г) c

9. При присваивании изменяется:

- а) тип переменной
- б) значение константы
- в) имя переменной
- г) значение переменной

10. Датчик случайных чисел генерирует...

- а) Различные графические фигуры
- б) Последовательность одинаковых чисел
- в) Последовательность случайных чисел
- г) Число, используя указанный диапазон

3.3 Оценочные средства для промежуточной аттестации

Вопросы к контрольной работе

1. Дайте определение алгоритму. Дайте пояснение, какими способами можно осуществить представление логики. Дайте определение свойствам алгоритма: конечность, дискретность.
2. Дайте определение алгоритму. Дайте пояснение, какими способами можно осуществить представление логики.
3. Дайте определение понятию алгоритм. Дайте определение свойствам алгоритма: определенность, массовость
4. Дайте определения следующим понятиям: линейный алгоритм, разветвляющийся алгоритм, циклический алгоритм, цикл программы, вспомогательный алгоритм
5. Приведите основные структуры используемые при словесном описании алгоритма. Дайте определение алгоритму, перечислите свойства алгоритма
6. Представьте основные графические элементы для отображения логики программы. Дайте следующие определения: циклический алгоритм, цикл программы.
7. Дайте пояснение, что значит интерпретируемый язык программирования. Дайте определение понятию скрипт. Назовите основные типы данных языка Python. Приведите функции применяемые в модуле Math.
8. Приведите пример использования логических выражений в языке Python. Поясните принцип работы конструкций с логическими операциями OR и AND
9. Приведите структуру условного оператора IF. Приведите пример с использованием вложенного кода. Приведите примеры каскадных условных конструкций
10. Поясните каким образом осуществляется объединение блока кода в одну инструкцию. Приведите пример конструкций «вилка», «обход».
11. Дайте определение циклу. Дайте определение циклу с предусловием. Приведите пример использования такого цикла. Изобразите такой цикл в блок-диаграмме
12. Дайте определение циклу. Дайте определение циклу с постусловием. Приведите пример использования такого цикла. Изобразите такой цикл в блок-диаграмме
13. Дайте определение циклу. Дайте определение циклу с определенным числом повторений. Приведите пример использования такого цикла. Изобразите такой цикл в блок-диаграмме. Назовите каким образом можно реализовать обратный цикл
14. Дайте определение типу данных «строка». Поясните принцип обработки строк в Python. Поясните, каким образом в питоне можно осуществить срез в строке. Перечислите методы работы со строковыми типами данных и поясните принцип их работы их использования. Приведите пример рекурсии функции.
15. Дайте определение типу данных «строка». Поясните принцип обработки строк в Python. Поясните, каким образом в питоне можно осуществить срез в строке. Перечислите методы работы со строковыми типами данных и поясните принцип их работы.
16. Поясните принцип работы следующих методов работы со строками: find, rfind, replace, count. Приведите пример их использования. Дайте определение понятию «строка»
17. Дайте определение массиву. Приведите примеры массивов. Назовите основные отличия при работе со списками в Python. Перечислите и поясните принцип работы методов при работе со списками.
18. Дайте определение типу данных «словарь» в языке Python. Перечислите и поясните принцип работы методов для обработки словарей
19. Дайте определение типу данных «список». Назовите основные группы массивов и изобразите их графически. Приведите пример программы с вложенным циклом. Поясните, каким образом вложенный цикл позволит обрабатывать вложенные списки. Опишите структуру кода функции.
20. Дайте определение процедуре и подпрограмме. Дайте определение аргументу функции.
21. Приведите пример создания процедуры или функции. Поясните понятие рекурсия

функции.

22. Дайте определения следующим понятиям: подпрограмма, процедура, функция. Поясните назначение подпрограмм. Поясните понятия глобальная и локальная переменная
23. Дайте определение понятию файл. Приведите примеры открытия и чтения файла в разных режимах.
24. Дайте определение понятию файл. Приведите примеры открытия и записи данных в файл.
25. Опишите принцип работы следующих методов: split, join, append. Приведите примеры использования данных методов. Дайте определение понятию «Список»
26. Приведите пример использования логических выражений в языке Python. Поясните принцип работы конструкций с логическими операциями OR и AND
27. Дайте определение типу данных «словарь» в языке Python. Перечислите и поясните принцип работы методов для обработки словарей

Вопросы к экзамену

1. Развитие языков программирования. Обзор языков программирования.
2. Области применения языков программирования.
3. Стандарты языков программирования. Среда проектирования.
4. Компиляторы и интерпретаторы.
5. Жизненный цикл программы.
6. Программа. Программный продукт и его характеристики
7. Типы данных.
8. Простые типы данных.
9. Производные типы данных.
10. Структурированные типы данных.
11. Операции и выражения.
12. Правила формирования и вычисления выражений.
13. Структура программы. Ввод и вывод данных.
14. Оператор присваивания.
15. Составной оператор.
16. Условный оператор.
17. Оператор выбора.
18. Цикл с постусловием.
19. Цикл с предусловием.
20. Цикл с параметром.
21. Вложенные циклы.
22. Массивы. Двумерные массивы.
23. Строки. Стандартные процедуры и функции для работы со строками
24. Структурированный тип данных – множество.
25. Операции над множествами.
26. Комбинированный тип данных – запись.
27. Файлы последовательного доступа. Файлы прямого доступа
28. Общие сведения о подпрограммах. Определение и вызов подпрограмм.
29. Область видимости и время жизни переменной.
30. Механизм передачи параметров.
31. Организация функций.
32. Рекурсия. Программирование рекурсивных алгоритмов
33. Основы структурного программирования. Методы структурного программирования
34. Модульное программирование. Понятие модуля.
35. Структура модуля. Компиляция и компоновка программы. Стандартные модули
36. Указатели. Описание указателей.
37. Основные понятия и применение динамически распределяемой памяти.
38. Создание и удаление динамических переменных.

39. Структуры данных на основе указателей. Задача о стеке.
40. История развития объектно-ориентированного программирования (ООП).
41. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс.
42. Основные принципы ООП: инкапсуляция, наследование, полиморфизм.
43. Классы объектов. Компоненты и их свойства.
44. Событийно-управляемая модель программирования. Компонентноориентированный подход
45. Требования к аппаратным и программным средствам интегрированной среды разработчика.
46. Интерфейс среды разработчика: характеристика, основные окна, инструменты, объекты.
47. Форма и размещение на ней управляющих элементов. Панель компонентов и их свойства. Окно кода проекта.
48. Состав и характеристика проекта. Выполнение проекта.
49. Настройка среды и параметров проекта. Панель компонентов и их свойства.
50. Окно кода проекта. Состав и характеристика проекта.
51. Выполнение проекта. Настройка среды и параметров проекта.
52. Основные компоненты (элементы управления) интегрированной среды разработки, их состав и назначение. Дополнительные элементы управления.
53. Свойства компонентов. Виды свойств.
54. Синтаксис определения свойств. Назначения свойств и их влияние на результат.
55. Управление объектом через свойства. События компонентов (элементов управления), их сущность и назначение.
56. Создание процедур на основе событий.
57. Разработка функционального интерфейса приложения.
58. Создание интерфейса приложения. Разработка функциональной схемы работы приложения. Разработка игрового приложения
59. Разработка приложения. Проектирование объектно-ориентированного приложения. Создание интерфейса пользователя.
60. Тестирование, отладка приложения
61. Классы ООП: виды, назначение, свойства, методы, события.
62. Перегрузка методов. Тестирование и отладка приложения.

4. КРИТЕРИИ ОЦЕНКИ

Для обучающихся по образовательным программам среднего профессионального образования применяется пятибалльная шкала знаний, умений, практического опыта.

Шкалы оценивания		Критерии оценивания
пятибалльная	зачет	
«Отлично» - 5 баллов	зачет	<p>Показывает высокий уровень сформированности компетенций, т.е.:</p> <ul style="list-style-type: none"> – демонстрирует глубокое и прочное освоение материала; – исчерпывающе, четко, последовательно, грамотно и логически стройно излагает теоретический материал; – правильно формирует определения; – демонстрирует умения самостоятельной работы с нормативно-правовой литературой; – умеет делать выводы по излагаемому материалу.
«Хорошо» - 4 балла	зачет	<p>Показывает достаточный уровень сформированности компетенций, т.е.:</p> <ul style="list-style-type: none"> – демонстрирует достаточно полное знание материала, основных теоретических положений; – достаточно последовательно, грамотно логически стройно излагает материал; – демонстрирует умения ориентироваться в нормативно-правовой литературе; – умеет делать достаточно обоснованные выводы по излагаемому материалу.
«Удовлетворительно» - 3 балла	Зачтено	<p>Показывает пороговый уровень сформированности компетенций, т.е.:</p> <ul style="list-style-type: none"> – демонстрирует общее знание изучаемого материала; – испытывает затруднения при ответах на дополнительные вопросы; – знает основную рекомендуемую литературу; – умеет строить ответ в соответствии со структурой излагаемого материала.
«Неудовлетворительно» - 2 балла	Не зачтено	<p>Ставится в случае:</p> <ul style="list-style-type: none"> – незнания значительной части программного материала; – не владения понятийным аппаратом дисциплины; – допущения существенных ошибок при изложении учебного материала; – неумения строить ответ в соответствии со структурой излагаемого вопроса; – неумения делать выводы по излагаемому материалу.

Критерии оценки тестовых заданий

Процент выполненных тестовых заданий	Оценка
до 50%	неудовлетворительно
50-69%	удовлетворительно
70-84%	хорошо
85-100%	отлично