

**Приложение ППСЗ по специальности 10.02.05 Обеспечение информационной безопасности автоматизированных систем 2021-2022 уч.г.:
Комплект контрольно-оценочных средств учебной дисциплины
ОП.03 Основы алгоритмизации и программирования**

**ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ БЕЛГОРОДСКОЙ ОБЛАСТИ
ОБЛАСТНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
«АЛЕКСЕЕВСКИЙ КОЛЛЕДЖ»**

**Комплект
контрольно-оценочных средств**
учебной дисциплине
ОП.03 Основы алгоритмизации и программирования
для специальности
**10.02.05 Обеспечение информационной безопасности
автоматизированных систем**

Алексеевка – 2021

Комплект контрольно-оценочных средств разработан на основе Федерального государственного образовательного стандарта среднего профессионального образования по специальности 10.02.05 Обеспечение информационной безопасности автоматизированных систем, утвержденного приказом Министерства образования и науки Российской Федерации от 9 декабря 2016 года № 1553.

Составитель:

Дешина И.А., преподаватель ОГАОУ «Алексеевский колледж»

СОДЕРЖАНИЕ

1. Паспорт комплекта оценочных средств
 - 1.1 Область применения комплекта оценочных средств
 - 1.2 Цель и планируемые результаты освоения учебной дисциплины
 - 1.3. Контроль и оценка результатов освоения учебной дисциплины
2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, характеризующих этапы формирования компетенций в процессе освоения учебной дисциплины для проведения текущего контроля успеваемости обучающихся
3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, характеризующих этапы формирования компетенций в процессе освоения учебной дисциплины для организации промежуточной аттестации в форме дифференцированного зачета
4. Информационное обеспечение

1. Паспорт комплекта оценочных средств

В соответствии с Федеральным государственным образовательным стандартом среднего профессионального образования (далее – ФГОС СПО) колледж самостоятельно планирует результаты обучения по учебной дисциплине ОП.03 Основы алгоритмизации и программирования, которые соотнесены с требуемыми результатами освоения образовательной программы (компетенциями выпускников). Совокупность запланированных результатов обучения должна обеспечивать выпускнику освоение всех общих компетенций (далее – ОК), профессиональных компетенций (далее – ПК), установленных ФГОС СПО.

Контрольно-оценочные средства (далее - КОС) предназначены для контроля и оценки образовательных достижений обучающихся по учебной дисциплине ОП.03 Основы алгоритмизации и программирования.

КОС включают типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, характеризующих этапы формирования компетенций в процессе освоения образовательной программы для проведения текущего контроля успеваемости обучающихся и организации промежуточной аттестации в форме **дифференцированного зачета**.

КОС разработан на основании рабочей программы учебной дисциплины ОП.03 Основы алгоритмизации и программирования.

1.2 Цель и планируемые результаты освоения учебной дисциплины:

Таблица 1

Код ПК, ОК	Умения	Знания
ОК 01, ОК 02, ОК 03, ПК 2.1, ПК 2.2, ПК 2.3, ПК 2.4, ПК 2.6.	– работать в среде программирования; – использовать языки программирования высокого уровня.	– типы данных; – базовые конструкции изучаемых языков программирования; – интегрированные среды программирования на изучаемых языках.

В результате освоения учебной дисциплины обучающийся должен **уметь**:

- У1. работать в среде программирования;
- У2. использовать языки программирования высокого уровня.

В результате освоения учебной дисциплины обучающийся должен **знать**:

- З1. типы данных;
- З2. базовые конструкции изучаемых языков программирования;
- З3. интегрированные среды программирования на изучаемых языках.

Профессиональные и общие компетенции, которые формируются при изучении учебной дисциплины:

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам

ОК 02 Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности

ОК 03 Планировать и реализовывать собственное профессиональное и личностное развитие

ПК 2.1. Осуществлять установку и настройку отдельных программных, программно-аппаратных средств защиты информации.

ПК 2.2. Обеспечивать защиту информации в автоматизированных системах отдельными программными, программно-аппаратными средствами.

ПК 2.3. Осуществлять тестирование функций отдельных программных и программно-аппаратных средств защиты информации.

ПК 2.4. Осуществлять обработку, хранение и передачу информации ограниченного доступа

ПК 2.6. Осуществлять регистрацию основных событий в автоматизированных (информационных) системах, в том числе с использованием программных и программно-аппаратных средств обнаружения, предупреждения и ликвидации последствий компьютерных атак.

Планируемые личностные результаты освоения рабочей программы учебной дисциплины:

ЛР 2. Проявляющий активную гражданскую позицию, демонстрирующий приверженность принципам честности, порядочности, открытости, экономически активный и участвующий в студенческом и территориальном самоуправлении, в том числе на условиях добровольчества, продуктивно взаимодействующий и участвующий в деятельности общественных организаций.

ЛР 3. Соблюдающий нормы правопорядка, следующий идеалам гражданского общества, обеспечения безопасности, прав и свобод граждан России. Лояльный к установкам и проявлениям представителей субкультур, отличающий их от групп с деструктивным и девиантным поведением. Демонстрирующий неприятие и предупреждающий социально опасное поведение окружающих.

ЛР 4. Проявляющий и демонстрирующий уважение к людям труда, осознающий ценность собственного труда. Стремящийся к формированию в сетевой среде лично и профессионального конструктивного «цифрового следа».

ЛР 7. Осознающий приоритетную ценность личности человека; уважающий собственную и чужую уникальность в различных ситуациях, во всех формах и видах деятельности.

1.3. Контроль и оценка результатов освоения учебной дисциплины

Таблица 2

<i>Результаты обучения</i>	<i>Критерии оценки</i>	<i>Формы и методы оценки</i>
Знания: – типы данных; – базовые конструкции изучаемых языков программирования; – интегрированные среды программирования на изучаемых языках	Демонстрация знаний базовых конструкций изучаемых языков программирования, интегрированных сред	Оценка знаний в ходе тестирования и опроса
Умения: – работать в среде программирования; – использовать языки программирования высокого уровня	Умение работать в среде программирования, выполнять индивидуальные практические задания	Оценка результатов деятельности обучающегося при выполнении и защите результатов практических занятий, тестирование

2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, характеризующих этапы формирования компетенций в процессе освоения учебной дисциплины для проведения текущего контроля успеваемости обучающихся

2.1. Тестовые задания

Раздел 1. Основные принципы алгоритмизации и программирования

Задание № 1. В задании установите соответствие между понятием и его определением. Ответ запишите в таблицу.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 02, ОК 03, ПК 2.1., ПК 2.4.)

Сопоставьте определение с его описанием:

1. Алгоритм	а. Последовательность действий, направленных на решение задачи.
2. Переменная	б. Конструкция, позволяющая выполнять разные действия в зависимости от условия
3. Цикл	в. Конструкция, позволяющая повторять выполнение блока кода.
4. Условный оператор	г. Именованная область памяти для хранения данных.

Запишите ответ:

1.	
2.	
3.	
4.	

Задание № 2. Прочитайте вопрос, выберите один правильный ответ. Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 03, ПК 2.1.)

Что такое алгоритм?

1. Язык программирования
2. Последовательность действий для решения задачи
3. Тип данных
4. Математическая формула

Задание № 3. Прочитайте вопрос, выберите один правильный ответ. Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, 31, 32, 33, ОК 01, ОК 02, ПК 2.4.)

Какой тип цикла выполняется, пока условие истинно?

1. Цикл с предусловием
2. Цикл с постусловием
3. Бесконечный цикл
4. Условный оператор

Задание № 4. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.

(оцениваемые знания, умения, компетенции: У1, У2, 31, 33, ОК 01, ОК 03, ПК 2.1. ПК 2.6.)

Какие элементы являются основными в алгоритмизации?

1. Переменные
2. Комментарии
3. Условные операторы
4. Графические элементы

Задание № 5. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.

(оцениваемые знания, умения, компетенции: У2, 31, 32, ОК 02, ОК 03, ПК 2.1.)

Какие типы данных используются в программировании?

1. Целочисленный
2. Алгоритмический
3. Строковый
4. Графический

Задание № 6. Прочитайте вопрос, запишите развернутый ответ.

(оцениваемые знания, умения, компетенции: У1, У2, 31, 32, 33, ОК 01, ОК 02, ПК 2.4., ПК 2.6.)

Опишите основные этапы разработки алгоритма

Ответ:

Задание № 7. Прочитайте вопрос, запишите развернутый ответ.
(оцениваемые знания, умения, компетенции: У1, У2, З2, З3, ОК 01, ОК 03, ПК 2.1., ПК 2.2., ПК 2.3., ПК 2.4.)

Какие преимущества предоставляет использование функций в программировании?

Ответ:

Задание № 8. Прочитайте вопрос, запишите короткий ответ.
(оцениваемые знания, умения, компетенции: У1, З1, З2, З3, ОК 01, ОК 03, ПК 2.1., ПК 2.4.)

Назовите два основных типа циклов в программировании

Ответ:

Задание № 9. Прочитайте вопрос, запишите короткий ответ.
(оцениваемые знания, умения, компетенции: У1, У2, З3, ОК 01, ОК 02, ПК 2.4.)

Именованная область памяти для хранения данных в программировании?

Ответ:

	<p>4. Реализация алгоритма — написание кода на языке программирования.</p> <p>5. Тестирование и отладка — проверка корректности работы алгоритма.</p>
7	<p>Использование функций позволяет:</p> <ol style="list-style-type: none"> 1. Упростить код за счет повторного использования. 2. Улучшить читаемость программы. 3. Упростить тестирование и отладку. 4. Разделить программу на логические блоки.
8	с предусловием и с постусловием.
9	переменная
10	<ol style="list-style-type: none"> 1. Начало алгоритма. 2. Ввод числа N. 3. Инициализация переменной $sum = 0$. 4. Запуск цикла от 1 до N: <p style="margin-left: 40px;">Если текущее число четное, добавить его к sum.</p> 5. Вывод sum. 6. Конец алгоритма <p>$sum = 0$ Ввод N Для i от 1 до N: Если $i \% 2 == 0$: $sum = sum + i$ Вывод sum</p>

Критерии оценивания ответов, полученных в ходе тестирования

За каждый верный ответ выставляется 1 балл, за неверный ответ – 0 баллов. Баллы, полученные обучающимися за выполненные задания, суммируются.

Результаты тестирования определяются в разрезе каждого обучающегося в баллах и оценках.

Результаты тестирования			
Баллы	Оценка	Доля выполненных заданий	Уровень сформированности компетенций
0-2 баллов	2 (неудовлетворительно)	0-20%	низкий
3-6 баллов	3 (удовлетворительно)	30-60%	базовый
7-8 баллов	4 (хорошо)	70-80%	повышенный
9-10 баллов	5 (отлично)	90-100%	высокий

Раздел 2. Язык программирования

Задание № 1. В задании установите правильную последовательность. Ответ запишите в таблицу.

(оцениваемые знания, умения, компетенции: У1, У2, ЗЗ, ОК 01, ПК 2.1.)

Установите правильную последовательность шагов для установки Visual Studio Code на Windows:

- а. Скачайте установочный файл с официального сайта.
- б. Запустите установочный файл.
- в. Прочитайте и примите лицензионное соглашение.
- г. Выберите папку для установки.
- д. Нажмите "Finish" для завершения установки.
- е. Выберите дополнительные параметры установки (например, создание ярлыка на рабочем столе).
- ж. Запустите Visual Studio Code.

Запишите ответ:

1.	
2.	
3.	
4.	
5.	
6.	
7.	

Задание № 2. В задании установите соответствие между понятием и его определением. Ответ запишите в таблицу.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 02, ОК 03, ПК 2.4.)

1. Компилятор	а. Программа, которая выполняет код построчно, без предварительной компиляции.
2. Интерпретатор	б. Программа, которая переводит исходный код в машинный код.
3. Синтаксис	в. Набор правил, определяющих структуру и порядок написания программ.
4. Алгоритм	г. Последовательность действий для решения задачи.

Запишите ответ:

1.	
2.	

3.	
4.	

Задание № 3. Прочитайте вопрос, выберите один правильный ответ. Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 02, ОК 03, ПК 2.1.)

Какое расширение необходимо установить для работы с Python в VS Code?

1. C++ Intellisense
2. Python
3. Java Extension Pack
4. HTML CSS Support

Задание № 4. Прочитайте вопрос, выберите один правильный ответ. Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 03, ПК 2.1.)

Как можно выбрать интерпретатор Python в VS Code?

1. Через меню "File"
2. Нажав Ctrl+Shift+P и выбрав "Python: Select Interpreter"
3. В настройках темы
4. Через панель "Explorer"

Задание № 5. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.

(оцениваемые знания, умения, компетенции: У2, З1, З2, З3, ОК 01, ОК 02, ПК 2.4.)

Какие из следующих операторов используются для работы с множествами в Python?

1. & (пересечение)
2. + (конкатенация)
3. | (объединение)
4. * (умножение)

Задание № 6. В задании установите соответствие между понятием и его определением. Ответ запишите в таблицу.

(оцениваемые знания, умения, компетенции: У1, У2, З3, ОК 02, ОК 03, ПК 2.4.)

1. Переменная	а. Структура данных, которая может хранить несколько значений одного типа
2. Функция	б. Именованная область памяти, используемая для хранения данных.
3. Массив	в. Блок кода, который выполняет определенную задачу и может возвращать значение.
4. Цикл	г. конструкция, позволяющая повторять выполнение блока кода несколько раз.

Запишите ответ:

1.	
2.	
3.	
4.	

Задание № 7. В задании установите правильную последовательность. Ответ запишите в таблицу.

(оцениваемые знания, умения, компетенции: У1, У2, ЗЗ, ОК 01, ОК 03, ПК 2.1.)

Установите правильную последовательность шагов для установки и настройке расширения Python в Visual Studio Code:

- а. Откройте Visual Studio Code.
- б. Перейдите в раздел расширений, нажав на иконку Extensions или используя сочетание клавиш Ctrl+Shift+X.
- в. Настройте интерпретатор Python
- г. В строке поиска введите "Python".
- д. Найдите расширение "Python" от Microsoft и нажмите "Install".
- е. После установки расширения перезапустите VS Code (если требуется).

Запишите ответ:

1.	
2.	
3.	
4.	
5.	
6.	

Задание № 8. Прочитайте вопрос, запишите короткий ответ.
(оцениваемые знания, умения, компетенции: У1, З1, З2, З3, ОК 01, ОК 03, ПК 2.1.)

Ключи ответов

Номер задания	Правильный ответ
1	а, б, в, г, е, д, ж
2	1 – б 2 – а 3 – в 4 – г
3	2
4	2
5	1,3
6	1 – б 2 – в 3 – а 4 – г
7	а, б, г, д, е, в,
8	Метод append().
9	Кортеж (Tuple).
10	<pre>def filter_even_numbers(numbers): return [num for num in numbers if num % 2 == 0] # Пример использования input_list = [1, 2, 3, 4, 5, 6, 7, 8, 9] output_list = filter_even_numbers(input_list) print(output_list) # Вывод: [2, 4, 6, 8]</pre>

Критерии оценивания ответов, полученных в ходе тестирования

За каждый верный ответ выставляется 1 балл, за неверный ответ – 0 баллов. Баллы, полученные обучающимися за выполненные задания, суммируются.

Результаты тестирования определяются в разрезе каждого обучающегося в баллах и оценках.

Результаты тестирования			
Баллы	Оценка	Доля выполненных заданий	Уровень сформированности компетенций
0-2 баллов	2 (неудовлетворительно)	0-20%	низкий
3-6 баллов	3 (удовлетворительно)	30-60%	базовый
7-8 баллов	4 (хорошо)	70-80%	повышенный

9-10 баллов	5 (отлично)	90-100%	высокий
-------------	-------------	---------	---------

Раздел 3. Основы объектно-ориентированного программирования

Задание № 1. В задании установите правильную последовательность. Ответ запишите в таблицу.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, ОК 01, ОК 03, ПК 2.1.)

Установите правильную последовательность этапов разработки объекта в объектно-ориентированном программировании:

а. Определение класса
б. Создание экземпляра (объекта) класса
в. Реализация методов и свойств
г. Использование объекта в программе
д. Наследование и полиморфизм (если необходимо)

Запишите ответ:

1.	
2.	
3.	
4.	
5.	

Задание № 2. Прочитайте вопрос, выберите один правильный ответ. Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З1, ОК 01, ОК 02, ОК 03, ПК 2.4., ПК 2.6.)

Какой из следующих языков программирования является объектно-ориентированным?

1. С
2. Java
3. Assembly
4. HTML

Задание № 3. Прочитайте вопрос, выберите один правильный ответ. Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 03, ПК 2.1., ПК 2.2., ПК 2.3., ПК 2.4.)

Какой из следующих принципов не является основным принципом объектно-ориентированного программирования?

1. Инкапсуляция
2. Наследование
3. Полиморфизм
4. Компиляция

Задание № 4. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.
(оцениваемые знания, умения, компетенции: У2, 31, 32, ОК 01, ОК 02, ОК 03, ПК 2.1., ПК 2.2., ПК 2.6.)

Какие из следующих утверждений верны для инкапсуляции?

1. Защита данных от несанкционированного доступа
2. Объединение данных и методов в одном классе
3. Позволяет создавать иерархии классов
4. Упрощает использование объектов

Задание № 5. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.
(оцениваемые знания, умения, компетенции: У1, У2, 31, 32, 33, ОК 01, ОК 02, ПК 2.1., ПК 2.2.)

Какие из следующих языков поддерживают полиморфизм?

1. C++
2. Python
3. Java
4. HTML

Задание № 6. Прочитайте вопрос, запишите развернутый ответ.
(оцениваемые знания, умения, компетенции: У1, У2, 31, 32, ОК 02, ОК 03, ПК 2.1., ПК 2.2.)

Объясните, что такое инкапсуляция и как она используется в объектно-ориентированном программировании

Ответ:

Задание № 7. Прочитайте вопрос, запишите развернутый ответ.
(оцениваемые знания, умения, компетенции: У1, , 32, 33, ОК 01, ОК 02, ОК 03, ПК 2.4.)

Опишите, как наследование помогает в повторном использовании кода и приведите пример его использования

Ответ:

Задание № 8. Прочитайте вопрос, запишите короткий ответ.
(оцениваемые знания, умения, компетенции: У1, У2, 31, 32, , ОК 02, ОК 03, ПК 2.1., ПК 2.2., ПК 2.6.)

Что такое объект в контексте объектно-ориентированного программирования?

Ответ:

Задание № 9. В задании установите правильную последовательность.
Ответ запишите в таблицу.
(оцениваемые знания, умения, компетенции: У1, У2, 31, 32, ОК 02, ОК 03, ПК 2.4.)

Установите правильную последовательность основных принципов объектно-ориентированного программирования:

а. Инкапсуляция
б. Наследование
в. Полиморфизм
г. Абстракция

Запишите ответ:

6	<p>Инкапсуляция — это один из основных принципов объектно-ориентированного программирования, который заключается в объединении данных и методов, работающих с этими данными, в одном классе, а также в ограничении доступа к внутренним состояниям объекта. Это достигается с помощью модификаторов доступа, таких как <code>private</code>, <code>protected</code> и <code>public</code>.</p> <p>Как используется инкапсуляция:</p> <p>Защита данных: Инкапсуляция позволяет скрыть внутренние детали реализации класса от внешнего мира. Это предотвращает несанкционированный доступ к данным и защищает их от случайных изменений.</p> <p>Упрощение интерфейса: Пользователи класса взаимодействуют с ним через публичные методы (интерфейс), не беспокоясь о том, как реализованы внутренние механизмы.</p> <p>Поддержка изменений: Изменения в реализации класса могут быть выполнены без влияния на код, который использует этот класс, если интерфейс остается неизменным.</p>
7	<p>Наследование — это механизм в объектно-ориентированном программировании, который позволяет создавать новый класс на основе существующего. Новый класс (называемый подклассом или производным классом) наследует свойства и методы родительского класса (суперкласса), что способствует повторному использованию кода и упрощает его поддержку.</p> <p>Как наследование помогает в повторном использовании кода:</p> <p>Снижение дублирования: Вместо того чтобы повторять один и тот же код в нескольких классах, можно вынести общие свойства и методы в родительский класс.</p> <p>Расширяемость: Подклассы могут добавлять новые функции или изменять поведение, унаследованное от суперкласса, что позволяет легко расширять функциональность программы.</p> <p>Упрощение поддержки: Изменения в родительском классе автоматически применяются ко всем подклассам, что упрощает процесс обновления и исправления ошибок.</p>
8	<p>Объект - это экземпляр класса, который содержит данные и методы для работы с этими данными</p>
9	<p>а, г, б, в</p>
10	<p>При разработке программы для управления библиотекой можно эффективно использовать принципы объектно-ориентированного программирования (ООП) — инкапсуляцию, наследование и полиморфизм. Вот как это можно реализовать:</p>

	<p>1. Инкапсуляция Инкапсуляция позволяет скрыть внутренние детали реализации классов и защитить данные от несанкционированного доступа. В нашей системе мы можем создать классы для книг, читателей и библиотекарей, где данные будут защищены с помощью модификаторов доступа.</p> <p>2. Наследование Наследование позволяет создавать новые классы на основе существующих, что способствует повторному использованию кода. В нашей системе можно создать базовый класс User, от которого будут наследоваться классы Reader и Librarian.</p> <p>3. Полиморфизм Полиморфизм позволяет использовать один интерфейс для работы с разными типами объектов. В нашей системе можно создать метод, который будет принимать объект типа User, и в зависимости от типа пользователя (читатель или библиотекарь) выполнять разные действия. Таким образом, в нашей программе для управления библиотекой мы используем:</p> <p>Инкапсуляцию для защиты данных о книгах и пользователях, предоставляя доступ к ним только через методы. Наследование для создания иерархии классов пользователей, что позволяет избежать дублирования кода и упрощает его поддержку. Полиморфизм для обработки действий пользователей (читателей и библиотекарей) через общий интерфейс, что делает код более гибким и расширяемым. Эта структура позволяет легко добавлять новые функции и классы в будущем, например, классы для администраторов или для управления категориями книг.</p>
--	---

Критерии оценивания ответов, полученных в ходе тестирования

За каждый верный ответ выставляется 1 балл, за неверный ответ – 0 баллов. Баллы, полученные обучающимися за выполненные задания, суммируются.

Результаты тестирования определяются в разрезе каждого обучающегося в баллах и оценках.

Результаты тестирования			
Баллы	Оценка	Доля выполненных заданий	Уровень сформированности компетенций
0-2 баллов	2 (неудовлетворительно)	0-20%	низкий
3-6 баллов	3 (удовлетворительно)	30-60%	базовый

7-8 баллов	4 (хорошо)	70-80%	повышенный
9-10 баллов	5 (отлично)	90-100%	высокий

Раздел 4. Модульное программирование

Задание № 1. В задании установите соответствие между понятием и его определением. Ответ запишите в таблицу.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, ОК 02, ОК 03, ПК 2.1.)

Сопоставьте определение с его описанием:

1. Модуль	а. Функция, которая возвращает значение, вычисленное на основе входных данных
2. Пакет	б. Файл, содержащий код на Python, который можно импортировать и использовать
3. Функция	в. Коллекция модулей, организованных в директории с файлом <code>__init__.py</code>
4. Импорт	г. Процесс включения кода из одного модуля в другой

Запишите ответ:

1.	
2.	
3.	
4.	

Задание № 2. Прочитайте вопрос, выберите один правильный ответ.

Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З3, ОК 01, ОК 03, ПК 2.1., ПК 2.2., ПК 2.3.)

Что такое модуль в Python?

1. Это функция, которая выполняет определенную задачу.
2. Это файл, содержащий код на Python, который можно импортировать и использовать.
3. Это пакет, содержащий несколько модулей.
4. Это встроенная библиотека Python

Задание № 3. Прочитайте вопрос, выберите один правильный ответ.

Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З2, З3, ОК 01, ОК 02, ОК 03, ПК 2.1., ПК 2.2.)

Какой оператор используется для импорта всего содержимого модуля?

1. `import module_name`
2. `from module_name import *`
3. `import * from module_name`
4. `include module_name`

Задание № 4. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.

(оцениваемые знания, умения, компетенции: У1, У2, З1, ОК 02, ОК 03, ПК 2.1., ПК 2.2., ПК 2.6.)

Какие из следующих утверждений верны относительно модулей в Python?

1. Модуль может содержать функции, классы и переменные.
2. Модуль может быть импортирован только один раз в программе.
3. Модуль может быть пакетом, если содержит файл `__init__.py`.
4. Модуль всегда должен находиться в той же директории, что и основной скрипт.

Задание № 5. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 02, ПК 2.6.)

Какие из следующих операторов импорта являются корректными?

1. `import math`
2. `from math import sqrt`
3. `import sqrt from math`
4. `from math import *`

Задание № 6. Прочитайте вопрос, запишите развернутый ответ.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, ОК 02, ОК 03, ПК 2.4., ПК 2.6.)

Объясните, как работает механизм импорта модулей в Python. Какие этапы происходят при импорте модуля?

Ответ:

Задание № 7. В задании установите правильную последовательность. Ответ запишите в таблицу.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, ОК 01, ОК 02, ОК 03, ПК 2.1., ПК 2.4.)

Установите правильную последовательность этапов разработки модульной программ:

а. Определение интерфейса модуля
б. Реализация функциональности модуля
в. Тестирование модуля
г. Документирование модуля
д. Интеграция модуля в основную программу

Запишите ответ:

1.	
2.	
3.	
4.	
5.	

Задание № 8. Прочитайте вопрос, запишите короткий ответ.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 02, ОК 03, ПК 2.1.)

Какой файл обязательно должен присутствовать в директории, чтобы она считалась пакетом в Python?

Ответ:

Задание № 9. Прочитайте вопрос, запишите короткий ответ.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 02, ОК 03, ПК 2.1., ПК 2.3., ПК 2.6.)

Как можно импортировать только одну функцию из модуля?

Ответ:

	<ol style="list-style-type: none"> 1. Поиск модуля: Python ищет модуль в директориях, указанных в <code>sys.path</code>. 2. Компиляция: Если модуль найден, он компилируется в байт-код (файл <code>.pyc</code>). 3. Выполнение: Код модуля выполняется, и все определения (функции, классы, переменные) загружаются в память. 4. Создание объекта модуля: Python создает объект модуля, который добавляется в <code>sys.modules</code>. 5. Импорт: Импортируемый модуль становится доступным в текущем пространстве имен.
7	а, б, в, д, г
8	<code>__init__.py</code>
9	<code>from module_name import function_name</code>
10	<pre># math_operations.py def add(a, b): return a + b # main.py from math_operations import add result = add(5, 3) print(result) # Вывод: 8</pre>

Критерии оценивания ответов, полученных в ходе тестирования

За каждый верный ответ выставляется 1 балл, за неверный ответ – 0 баллов. Баллы, полученные обучающимися за выполненные задания, суммируются.

Результаты тестирования определяются в разрезе каждого обучающегося в баллах и оценках.

Результаты тестирования			
Баллы	Оценка	Доля выполненных заданий	Уровень сформированности компетенций
0-2 баллов	2 (неудовлетворительно)	0-20%	низкий
3-6 баллов	3 (удовлетворительно)	30-60%	базовый
7-8 баллов	4 (хорошо)	70-80%	повышенный
9-10 баллов	5 (отлично)	90-100%	высокий

2.2. Вопросы для устного опроса.

Тема 1.1 Основные понятия алгоритмизации

Вопросы:

1. Понятие алгоритма и его свойства (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 03, ПК 2.3., ПК 2.4.*)
2. Основные базовые типы данных и их характеристика (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 02, ОК 03, ПК 2.2., ПК 2.6.*)

Тема 1.2 Принципы разработки алгоритмов

Вопросы:

1. Принципы построения алгоритмов: использование базовых структур, метод последовательной детализации, сборочный метод. (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 03, ПК 2.1., ПК 2.2., ПК 2.3.*)
2. Разработка алгоритмов сложной структуры (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 02, ПК 2.4., ПК 2.6.*)

Тема 1.3 Языки и системы программирования

Вопросы:

1. Классификация языков программирования (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 02, ОК 03, ПК 2.1., ПК 2.2., ПК 2.3., ПК 2.6.*)

Тема 1.4 Парадигмы программирования

Вопросы:

1. Этапы разработки программ: системный анализ, алгоритмизация, программирование, отладка, сопровождение (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 02, ОК 03, ПК 2.3., ПК 2.4.*)

Тема 1.5 Принципы отладки и тестового контроля

Вопросы:

1. Понятие отладки. Понятие тестового контроля и набора тестов (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, ОК 02, ОК 03, ПК 2.1., ПК 2.2., ПК 2.6.*)

Тема 2.1 Характеристика языка

Вопросы:

1. История и особенности языка. Области применения (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, ОК 01, ОК 02, ПК 2.2., ПК 2.3., ПК 2.4.*)

Тема 2.2 Элементы языка. Простые типы данных

Вопросы:

1. Алфавит и лексика языка. (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 03, ПК 2.1., ПК 2.2., ПК 2.6.*)

Тема 2.3 Базовые конструкции структурного программирования

Вопросы:

1. Организация ветвлений. Операторы циклов (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 02, ПК 2.1., ПК 2.2., ПК 2.3.*)

Тема 2.4 Работа с массивами и указателями. Структурные типы данных

Вопросы:

1. Одномерные и многомерные массивы, их формирование, сортировка, обработка (*оцениваемые знания, умения, компетенции: У1, У2, З2, З3, ОК 01, ОК 02, ОК 03, ПК 2.1., ПК 2.2.*)
2. Работа со строками. Структуры и объединения (*оцениваемые знания, умения, компетенции: У1, У2, З1, З3, ОК 01, ОК 02, ОК 03, ПК 2.1., ПК 2.2., ПК 2.6.*)

Тема 2.5 Процедуры и функции

Вопросы:

1. Определение процедур и функций (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 02, ОК 03, ПК 2.1., ПК 2.6.*)

Тема 2.6 Работа с файлами

Вопросы:

1. Файловый ввод/вывод (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 03, ПК 2.4., ПК 2.6.*)
2. Дополнительные операции с файлами (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 02, ОК 03, ПК 2.3., ПК 2.4., ПК 2.6.*)

Тема 3.1 Класс - как механизм создания объектов

Вопросы:

1. Понятия: класс, объект, свойства объекта, методы (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 02, ОК 03, ПК 2.2., ПК 2.3., ПК 2.4., ПК 2.6.*)
2. Спецификаторы доступа (private, public, protected) (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, ОК 01, ОК 02, ПК 2.2., ПК 2.3.*)

Тема 3.2 Принципы наследования и полиморфизма

Вопросы:

1. Механизм наследования для формирования иерархии классов (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, ОК 03, ПК 2.3., ПК 2.4.*)

Тема 3.3 Понятия деструктора и конструктора

Вопросы:

1. Назначение и свойства конструкторов, деструкторов. Их описание (*оцениваемые знания, умения, компетенции: У1, У2, З2, З3, ОК 03, ПК 2.1., ПК 2.2.*)

Тема 4.1 Понятие модульного программирования

Вопросы:

1. Модульное программирование как метод разработки программ (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 03, 2.6.*)

2. Порядок разработки программного модуля. Связность модулей (*оцениваемые знания, умения, компетенции: У1, У2, ЗЗ, ОК 01, ПК 2.1., ПК 2.6.*)

Тема 4.2 Разработка приложений

Вопросы:

1. Среда разработки приложений. Архитектура оконных приложений. (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 02, ОК 03, ПК 2.1., ПК 2.2., ПК 2.3., ПК 2.4., ПК 2.6.*)
2. Разработка приложений как многомодульного проекта. (*оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 02, ОК 03, ПК 2.1., ПК 2.2., ПК 2.3., ПК 2.4., ПК 2.6.*)

Критерии оценивания ответов на вопросы

«5» «отлично» – студент показывает глубокое и полное овладение содержанием программного материала по учебной дисциплине, в совершенстве владеет понятийным аппаратом и демонстрирует умение применять теорию на практике, решать различные практические и профессиональные задачи, высказывать и обосновывать свои суждения в форме грамотного, логического ответа (устного или письменного), а также высокий уровень овладения общими и профессиональными компетенциями и демонстрирует готовность к профессиональной деятельности;

«4» «хорошо» – студент в полном объеме освоил программный материал по учебной дисциплине, владеет понятийным аппаратом, хорошо ориентируется в изучаемом материале, осознанно применяет знания для решения практических и профессиональных задач, грамотно излагает ответ, но содержание, форма ответа (устного или письменного) имеют отдельные неточности, демонстрирует средний уровень овладения общими и профессиональными компетенциями и готовность к профессиональной деятельности;

«3» «удовлетворительно» – студент обнаруживает знание и понимание основных положений программного материала по учебной дисциплине, но излагает его неполно, непоследовательно, допускает неточности в определении понятий, в применении знаний для решения практических и профессиональных задач, не умеет доказательно обосновать свои суждения, но при этом демонстрирует низкий уровень овладения общими и профессиональными компетенциями и готовность к профессиональной деятельности;

«2» «неудовлетворительно» – студент имеет разрозненные, бессистемные знания, не умеет выделять главное и второстепенное, допускает ошибки в определении понятий, беспорядочно и неуверенно излагает программный материал по учебной дисциплине, не умеет применять знания для решения практических и профессиональных задач, не демонстрирует овладение общими

и профессиональными компетенциями и готовность к профессиональной деятельности.

3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, характеризующих этапы формирования компетенций в процессе освоения учебной дисциплины для организации промежуточной аттестации в форме дифференцированного зачета

3.1. Тестовые задания

ВАРИАНТ 1

Задание № 1. В задании установите соответствие между понятием и его определением. Ответ запишите в таблицу.

(оцениваемые знания, умения, компетенции: У1, У2, ЗЗ, ОК 01, ПК 2.2., ПК 2.3., ПК 2.4.)

Сопоставьте тип угрозы с её описанием:

1. Список (List)	а. Неизменяемая последовательность элементов.
2. Кортеж (Tuple)	б. Изменяемая последовательность элементов.
3.Словарь (Dictionary)	в. Коллекция пар "ключ-значение".
4. Множество (Set)	г. Коллекция уникальных элементов без порядка.

Запишите ответ:

1.	
2.	
3.	
4.	

Задание № 2. Прочитайте вопрос, выберите один правильный ответ. Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З2, З3, ОК 01, ОК 02, ОК 03, ПК 2.6.)

Какой тип данных в Python используется для хранения пар "ключ-значение"?

1. Список (List)
2. Кортеж (Tuple)
3. Словарь (Dictionary)
4. Множество (Set)

Задание № 3. Прочитайте вопрос, выберите один правильный ответ. Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, 31, 32, 33, ОК 02, ОК 03, ПК 2.1.)

Какой оператор используется для выполнения целочисленного деления в Python?

1. A) /
2. B) //
3. C) %
4. D) **

Задание № 4. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.

(оцениваемые знания, умения, компетенции: У1, У2, 31, 32, 33, ОК 01, ОК 03, ПК 2.1., ПК 2.6.)

Какие из следующих типов данных являются изменяемыми в Python?
(Выберите два варианта)

1. Список (List)
2. Кортеж (Tuple)
3. Словарь (Dictionary)
4. Множество (Set)

Задание № 5. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.

(оцениваемые знания, умения, компетенции: У2, 31, 32, 33, ОК 01, ОК 02, ПК 2.4.)

Какие из следующих операторов используются для работы с множествами в Python?

1. & (пересечение)
2. + (конкатенация)
3. | (объединение)
4. * (умножение)

Задание № 6. Прочитайте вопрос, запишите развернутый ответ.

(оцениваемые знания, умения, компетенции: У1, У2, 31, 32, 33, ОК 01, ОК 02, ОК 03, ПК 2.6.)

Опишите основные особенности языка Python

Ответ:

Задание № 7. Прочитайте вопрос, запишите развернутый ответ.
(оцениваемые знания, умения, компетенции: У1, У2, З1, З3, ОК 01, ОК 02, ОК 03, ПК 2.3., ПК 2.6.)

Напишите код на Python для нахождения суммы всех элементов списка

Ответ:

Задание № 8. Прочитайте вопрос, запишите короткий ответ.
(оцениваемые знания, умения, компетенции: У1, З1, З2, З3, ОК 01, ОК 03, ПК 2.1.)

Какой метод используется для добавления элемента в конец списка в Python?

Ответ:

Задание № 9. Прочитайте вопрос, запишите короткий ответ.
(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 02, ПК 2.4.)

Какой тип данных в Python используется для хранения неизменяемых последовательностей?

	директории с файлом <code>__init__.py</code>
4. Импорт	г. Процесс включения кода из одного модуля в другой

Запишите ответ:

1.	
2.	
3.	
4.	

Задание № 12. Прочитайте вопрос, выберите один правильный ответ. Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З3, ОК 01, ОК 03, ПК 2.1., ПК 2.2., ПК 2.3.)

Что такое модуль в Python?

1. Это функция, которая выполняет определенную задачу.
2. Это файл, содержащий код на Python, который можно импортировать и использовать.
3. Это пакет, содержащий несколько модулей.
4. Это встроенная библиотека Python

Задание № 13. Прочитайте вопрос, выберите один правильный ответ. Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З2, З3, ОК 01, ОК 02, ОК 03, ПК 2.1., ПК 2.2.)

Какой оператор используется для импорта всего содержимого модуля?

1. `import module_name`
2. `from module_name import *`
3. `import * from module_name`
4. `include module_name`

Задание № 14. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.

(оцениваемые знания, умения, компетенции: У1, У2, З1, ОК 02, ОК 03, ПК 2.1., ПК 2.2., ПК 2.6.)

Какие из следующих утверждений верны относительно модулей в Python?

1. Модуль может содержать функции, классы и переменные.
2. Модуль может быть импортирован только один раз в программе.
3. Модуль может быть пакетом, если содержит файл `__init__.py`.
4. Модуль всегда должен находиться в той же директории, что и основной скрипт.

Задание № 15. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.
(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 02, ПК 2.6.)

Какие из следующих операторов импорта являются корректными?

1. `import math`
2. `from math import sqrt`
3. `import sqrt from math`
4. `from math import *`

Задание № 16. Прочитайте вопрос, запишите развернутый ответ.
(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, ОК 02, ОК 03, ПК 2.4., ПК 2.6.)

Объясните, как работает механизм импорта модулей в Python. Какие этапы происходят при импорте модуля?

Ответ:

Задание № 17. Прочитайте вопрос, запишите развернутый ответ.
(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, ОК 01, ОК 02, ОК 03, ПК 2.1., ПК 2.4., ПК 2.6.)

Напишите, как можно организовать пакет в Python. Какие файлы и директории должны быть созданы?

Ответ:

Задание № 18. Прочитайте вопрос, запишите короткий ответ.
(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 02, ОК 03, ПК 2.1.)

Какой файл обязательно должен присутствовать в директории, чтобы она считалась пакетом в Python?

Ответ:

Задание № 19. Прочитайте вопрос, запишите короткий ответ.
(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 02, ОК 03, ПК 2.1., ПК 2.3., ПК 2.6.)

Как можно импортировать только одну функцию из модуля?

Ответ:

Задание № 20. Прочитайте ситуационную задачу и опишите
Напишите код для импорта и использования функции `add` из модуля `math_operations` в `main.py`.
(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, ОК 02, ОК 03, ПК 2.1.)

Вы разрабатываете проект, который состоит из нескольких модулей. Один из модулей (`math_operations.py`) содержит функции для математических операций, а другой (`main.py`) должен использовать эти функции.

Ответ:

ВАРИАНТ 2

Задание № 1. В задании установите соответствие между понятием и его определением. Ответ запишите в таблицу.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, ОК 01, ОК 03, ПК 2.2.)

Сопоставьте тип угрозы с её описанием:

1. Класс	а .Объект, который содержит данные и методы для работы с этими данными
2.Объект	б. Шаблон или чертеж для создания объектов
3.Наследование	в. Механизм, позволяющий создавать новый класс на основе существующего
4.Полиморфизм	г. Способ, позволяющий использовать один интерфейс для работы с разными типами объектов

Запишите ответ:

1.	
2.	
3.	
4.	

Задание № 2. Прочитайте вопрос, выберите один правильный ответ.

Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З1, ОК 01, ОК 02, ОК 03, ПК 2.4., ПК 2.6.)

Какой из следующих языков программирования является объектно-ориентированным?

1. С
2. Java
3. Assembly
4. HTML

Задание № 3. Прочитайте вопрос, выберите один правильный ответ. Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 03, ПК 2.1., ПК 2.2., ПК 2.3., ПК 2.4.)

Какой из следующих принципов не является основным принципом объектно-ориентированного программирования?

1. Инкапсуляция
2. Наследование
3. Полиморфизм
4. Компиляция

Задание № 4. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.

(оцениваемые знания, умения, компетенции: У2, З1, З2, ОК 01, ОК 02, ОК 03, ПК 2.1., ПК 2.2., ПК 2.6.)

Какие из следующих утверждений верны для инкапсуляции?

1. Защита данных от несанкционированного доступа
2. Объединение данных и методов в одном классе
3. Позволяет создавать иерархии классов
4. Упрощает использование объектов

Задание № 5. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 02, ПК 2.1., ПК 2.2.)

Какие из следующих языков поддерживают полиморфизм?

1. C++
2. Python
3. Java
4. HTML

Задание № 6. Прочитайте вопрос, запишите развернутый ответ.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, ОК 02, ОК 03, ПК 2.1., ПК 2.2.)

Объясните, что такое инкапсуляция и как она используется в объектно-ориентированном программировании

Ответ:

Задание № 7. Прочитайте вопрос, запишите развернутый ответ.
(оцениваемые знания, умения, компетенции: У1, , 32, 33, ОК 01, ОК 02, ОК 03, ПК 2.4.)

Опишите, как наследование помогает в повторном использовании кода и приведите пример его использования

Ответ:

Задание № 8. Прочитайте вопрос, запишите короткий ответ.
(оцениваемые знания, умения, компетенции: У1, У2, 31, 32, , ОК 02, ОК 03, ПК 2.1., ПК 2.2., ПК 2.6.)

Что такое объект в контексте объектно-ориентированного программирования?

Ответ:

Задание № 9. Прочитайте вопрос, запишите короткий ответ.
(оцениваемые знания, умения, компетенции: У1, У2, 31, 32, ОК 02, ОК 03, ПК 2.1., ПК 2.6.)

Назовите три основных принципа объектно-ориентированного программирования

4. Условный оператор	г. Именованная область памяти для хранения данных.
----------------------	--

Запишите ответ:

1.	
2.	
3.	
4.	

Задание № 12. Прочитайте вопрос, выберите один правильный ответ. Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 03, ПК 2.1.)

Что такое алгоритм?

1. Язык программирования
2. Последовательность действий для решения задачи
3. Тип данных
4. Математическая формула

Задание № 13. Прочитайте вопрос, выберите один правильный ответ. Обведите кружочком номер правильного ответа.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З2, З3, ОК 01, ОК 02, ПК 2.4.)

Какой тип цикла выполняется, пока условие истинно?

1. Цикл с предусловием
2. Цикл с постусловием
3. Бесконечный цикл
4. Условный оператор

Задание № 14. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.

(оцениваемые знания, умения, компетенции: У1, У2, З1, З3, ОК 01, ОК 03, ПК 2.1. ПК 2.6.)

Какие элементы являются основными в алгоритмизации?

1. Переменные
2. Комментарии
3. Условные операторы
4. Графические элементы

Задание № 15. Прочитайте вопрос, выберите несколько правильных ответов. Обведите кружочками номера правильных ответов.

(оцениваемые знания, умения, компетенции: У2, 31, 32, ОК 02, ОК 03, ПК 2.1.)

Какие типы данных используются в программировании?

1. Целочисленный
2. Алгоритмический
3. Строковый
4. Графический

Задание № 16. Прочитайте вопрос, запишите развернутый ответ.
(оцениваемые знания, умения, компетенции: У1, У2, 31, 32, 33, ОК 01, ОК 02, ПК 2.4., ПК 2.6.)

Опишите основные этапы разработки алгоритма

Ответ:

Задание № 17. Прочитайте вопрос, запишите развернутый ответ.
(оцениваемые знания, умения, компетенции: У1, У2, 32, 33, ОК 01, ОК 03, ПК 2.1., ПК 2.2., ПК 2.3., ПК 2.4.)

Какие преимущества предоставляет использование функций в программировании?

Ответ:

Ключи ответов

ВАРИАНТ 1	
Номер задания	Правильный ответ
1	1-б 2-а 3-в 4-г
2	3
3	2
4	1,3
5	1,3
6	1. Простой и понятный синтаксис. 2. Динамическая типизация. 3. Поддержка объектно-ориентированного, функционального и процедурного программирования. 4. Большое количество библиотек и frameworks. 5. Кроссплатформенность (работает на Windows, Linux, macOS). 6. Интерпретируемый язык (не требует компиляции).
7	<pre>numbers = [1, 2, 3, 4, 5] total_sum = sum(numbers) print(total_sum)</pre>
8	Метод append().
9	Кортеж (Tuple).
10	<pre>def filter_even_numbers(numbers): return [num for num in numbers if num % 2 == 0] # Пример использования input_list = [1, 2, 3, 4, 5, 6, 7, 8, 9] output_list = filter_even_numbers(input_list) print(output_list) # Вывод: [2, 4, 6, 8]</pre>
11	1-б 2-в

	3-а 4-г
12	2
13	2
14	1,3
15	1,2,4
16	<p>При импорте модуля Python выполняет следующие шаги:</p> <ol style="list-style-type: none"> 1. Поиск модуля: Python ищет модуль в директориях, указанных в <code>sys.path</code>. 2. Компиляция: Если модуль найден, он компилируется в байт-код (файл <code>.pyc</code>). 3. Выполнение: Код модуля выполняется, и все определения (функции, классы, переменные) загружаются в память. 4. Создание объекта модуля: Python создает объект модуля, который добавляется в <code>sys.modules</code>. 5. Импорт: Импортируемый модуль становится доступным в текущем пространстве имен.
17	<p>Для создания пакета в Python необходимо:</p> <ol style="list-style-type: none"> 1. Создать директорию с именем пакета. 2. Внутри директории создать файл <code>__init__.py</code>. Этот файл может быть пустым или содержать код инициализации пакета. 3. Добавить модули (файлы <code>.py</code>) в эту директорию. 4. (Опционально) Создать поддиректории для вложенных пакетов, каждая из которых также должна содержать <code>__init__.py</code>.
18	<code>__init__.py</code>
19	<code>from module_name import function_name</code>
20	<pre># math_operations.py def add(a, b): return a + b # main.py from math_operations import add result = add(5, 3) print(result) # Вывод: 8</pre>
ВАРИАНТ 2	
Номер задания	Правильный ответ
1	1-б

	2-а 3-в 4-г
2	2
3	4
4	1,2,4
5	1,2,3
6	<p>Инкапсуляция — это один из основных принципов объектно-ориентированного программирования, который заключается в объединении данных и методов, работающих с этими данными, в одном классе, а также в ограничении доступа к внутренним состояниям объекта. Это достигается с помощью модификаторов доступа, таких как <code>private</code>, <code>protected</code> и <code>public</code>.</p> <p>Как используется инкапсуляция:</p> <p>Защита данных: Инкапсуляция позволяет скрыть внутренние детали реализации класса от внешнего мира. Это предотвращает несанкционированный доступ к данным и защищает их от случайных изменений.</p> <p>Упрощение интерфейса: Пользователи класса взаимодействуют с ним через публичные методы (интерфейс), не беспокоясь о том, как реализованы внутренние механизмы.</p> <p>Поддержка изменений: Изменения в реализации класса могут быть выполнены без влияния на код, который использует этот класс, если интерфейс остается неизменным.</p>
7	<p>Наследование — это механизм в объектно-ориентированном программировании, который позволяет создавать новый класс на основе существующего. Новый класс (называемый подклассом или производным классом) наследует свойства и методы родительского класса (суперкласса), что способствует повторному использованию кода и упрощает его поддержку.</p> <p>Как наследование помогает в повторном использовании кода:</p> <p>Снижение дублирования: Вместо того чтобы повторять один и тот же код в нескольких классах, можно вынести общие свойства и методы в родительский класс.</p> <p>Расширяемость: Подклассы могут добавлять новые функции или изменять поведение, унаследованное от суперкласса, что позволяет легко расширять функциональность программы.</p> <p>Упрощение поддержки: Изменения в родительском классе автоматически применяются ко всем подклассам, что упрощает</p>

	процесс обновления и исправления ошибок.
8	Объект - это экземпляр класса, который содержит данные и методы для работы с этими данными
9	Три основных принципа: инкапсуляция, наследование, полиморфизм.
10	<p>При разработке программы для управления библиотекой можно эффективно использовать принципы объектно-ориентированного программирования (ООП) — инкапсуляцию, наследование и полиморфизм. Вот как это можно реализовать:</p> <p>1. Инкапсуляция Инкапсуляция позволяет скрыть внутренние детали реализации классов и защитить данные от несанкционированного доступа. В нашей системе мы можем создать классы для книг, читателей и библиотекарей, где данные будут защищены с помощью модификаторов доступа.</p> <p>2. Наследование Наследование позволяет создавать новые классы на основе существующих, что способствует повторному использованию кода. В нашей системе можно создать базовый класс User , от которого будут наследоваться классы Reader и Librarian.</p> <p>3. Полиморфизм Полиморфизм позволяет использовать один интерфейс для работы с разными типами объектов. В нашей системе можно создать метод, который будет принимать объект типа User , и в зависимости от типа пользователя (читатель или библиотекарь) выполнять разные действия. Таким образом, в нашей программе для управления библиотекой мы используем:</p> <p>Инкапсуляцию для защиты данных о книгах и пользователях, предоставляя доступ к ним только через методы. Наследование для создания иерархии классов пользователей, что позволяет избежать дублирования кода и упрощает его поддержку. Полиморфизм для обработки действий пользователей (читателей и библиотекарей) через общий интерфейс, что делает код более гибким и расширяемым. Эта структура позволяет легко добавлять новые функции и классы в будущем, например, классы для администраторов или для управления категориями книг.</p>
11	1-а 2-г 3-в 4-б

12	2
13	1
14	1,3
15	1,3
16	<p>Основные этапы разработки алгоритма:</p> <ol style="list-style-type: none"> 1. Постановка задачи — определение цели и условий. 2. Анализ задачи — выделение входных и выходных данных. 3. Разработка алгоритма — создание последовательности действий. 4. Реализация алгоритма — написание кода на языке программирования. 5. Тестирование и отладка — проверка корректности работы алгоритма.
17	<p>Использование функций позволяет:</p> <ol style="list-style-type: none"> 1. Упростить код за счет повторного использования. 2. Улучшить читаемость программы. 3. Упростить тестирование и отладку. 4. Разделить программу на логические блоки.
18	Цикл с предусловием и цикл с постусловием.
19	Именованная область памяти для хранения данных.
20	<ol style="list-style-type: none"> 1. Начало алгоритма. 2. Ввод числа N. 3. Инициализация переменной $sum = 0$. 4. Запуск цикла от 1 до N: Если текущее число четное, добавить его к sum. 5. Вывод sum. 6. Конец алгоритма <p>$sum = 0$ Ввод N Для i от 1 до N: Если $i \% 2 == 0$: $sum = sum + i$ Вывод sum</p>

Критерии оценивания ответов, полученных в ходе тестирования

За каждый верный ответ выставляется 1 балл, за неверный ответ – 0 баллов. Баллы, полученные обучающимися за выполненные задания, суммируются.

Результаты тестирования определяются в разрезе каждого обучающегося в баллах и оценках.

Результаты тестирования			
Баллы	Оценка	Доля выполненных заданий	Уровень сформированности компетенций
0-5 баллов	2 (неудовлетворительно)	0-25%	низкий
6-10 баллов	3 (удовлетворительно)	30-50%	базовый
11-15 баллов	4 (хорошо)	55-75%	повышенный
16-20 баллов	5 (отлично)	80-100%	высокий

4. Информационное обеспечение

перечень учебных изданий, электронных изданий, электронных и Интернет ресурсов, образовательных платформ, электронно-библиотечных систем, веб систем для организации дистанционного обучения и управления им, используемые в образовательном процессе как основные и дополнительные источники.

Основные источники:

1. Алгоритмизация и программирование, учебное пособие/ Канцедал СА. - М.: ИД ФОРУМ,2017-352 с.
2. Черпаков ИВ. Основы программирования: Учебник и практикум для СПО- М.: Юрайт,2017.-219 с.
3. Основы алгоритмизации и программирования (1-е изд.) учебник [Семакин И.Г. — М.:ИД Академия,2017 — 304 с.
4. Черпаков ИВ. Основы программирования: Учебник и практикум для СПО- М.: Юрайт,2018.-219 с.
5. Основы алгоритмизации и программирования: учебное пособие / О. Л.Голицына, И. И.Попов.- М.: Форум, 2020 -431 с.

Дополнительные источники:

1. Богомазова Г.Н. Модернизация программного обеспечения персональных компьютеров, серверов, периферийных устройств и оборудования: учебник. — М.: ИЦ «Академия», 2016. — 192 с.
2. Голицына О.Л. Языки программирования: учебное пособие / О.Л. Голицына, Т.Л. Партыка, И.И. Попов. — 3-е изд., перераб. и доп. — М.: ФОРУМ; ШОРА-М, 2013. 400 с.: ил, — (Профессиональное образование).
3. Канцедал СА. Алгоритмизация и программирование: учебное пособие М.: ИД «ФОРУМ»: РЖФРА-М, 2010.
4. Кормен Т. Алгоритмы: вводный курс.: Пер. с англ. — М.: ООО «ИД. Вильямс», 2014.
5. Моргун А.Н. Программирование на языке Pascal. Основы обработки структур данных. — М.: «Вильямс», 2006.
6. Рыбальченко МВ. Архитектура информационных систем. Учебное пособие для СПО / МВ. Рыбальченко — М. ИЦ «Юрайт», 2017 — 91 с.

7. Рапаков ГГ. Turbo Pascal для студентов и школьников / ГГ. Рапаков, С.Ю. Ржеуцкая. - СМ.: БХВ-Петербург, 2013.-352 с.: ил.

8. Семакин И.Г., Шестаков А.П. / Основы алгоритмизации и программирования. Практикум: учебн. пособие для студ. учреждений СПО — М.: Издательский центр «Академия», 2014.

Электронные издания (электронные ресурсы):

CNews [Электронный ресурс] / Официальный сайт периодического 90 издания — журнал «CNews». Режим доступа: <http://www.cnews.ru>, свободный.

Computerworld — Россия [Электронный ресурс] / Официальный сайт периодического издания международный компьютерный журнал 10.

«Computerworld». Режим доступа: <http://www.computerworld.ru>, свободный.

PascalABC.NET [Электронный ресурс] Современное программирование на языке Паскаль. Режим доступа: <http://pascalabc.net>, свободный.

VisualBasic.NET [Электронный ресурс] / Мультимедийный Обучающий Курс. — Режим доступа: <http://gratisfl1e.com/publ/140-1-О-13>, свободный.

Википедия — свободная энциклопедия [Электронный ресурс] / Сайт международного информационного ресурса «Википедия» Режим доступа: <http://ru.wikipedia.org>, свободный.

Компьютерные видео уроки. [Электронный ресурс] / Компьютерные видео уроки по программированию. Режим доступа: <http://compteacher.ru/programming>, свободный.

Мир ПК [Электронный ресурс] / Официальный сайт периодического издания журнал «Мир ПК». Режим доступа: <http://www.osp.ru/pcworld/#/home>, свободный.

Открытые системы. СУБД [Электронный ресурс] / Официальный сайт периодического издания — журнал «Открытые системы. СУБД».— Режим доступа: <http://www.osp.ru/os/#/home>, свободный.

Программные продукты и системы [Электронный ресурс] / 17. Официальный сайт периодического издания — журнал «Программные продукты и системы». — Режим доступа: <http://www.swsys.ru>, свободный. Федеральный центр информационно-образовательных ресурсов[Электронный ресурс] / Информационный сайт. Режим доступа: <http://fcior.edu.ru>.

Полное руководство по языку программирования C# 9.0 и платформе 19. .NET 5: <https://metanit.com/sharp/tutorial>, свободный.

Цифровая образовательная среда СПО ProfОбразование.:

Алексеев, Е. Р. Free Pascal и Lazarus : учебник по программированию / Е. Р. Алексеев, О. В. Чеснокова, Т. В. Кучер. — 2-е изд. — Саратов .

Профобразование, 2019. — 438 с. — ISBN 978-5-4488-0105-1. — текст .
электронный // Электронный ресурс цифровой образовательной среды СПО

PROF06pa30Bawre : [сайт]. — URL: <https://profspo.ru/books/87979> (дата обращения: 04.09.2020). — Режим доступа: для авторизир.пользователей

Златопольский, Д. М. Программирование: типовые задачи, алгоритмы, методы / Д. М. Златопольский. — 4-е изд. — Москва : Лаборатория знаний, 2020. — 224 с. — ISBN 978-5-00101-789-9. Текст электронный

// Электронный ресурс цифровой образовательной среды СПО PROF06pa30Barme : [сайт]. — URL: <https://profspo.ru/books/12264> (дата обращения: 03.09.2020). Режим доступа: для авторизир.пользователей

Кудинов, Ю. И. Основы алгоритмизации и программирования : учебное пособие для СПО / Ю. И. Кудинов, А. Ю. Келина. — 2-е изд. —

Липецк, Саратов : Липецкий государственный технический университет, Профобразование, 2020. 71 с. — ISBN 978-5-88247-956-4, 978-5-44880757-2. Текст электронный // Электронный ресурс цифровой образовательной среды СПО PROF06pa30Barme [сайт]. — URL: <https://profspo.ru/books/92834> (дата обращения: 07.09.2020). — Режим доступа: для авторизир. пользователей

Лубашева, Т. В. Основы алгоритмизации и программирования : учебное пособие / Т. В. Лубашева, Б. А. Железко. — Минск : Республиканский институт профессионального образования (РПШО), 2016. — 379 с. — ISBN 978-985-503-625-9. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО : [сайт]. — URL: <https://profspo.ru/books/67689> (дата обращения: 06.09.2020). Режим доступа: для авторизир. пользователей

Мухаметзянов, Р. Р. Основы программирования в Delphi: учебнометодическое пособие / Р. Р. Мухаметзянов. Набережные Челны: Набережночелнинский государственный педагогический университет, 2017. 137 с. ISBN 2227-8397. Текст электронный // Электронный ресурс цифровой образовательной среды СПО PROF06pa30Barme : [сайт]. — URL: <https://profspo.ru/books/66811> (дата обращения: 05.09.2020). Режим доступа: для авторизир.пользователей

Токманцев, Т. Б. Алгоритмические языки и программирование : учебное пособие для СПО / Т. Б. Токманцев ; под редакцией В. Б. Костоусова.—

2-е изд.Саратов, Екатеринбург : Профобразование, Уральский федеральный университет, 2019. — 102 с. — ISBN 978-5-4488-05103, 978-5-7996-2899-4. Текст электронный // Электронный ресурс цифровой образовательной среды СПО РКО(н)образование : [сайт]. — URL: <https://profspo.ru/books/87785> (дата обращения: 01.09.2020).— Режим доступа: для авторизир. Пользователей

Уйманова, н. Основы объектно-ориентированного программирования : практикум для СПО / Н, А. Уйманова, М. Г. Таспаева. — Саратов : Профобразование, 2019. — 155 с. — 978-526.4488-0352-9. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО [сайт]. — URL: <https://profspo.ru/books/86199> (дата обращения: 04.09.2020). — Режим доступа: для авторизир. пользователей

Устинов, В. В. Основы алгоритмизации и программирование. Часть 2 : конспект лекций / В. В. Устинов. — Новосибирск : Новосибирский

государственный технический университет, 2013. —32 с. — ISBN 978275-7782-2337-0. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО: [сайт]. — URL: <https://profspo.ru/books/44675> (дата обращения: 06.09.2020). Режим доступа: для авторизир. пользователей

Цифровая образовательная среда СПО PROF0бра30Barme:

Устинов, В. В. Основы алгоритмизации и программирования. Часть 1 : конспект лекций / В. В. Устинов. Новосибирск : Новосибирский государственный технический университет, 2010. 40 с. — ISBN 978285-7782-1366-1. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО PROFобразование [сайт]. — URL: <https://profspo.ru/books/44676> (дата обращения: 07.09.2020). — Режим доступа: для авторизир. пользователей

Чурина, Т. Г. Основы алгоритмизации и программирования : учебное пособие для СПО / Т. Г. Чурина, Т. В. Нестеренко. — Саратов, Москва : Профобразование, Ай Пи Ар Медиа, 2020. — 214 с. — ISBN 978-5-448829-0802-9, 978-5-4497-0465-8. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО [сайт]. URL: <https://profspo.ru/books/96017> (дата обращения: 03.09.2020). — Режим доступа: для авторизир. пользователей

Электронно-библиотечная система:

IPR BOOKS - <http://www.iprbookshop.ru/78574.html>

Веб-система для организации дистанционного обучения и управления им:

Система дистанционного обучения ОГАПОУ «Алексеевский колледж»
<http://moodle.alcollege.ru/>