

Документ подписан Министерством науки и высшего образования Российской Федерации
Информация о владельце:
ФИО: Макаренко Елена Николаевна
Должность: Декан
Федеральное государственное бюджетное образовательное учреждение высшего образования «Ростовский государственный экономический университет (РИНХ)»
Дата подписания: 17.06.2026 13:24:14 Финансово-экономический колледж
Уникальный программный ключ:
c098bc0c1041cb2a4cf926cf171d6715d99a6ae00adc8e27b55cbe1e2dbd7c78



УТВЕРЖДАЮ

Директор

Р. А. Сычев

2026г.

Рабочая программа дисциплины Основы алгоритмизации и программирования

Специальность
09.02.12 ТЕХНИЧЕСКАЯ ЭКСПЛУАТАЦИЯ И СОПРОВОЖДЕНИЕ
ИНФОРМАЦИОННЫХ СИСТЕМ

Форма обучения	очная
Часов по учебному плану	96
в том числе:	
аудиторные занятия	48
самостоятельная работа	42

Ростов-на-Дону
2026 г.

Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	3 (2.1)		Итого	
	УП	РП	УП	РП
Неделя	16			
Вид занятий	УП	РП	УП	РП
Лекции	16	16	16	16
Практические	32	32	32	32
Итого ауд.	48	48	48	48
Контактная работа	48	48	48	48
Сам. работа	42	42	42	42
Часы на контроль	6	6	6	6
Итого	96	96	96	96

ОСНОВАНИЕ

Федеральный государственный образовательный стандарт среднего профессионального образования по специальности 09.02.12 ТЕХНИЧЕСКАЯ ЭКСПЛУАТАЦИЯ И СОПРОВОЖДЕНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ (приказ Министерства просвещения Российской Федерации от 10.03.2025 г. № 184)

Рабочая программа составлена по образовательной программе 09.02.12 ТЕХНИЧЕСКАЯ ЭКСПЛУАТАЦИЯ И СОПРОВОЖДЕНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ для набора 2026 года
программа среднего профессионального образования

Учебный план утвержден учёным советом вуза от 03.03.2026 протокол № 9

Программу составил(и): Преп., Полуян О.С.

Председатель ЦМК: Ламин В. А.

Рассмотрено на заседании ЦМК от 06.03.2026 протокол № 7

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ	
1.1	Формирование навыков работы в среде программирования, разработки алгоритмов для решения конкретных задач, реализации готовых и разработанных алгоритмов на выбранном языке программирования.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ	
Цикл (раздел) ООП:	ОП
2.1	Требования к предварительной подготовке обучающегося:
2.1.1	Информатика
2.2	Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:
2.2.1	Математическое моделирование
2.2.2	Настройка и обеспечение работоспособности программных и аппаратных средств устройств инфокоммуникационных систем
2.2.3	Проектирование и разработка информационных систем
2.2.4	Разработка информационных систем
2.2.5	Тестирование и эксплуатация информационных систем

3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ	
3.1 Знать	
ОК.01: Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	
<ul style="list-style-type: none"> - актуальный профессиональный и социальный контекст, в котором приходится работать и жить; - структуру плана для решения задач, алгоритмы выполнения работ в профессиональной и смежных областях; - основные источники информации и ресурсы для решения задач и/или проблем в профессиональном и/или социальном контексте; - методы работы в профессиональной и смежных сферах; - порядок оценки результатов решения задач профессиональной деятельности; 	
ОК.02: Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	
<ul style="list-style-type: none"> - номенклатуру информационных источников, применяемых в профессиональной деятельности; - приемы структурирования информации; формат оформления результатов поиска информации 	
ОК.09: Пользоваться профессиональной документацией на государственном и иностранном языках	
<ul style="list-style-type: none"> - правила построения простых и сложных предложений на профессиональные темы 	
3.2 Уметь	
ОК.01: Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	
<ul style="list-style-type: none"> - распознавать задачу и/или проблему в профессиональном и/или социальном контексте, анализировать и выделять её составные части; - определять этапы решения задачи, составлять план действия, реализовывать составленный план, определять необходимые ресурсы; - выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы; - оценивать результат и последствия своих действий (самостоятельно или с помощью наставника); 	
ОК.02: Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	

<p>- осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности</p> <p>ОК.09: Пользоваться профессиональной документацией на государственном и иностранном языках</p> <p>- понимать тексты на базовые профессиональные темы на государственном и иностранном языках</p>
<p>3.3 Владеть</p> <p>ОК.01: Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам</p> <p>- навыками выявления и эффективного поиска информации, необходимой для решения задачи и/или проблемы;</p> <p>ОК.02: Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности</p> <p>- номенклатурой информационных источников, применяемых в профессиональной деятельности, а также приемами структурирования информации и форматом оформления результатов поиска информации</p> <p>ОК.09: Пользоваться профессиональной документацией на государственном и иностранном языках</p> <p>- правилами чтения текстов на государственном и иностранном языках</p>

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)						
Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Компетенции	Литература	Примечание
	Раздел 1. Тема 1.1. Основы алгоритмизации, языки и системы программирования.					
1.1	Алгоритм и его свойства. Способы записи алгоритмов. Эволюция и классификация языков программирования. /Лек/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.2	Типы данных. Операторы языка. Линейное программирование и разветвляющееся. /Лек/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.3	Составление программ линейной и разветвляющейся структуры. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.4	Циклические программы. Одномерные массивы. Двумерные массивы. Работа с массивами. Сортировка массивов. /Лек/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.5	Составление программ циклической структуры. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.6	Составление программ для обработки одномерных и двумерных массивов. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.7	Строки. Коллекции. /Лек/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.8	Символы и строки. обработка строк и использование коллекций. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	

1.9	Файлы. Доступ к файлам. Виды файлов. Считывание и запись в файл. /Лек/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.10	Работа с файлами. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.11	Составление программ линейной и разветвляющейся структуры. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.12	Составление программ линейной и разветвляющейся структуры. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.13	Составление программ циклической структуры. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.14	Составление программ циклической структуры. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.15	Составление программ для обработки одномерных и двумерных массивов. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.16	Составление программ для обработки одномерных и двумерных массивов. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.17	Символы и строки. обработка строк и использование коллекций. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.18	Символы и строки. обработка строк и использование коллекций. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.19	Работа с файлами. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
1.20	Работа с файлами. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
	Раздел 2. Тема 2.1 Модульное программирование.					
2.1	Модульное программирование Локальные и глобальные переменные Подпрограммы. Модификаторы. Передача данных в подпрограммы. Рекурсия. Разработка рекурсивных подпрограмм. /Лек/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
2.2	Разработка кода для создания модулей. Использование рекурсии. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
2.3	Использование подпрограмм в разработке кода. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
2.4	Использование подпрограмм в разработке кода. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	

2.5	Разработка кода для создания модулей. Использование рекурсии. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
2.6	Разработка кода для создания модулей. Использование рекурсии. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
2.7	Использование подпрограмм в разработке кода. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
2.8	Использование подпрограмм в разработке кода. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
Раздел 3. Тема 2.2 Основные принципы объектно-ориентированного программирования.						
3.1	Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс Основные принципы ООП: инкапсуляция, наследование, полиморфизм. /Лек/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
3.2	Работа с классами: объявление класса, создание экземпляров, конструктор, управление памятью, методы. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
3.3	Инкапсуляция: свойства, сокрытие реализации. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
3.4	Наследование: одиночное и множественное. Переопределение методов. Вызов родительского конструктора. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
3.5	Наследование: одиночное и множественное. Переопределение методов. Вызов родительского конструктора. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
3.6	Полиморфизм: перегрузка методов, виртуальные методы, абстрактные классы и методы. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
3.7	Полиморфизм: перегрузка методов, виртуальные методы, абстрактные классы и методы. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
3.8	Работа с классами: объявление класса, создание экземпляров, конструктор, управление памятью, методы. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
3.9	Работа с классами: объявление класса, создание экземпляров, конструктор, управление памятью, методы. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	

3.10	Наследование: одиночное и множественное. Переопределение методов. Вызов родительского конструктора. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
3.11	Наследование: одиночное и множественное. Переопределение методов. Вызов родительского конструктора. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
	Раздел 4. Тема 3.1 Этапы разработки приложений.					
4.1	Визуально-событийно управляемое программирование. Разработка приложения. Проектирование объектно-ориентированного приложения. Создание интерфейса пользователя. Тестирование, отладка приложения. Оптимизация программы. /Лек/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
4.2	Создание проекта с использованием компонентов для работы с текстом и кнопочными компонентами, переключателями. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
4.3	Создание проекта с использованием компонентов для отображения таблиц, компонентов для отображения дат и времени. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
4.4	Разработка интерфейса приложения. Тестирование приложения. /Пр/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
4.5	Создание проекта с использованием компонентов для работы с текстом и кнопочными компонентами, переключателями. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
4.6	Создание проекта с использованием компонентов для отображения таблиц, компонентов для отображения дат и времени. /Ср/	3	2	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	
4.8	Экзамен .	3	6	ОК 01., ОК 02., ОК 09.	Л1.1Л2.1 Л2.2 Э1, Э2, Э3	

5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

5.1. Фонд оценочных средств для проведения промежуточной аттестации

Промежуточная аттестация проходит в форме экзамена. Перечень вопросов к экзамену:
1. Что такое матрица? Приведите примеры матриц разных размеров.

2. Какие виды матриц вы знаете? Дайте краткую характеристику каждому виду.
3. Опишите правила сложения и вычитания матриц. При каких условиях эти операции возможны?
4. Как выполняется умножение матрицы на число? Приведите пример.
5. Изложите правила умножения матриц. В чём особенность этой операции?
6. Что такое определитель матрицы? Для каких матриц он может быть вычислен?
7. Как вычислить определитель матрицы второго порядка? Приведите формулу и пример.
8. Опишите способы вычисления определителя матрицы третьего порядка.
9. Перечислите основные свойства определителей. Приведите примеры их использования.
10. Что такое обратная матрица? Сформулируйте условие её существования.
11. Опишите алгоритм нахождения обратной матрицы.
12. Что такое ранг матрицы? Какими способами его можно определить?
13. Дайте определение системы линейных уравнений. Что значит решить систему?
14. В чём суть метода Крамера для решения систем линейных уравнений? При каких условиях он применим?
15. Запишите формулы Крамера для системы трёх уравнений с тремя неизвестными.
16. Изложите алгоритм решения системы линейных уравнений методом Гаусса.
17. Как решить систему линейных уравнений матричным методом? Приведите формулу.
18. Дайте определение вектора. Какие бывают векторы?
19. Опишите основные операции над векторами (сложение, вычитание, умножение на число) и их свойства.
20. Что такое скалярное произведение векторов? Как его вычислить в координатной форме?
21. Каков геометрический смысл скалярного произведения? Как с его помощью найти угол между векторами?
22. Что такое векторное произведение векторов? Каковы его свойства?
23. Каков геометрический смысл векторного произведения? Как его использовать для нахождения площади параллелограмма?
24. Что такое смешанное произведение трёх векторов? Как оно вычисляется?
25. Каков геометрический смысл смешанного произведения? Как его применить для вычисления объёма параллелепипеда?
26. Дайте определение комплексного числа. Какие формы записи комплексных чисел существуют?
27. Как записать комплексное число в алгебраической форме? Приведите пример.
28. Опишите тригонометрическую форму комплексного числа. Как перейти от алгебраической формы к тригонометрической?
29. Что такое показательная форма комплексного числа? Как её получить из тригонометрической формы?
30. Как изобразить комплексное число на комплексной плоскости? Что такое модуль и аргумент комплексного числа?
31. Дайте определение высказывания в математической логике. Чем отличаются простые и сложные высказывания?
32. Перечислите основные логические операции (конъюнкция, дизъюнкция, импликация, эквиваленция, отрицание). Укажите их обозначения.
33. Что такое таблица истинности? Опишите пошаговый алгоритм её построения.
34. Приведите пример формулы логики и постройте для неё таблицу истинности.
35. Сформулируйте основные законы логики (закон двойного отрицания, законы де Моргана, закон исключённого третьего и др.).
36. Что такое равносильные преобразования логических формул? Приведите 2–3 примера.
37. Дайте общее определение множества. Какими способами можно задать множество?
38. Перечислите основные операции над множествами (объединение, пересечение, разность, дополнение) и опишите их свойства.
39. Как изобразить множества и операции над ними с помощью диаграмм Эйлера Венна? Приведите примеры для каждой операции.
40. Что такое декартово произведение множеств? Как оно определяется и как изображается?
41. Дайте определение графа. Какие элементы входят в его структуру?
42. Чем отличаются ориентированные и неориентированные графы? Приведите примеры.

43. Какими способами можно задать граф? Опишите каждый способ.
44. Что такое матрица смежности графа? Как её построить для ориентированного и неориентированного графа?
45. Что такое матрица инцидентий графа? В чём её отличие от матрицы смежности?
46. Дайте определения эйлерова и гамильтонова графа. Приведите примеры таких графов и объясните, как проверить граф на эйлеровость и гамильтоновость.
47. Что такое дерево в теории графов? Каковы его основные свойства? Приведите пример дерева.
48. Дайте определение предела функции в точке. Сформулируйте основные свойства пределов.
49. Что такое замечательные пределы? Запишите их формулы и приведите примеры применения.
50. Опишите методы раскрытия неопределённостей вида $\frac{0}{0}$ и $\frac{\infty}{\infty}$.
51. Что такое односторонние пределы? Как они связаны с классификацией точек разрыва функции?
52. Дайте определение производной функции в точке. Каков её геометрический и физический смысл?
53. Что такое производные и дифференциалы высших порядков? Как их вычислять? Приведите пример.
54. Опишите алгоритм полного исследования функции (область определения, чётность/нечётность, периодичность, асимптоты, экстремумы, интервалы монотонности и выпуклости, точки перегиба).
55. Как построить график функции на основе её полного исследования? Приведите пример исследования и построения графика.
56. Дайте определение предела и непрерывности функции нескольких переменных.
57. Что такое частные производные? Как их находить? Приведите пример вычисления частных производных для функции двух переменных.
58. Что такое производные и дифференциалы высших порядков для функций нескольких переменных? Как их вычислить?
59. Дайте определение неопределённого интеграла. Перечислите его основные свойства.
60. Что такое определённый интеграл? Сформулируйте его свойства и геометрический смысл.
61. Опишите методы вычисления определённых интегралов (замена переменной, интегрирование по частям). Приведите примеры.
62. Что такое несобственный интеграл с бесконечными пределами интегрирования? Как исследовать его сходимость?
63. Дайте определение двойного интеграла. Каковы его свойства?
64. Что такое повторные интегралы? Как с их помощью вычислять двойные интегралы? Приведите пример.
65. Перечислите элементы комбинаторики (размещение, перестановка, сочетание). Приведите формулы для их вычисления и примеры задач.
66. Что такое случайное событие? Как определить вероятность события?
67. Сформулируйте формулы сложения и умножения вероятностей. Приведите примеры их применения.
68. Что такое условная вероятность? Как она вычисляется? Приведите пример задачи на условную вероятность.
69. Сформулируйте формулу полной вероятности. Приведите пример её использования.
70. Дайте определение дискретной и непрерывной случайной величины. Перечислите их основные характеристики (математическое ожидание, дисперсия, среднее квадратичное отклонение).

Критерии оценивания:

5 баллов выставляется студентам за полный и правильный ответ на все вопросы билета с логическим обоснованием аргументов, в ответе нет ошибок.

4 балла выставляется студентам, если вопросы билета раскрыты полностью, но обоснования доказательства недостаточны, при этом допущены две-три несущественные ошибки, исправленные по требованию преподавателя.

3 балла ставится студентам за правильный ответ на вопросы билета, при этом допущено более одной ошибки по изложению фактов или более двух-трёх недочетов в ответе.

2 балла ставится студентам, если допущены существенные ошибки, показавшие, что обучающийся не обладает обязательными умениями по данной теме в полной мере.

5.2. Фонд оценочных средств для проведения текущего контроля

Представлен в Приложении 1 к рабочей программе дисциплины.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

6.1. Рекомендуемая литература

6.1.1 Основная литература

	Авторы,	Заглавие	Издательство,	Колич-во
Л1.1	Чернышев С.А.	Основы программирования на Python	Москва, Юрайт, 2026	https://urait.ru/book/osnovy-programmirovaniya-na-python-588671 неограниченный доступ для зарегистрированных

6.1.2 Дополнительная литература

	Авторы,	Заглавие	Издательство,	Колич-во
Л2.1	Черпаков И.В.	Алгоритмизация и программирование в Python	Москва, Юрайт, 2026	https://urait.ru/book/algoritmizaciya-i-programmirovanie-v-python-582413 неограниченный доступ для зарегистрированных пользователей
Л2.2	Федоров Д.Ю.	Программирование на Python	Москва, Юрайт, 2026	https://urait.ru/book/programmirovanie-na-python-585807 неограниченный доступ для зарегистрированных пользователей

6.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	Единое окно доступа к образовательным ресурсам
Э2	Учебный курс – Инструменты, алгоритмы и структуры данных
Э3	Федеральный центр информационно-образовательных ресурсов

6.3. Перечень программного обеспечения

6.3.1	Офисный пакет - LibreOffice
6.4 Перечень информационных справочных систем	
6.4.1	ИСС «КонсультантПлюс»
6.4.2	ИСС «Гарант»

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

7.1	Помещения для проведения всех видов работ, предусмотренных учебным планом, укомплектованы необходимой специализированной учебной мебелью и техническими средствами обучения
-----	---

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ

Методические указания по освоению дисциплины представлены в Приложении 2 к рабочей программе

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ОП.07 Основы алгоритмизации и программирования

1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

1.1 Показатели и критерии оценивания компетенций:

УУД, составляющие компетенцию	Показатели оценивания	Критерии оценивания	Средства оценивания
ОК. 01: Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам.			
<p>Знать:</p> <ul style="list-style-type: none"> - актуальный профессиональный и социальный контекст, в котором приходится работать и жить; - структуру плана для решения задач, алгоритмы выполнения работ в профессиональной и смежных областях; - основные источники информации и ресурсы для решения задач и/или проблем в профессиональном и/или социальном контексте; - методы работы в профессиональной и смежных сферах; - порядок оценки результатов решения задач профессиональной деятельности; 	<p>Получение систематических знаний об(о)</p> <p>об основных источниках информации и ресурсах для решения задач и проблем в профессиональном и/или социальном контексте</p>	<p>Уровень знания</p> <p>основные источники информации и ресурсы для решения задач и проблем в профессиональном и/или социальном контексте</p>	<p>Т(1-41), ПЗ (1-49)</p>
<p>Уметь:</p> <ul style="list-style-type: none"> - распознавать задачу и/или проблему в профессиональном и/или социальном контексте, анализировать и выделять её составные части; - определять этапы решения задачи, составлять план действия, реализовывать составленный план, определять необходимые ресурсы; - выявлять и эффективно искать информацию, 	<p>Сформировавшиеся систематические умения</p> <p>- распознавать задачу и/или проблему в профессиональном и/или социальном контексте</p>	<p>Уровень умения</p> <ul style="list-style-type: none"> - анализировать задачу и/или проблему и выделять её составные части; -определять этапы решения задачи; - выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы 	<p>Т(1-41), ПЗ (1-49)</p>

<p>необходимую для решения задачи и/или проблемы;</p> <p>- оценивать результат и последствия своих действий (самостоятельно или с помощью наставника);</p>			
<p>Владеть:</p> <p>- навыками выявления и эффективного поиска информации, необходимой для решения задачи и/или проблемы;</p>	<p>Сформировавшиеся систематические владения</p> <p>актуальными методами работы в профессиональной и смежных сферах</p>	<p>Уровень владения о</p> <p>реализовать составленный план; оценивать результат и последствия своих действий (самостоятельно или с помощью наставника)</p>	<p>Т(1-41), ПЗ (1-49)</p>

ОК. 02: Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности

<p>Знать:</p> <p>- номенклатуру информационных источников, применяемых в профессиональной деятельности;</p> <p>- приемы структурирования информации; формат оформления результатов поиска информации</p>	<p>Получение систематических знаний об(о)</p> <p>о приемах структурирования информации; формате оформления результатов поиска информации</p>	<p>Уровень знания</p> <p>основных понятий об автоматизированной обработке информации, общем составе и структуре персональных компьютеров и вычислительных систем, базовых системных программных продуктов и пакетов прикладных программ в области профессиональной деятельности</p>	<p>Т(1-41), ПЗ (1-49)</p>
<p>Уметь:</p> <p>- осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности</p>	<p>Сформировавшиеся систематические умения</p> <p>использовать технологии сбора, размещения, хранения, накопления, преобразования и передачи данных в профессионально ориентированных информационных системах, выделять профессионально-значимую профессиональную информацию, использовать основные методы и приемы обеспечения</p>	<p>Уровень умения</p> <p>определять задачи для поиска информации; структурировать получаемой информации; оценивать практическую значимость результатов поиска; оформлять результатов поиска</p>	<p>Т(1-41), ПЗ (1-49)</p>

	информационной безопасности		
Владеть: - номенклатурой информационных источников, применяемых в профессиональной деятельности, а также приемами структурирования информации и форматом оформления результатов поиска информации	Сформировавшиеся систематические владения технологиями сбора, размещения, хранения, накопления, преобразования и передачи данных в профессионально ориентированных информационных системах	Уровень владения выделять профессионально-значимую профессиональную информацию, использовать основные методы и приемы обеспечения информационной безопасности	Т(1-41), ПЗ (1-49)
ОК. 09.: Пользоваться профессиональной информацией на государственном и иностранном языках			
Знать: - правила построения простых и сложных предложений на профессиональные темы	Получение систематических знаний об(о) основные общеупотребительные глаголы (бытовая и профессиональная лексика); лексический минимум, относящийся к описанию предметов, средств и процессов профессиональной деятельности на государственном и иностранном языках	Уровень знания правил чтения текстов и составления профессиональной направленности на государственном и иностранном языках	Т(1-41), ПЗ (1-49)
Уметь: - понимать тексты на базовые профессиональные темы на государственном и иностранном языках	Сформировавшиеся систематические умения участвовать в диалогах на знакомые общие и профессиональные темы; строить простые высказывания о себе и о своей профессиональной деятельности	Уровень умения писать простые связные тексты документов, изучать и пользоваться профессиональной документацией на государственном и иностранном языках	Т(1-41), ПЗ (1-49)
Владеть: - правилами чтения текстов на государственном и иностранном языках	Сформировавшиеся систематические владения правилами и стандартами составления профессиональной документации на государственном и иностранном языках	Уровень владения на высокопрофессиональном уровне правилами и стандартами составления профессиональной документации на государственном и иностранном языках	Т(1-41), ПЗ (1-49)

2 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Тестовые задания:

1. Какое свойство алгоритма означает, что он должен приводить к правильному результату для всех допустимых входных данных?
 - a) Массовость
 - b) Результативность (конечность)
 - c) Определённость (детерминированность)
 - d) Дискретность
2. Какой способ записи алгоритма использует специальные графические символы (блоки) для обозначения действий?
 - a) Словесно-формульный
 - b) Программный (на языке программирования)
 - c) Псевдокод
 - d) Блок-схема
3. Что характеризует временная сложность алгоритма?
 - a) Количество используемой дополнительной памяти
 - b) Количество элементарных операций в зависимости от размера входных данных
 - c) Количество строк исходного кода
 - d) Максимальное значение входных данных
4. Какой вид алгоритма предполагает повторение некоторой последовательности действий?
 - a) Линейный
 - b) Разветвляющийся
 - c) Циклический
 - d) Вспомогательный
5. Что делает компилятор?
 - a) Выполняет программу строка за строкой
 - b) Преобразует всю программу в машинный код целиком до её выполнения
 - c) Редактирует исходный код программы
 - d) Управляет памятью во время выполнения программы
6. Какой язык программирования относится к языкам высокого уровня?
 - a) Ассемблер
 - b) Машинный код
 - c) Python
 - d) Бинарный код
7. Что такое среда программирования (IDE)?
 - a) Только текстовый редактор для написания кода
 - b) Комплекс программных средств для разработки, отладки и компиляции программ
 - c) Операционная система, предназначенная для программистов
 - d) База данных готовых программных кодов
8. Какой тип данных относится к простым (скалярным) типам?
 - a) Массив
 - b) Запись (структура)
 - c) Целый (integer)
 - d) Стек
9. Какой тип данных является структурированным?
 - a) Логический (boolean)
 - b) Символьный (char)
 - c) Массив (array)
 - d) Вещественный (float)
10. Что такое производный тип данных?

- a) Тип, созданный на основе стандартных типов (например, перечисление)
 - b) Тип, который нельзя изменить после объявления
 - c) Тип, используемый только в математических вычислениях
 - d) Тип, занимающий 1 байт памяти
11. Какой оператор используется для организации ветвления в программе?
- a) for
 - b) if-else
 - c) while
 - d) assignment (:= или =)
12. Для чего служит конструкция try-catch (или try-except)?
- a) Для объявления переменных
 - b) Для обработки исключительных ситуаций (ошибок времени выполнения)
 - c) Для организации циклов
 - d) Для ввода данных с клавиатуры
13. Дан фрагмент кода на псевдоязыке:
 ввод x
 если $x > 0$ то $y = 1$ иначе $y = -1$
 вывод y
 Что будет выведено при $x = -5$?
- a) 1
 - b) -1
 - c) 0
 - d) Ошибка
14. Какой цикл выполняется хотя бы один раз независимо от условия?
- a) for (со счётчиком)
 - b) while (с предусловием)
 - c) repeat-until (с постусловием)
 - d) бесконечный цикл
15. Сколько раз выполнится тело цикла?
- ```
i = 1
while i < 5:
 i = i + 1
```
- a) 4
  - b) 5
  - c) 3
  - d) 0
16. Какой индекс первого элемента в большинстве языков программирования (C, C++, Java, C#)?
- a) 1
  - b) 0
  - c) -1
  - d) Определяется программистом
17. Дан двумерный массив размером  $3 \times 4$ . Сколько всего элементов он содержит?
- a) 7
  - b) 12
  - c) 3
  - d) 4
18. Какой алгоритм сортировки многократно проходит по массиву и меняет местами соседние элементы, если они стоят в неправильном порядке?
- a) Быстрая сортировка
  - b) Сортировка вставками
  - c) Пузырьковая сортировка
  - d) Сортировка выбором
19. Дан фрагмент кода для одномерного массива A из 5 элементов:  
 $s = 0$   
 для i от 1 до 5:  
 $s = s + A[i]$

- Что вычисляется в этом фрагменте?
- Количество положительных элементов
  - Сумма всех элементов
  - Максимальный элемент
  - Среднее арифметическое
20. Какая операция над строками означает их соединение (склеивание)?
- Конкатенация
  - Трансляция
  - Итерация
  - Декомпозиция
21. Что из перечисленного является коллекцией в языках программирования (C#, Java, Python)?
- Целочисленная переменная
  - Список (List)
  - Вещественная константа
  - Логическое значение
22. Чем список (List) отличается от массива?
- Список не может хранить числа
  - Список может динамически изменять свой размер
  - Список всегда отсортирован
  - Массив занимает меньше памяти
23. Какой режим открытия файла используется для добавления данных в конец существующего файла?
- read (r)
  - write (w)
  - append (a)
  - execute (x)
24. Какой тип файла хранит данные в виде последовательности символов, которые можно открыть и прочитать в блокноте?
- Двоичный (бинарный) файл
  - Текстовый файл
  - Исполняемый файл
  - Архив
25. Что произойдет, если открыть существующий файл в режиме записи ("w") без использования режима добавления?
- Данные будут добавлены в конец файла
  - Файл будет очищен (его содержимое удалено)
  - Возникнет ошибка доступа
  - Файл будет переименован
26. Что такое локальная переменная?
- Переменная, доступная из любого модуля программы
  - Переменная, объявленная внутри подпрограммы и видимая только в ней
  - Переменная, хранящая адрес другой переменной
  - Переменная, которая не может изменять своё значение
27. Какой способ передачи данных в подпрограмму означает, что внутри подпрограммы создается копия переданного значения?
- Передача по ссылке
  - Передача по значению
  - Передача по имени
  - Передача по указателю
28. Что такое рекурсивная подпрограмма?
- Подпрограмма, которая вызывает саму себя
  - Подпрограмма, которая не возвращает значение
  - Подпрограмма, объявленная внутри другой подпрограммы
  - Подпрограмма, которая использует глобальные переменные
29. Какой обязательный элемент должна содержать любая рекурсивная функция, чтобы не привести к бесконечным вызовам?

- a) Глобальную переменную-счётчик
  - b) Условие выхода (базовый случай)
  - c) Не менее двух параметров
  - d) Рекурсивный вызов в самом начале
30. Что такое класс в объектно-ориентированном программировании?
- a) Конкретный экземпляр объекта
  - b) Шаблон (описание) для создания объектов, определяющий их свойства и методы
  - c) Отдельная переменная
  - d) Библиотека функций
31. Какой принцип ООП означает скрытие внутренних деталей реализации объекта и предоставление доступа только через открытые методы?
- a) Наследование
  - b) Полиморфизм
  - c) Инкапсуляция
  - d) Абстракция
32. Что позволяет сделать механизм наследования?
- a) Создать новый класс на основе существующего, расширяя или изменяя его функциональность
  - b) Использовать одно и то же имя метода для разных действий в разных классах
  - c) Скрыть данные внутри класса
  - d) Создать несколько объектов одного класса
33. Как называется способность объектов разных классов правильно реагировать на один и тот же метод (например, draw() для фигур)?
- a) Инкапсуляция
  - b) Полиморфизм
  - c) Модульность
  - d) Рекурсия
34. Что такое конструктор класса?
- a) Метод, который удаляет объект
  - b) Специальный метод, вызываемый при создании объекта для его инициализации
  - c) Переменная, хранящая ссылку на объект
  - d) Любой метод, объявленный в классе
35. В чём особенность событийно-ориентированного программирования?
- a) Программа выполняется строго линейно сверху вниз
  - b) Порядок выполнения определяется действиями пользователя (клики, нажатия клавиш)
  - c) Отсутствуют подпрограммы
  - d) Все переменные являются глобальными
36. Какой компонент обычно используется для отображения табличных данных в графическом интерфейсе?
- a) Button
  - b) TextBox
  - c) DataGridView (или аналогичный)
  - d) ComboBox
37. Какая группа компонентов позволяет пользователю выбрать один вариант из нескольких (например, да/нет)?
- a) Кнопочные компоненты (Button)
  - b) Переключатели (RadioButton, CheckBox)
  - c) Компоненты для ввода даты
  - d) Текстовые поля (TextBox)
38. Для чего служит компонент «Таймер» (Timer) в визуальной среде разработки?
- a) Для отображения текущего времени
  - b) Для выполнения определённого кода через заданные интервалы
  - c) Для ввода времени пользователем
  - d) Для хранения даты и времени в базе данных
39. Что такое отладка (debugging) программы?
- a) Написание документации к программе

- b) Процесс поиска и исправления ошибок в программе
  - c) Оптимизация кода для ускорения работы
  - d) Создание установочного пакета
40. Какая цель оптимизации программы?
- a) Увеличение количества строк кода
  - b) Улучшение читаемости кода
  - c) Повышение эффективности (быстродействия и/или уменьшение потребления памяти)
  - d) Добавление новых функций
41. Что из перечисленного относится к тестированию приложения?
- a) Запуск программы с различными входными данными и проверка соответствия ожидаемым результатам
  - b) Написание кода
  - c) Проектирование интерфейса
  - d) Компиляция программы

### Критерии оценивания:

- 5 баллов выставляется, если правильные ответы даны на 85-100% тестовых заданий
- 4 балла выставляется студенту, если правильные ответы даны на 65-84% тестовых заданий
- 3 балла выставляется студенту, если правильные ответы даны на 50-64% тестовых заданий
- 2 балла выставляется студенту, если правильные ответы даны на менее 50% тестовых заданий

### Практические задания:

1. Написать программу, которая запрашивает у пользователя три числа и выводит наибольшее из них.
2. Составить программу, определяющую, является ли введённое целое число чётным. Вывести соответствующее сообщение.
3. Дано целое число. Если оно положительное, прибавить к нему 1; если отрицательное, вычесть 2; если нулевое, заменить на 10. Вывести полученное число.
4. Написать программу решения квадратного уравнения  $ax^2 + bx + c = 0$  (коэффициенты вводятся с клавиатуры). Учесть случаи одного, двух корней и отсутствия действительных корней.
5. Программа запрашивает возраст пользователя. Вывести сообщение «Доступ разрешён», если возраст не менее 18, иначе «Доступ запрещён».
6. Найти сумму всех натуральных чисел от 1 до N (N вводится пользователем). Использовать цикл с параметром.
7. Вывести на экран таблицу умножения для числа K (от 1 до 10) в формате: «K × i = результат».
8. Написать программу, которая вычисляет факториал числа N ( $N! = 1 \cdot 2 \cdot \dots \cdot N$ ). Обработать случай N=0.
9. Дано целое число. Определить количество цифр в нём. (Использовать цикл с делением на 10.)
10. Вывести все простые числа в диапазоне от 2 до M (M вводится).
11. Создать целочисленный массив из 10 элементов, заполнить его случайными числами. Найти сумму всех элементов и вывести её.
12. В массиве из N элементов найти минимальный элемент и его индекс.
13. Дан массив вещественных чисел. Подсчитать количество положительных и отрицательных элементов.
14. Заполнить массив из 15 элементов числами, введёнными с клавиатуры. Вывести все элементы, стоящие на чётных позициях (индексах).
15. Отсортировать одномерный массив по возрастанию методом «пузырька» и вывести результат.

16. Создать матрицу размером  $4 \times 4$ , заполнить её целыми числами построчно (ввод с клавиатуры). Найти сумму всех элементов.
  17. Вывести на экран двумерный массив  $5 \times 5$ , заполненный случайными числами. Затем найти и вывести максимальный элемент в каждой строке.
  18. Дана квадратная матрица порядка  $N$ . Найти сумму элементов на главной диагонали.
  19. Заполнить матрицу  $3 \times 3$  числами от 1 до 9. Транспонировать матрицу (строки поменять местами со столбцами) и вывести результат.
  20. Подсчитать, сколько в заданной матрице элементов, равных нулю.
  21. Написать программу, которая запрашивает строку и выводит её длину (количество символов) без учёта пробелов.
  22. Дана строка. Определить, сколько раз в ней встречается заданный символ (вводится пользователем).
  23. Ввести строку, состоящую из слов, разделённых пробелами. Вывести третье слово (если оно существует).
  24. Заменить в строке все буквы «а» на «о». Исходную и изменённую строки вывести.
  25. Проверить, является ли введённая строка палиндромом (читается одинаково слева направо и справа налево, например, «шалаш»).
  26. Создать список (List) из 5 целых чисел, введённых пользователем. Добавить в него ещё одно число в конец, затем удалить второй элемент. Вывести итоговый список.
  27. Используя коллекцию «словарь» (Dictionary), создать телефонную книгу: ключ – имя, значение – номер телефона. Добавить 3 контакта, затем вывести номер по запрашиваемому имени.
  28. Заполнить список случайными числами от 1 до 100 (размер списка 20). Найти и вывести все уникальные значения (без повторов) с помощью множества (Set).
  29. Дан список строк. Отсортировать его по алфавиту и вывести.
  30. Создать список из 10 целых чисел. Используя методы коллекций, найти максимальное и минимальное значения, а затем развернуть список в обратном порядке.
  31. Написать программу, которая создаёт текстовый файл «numbers.txt» и записывает в него числа от 1 до 10, каждое с новой строки.
  32. Читать все числа из файла «input.txt» (числа записаны через пробел в одной строке), вычислить их сумму и записать результат в файл «output.txt».
  33. Дан текстовый файл «text.txt». Подсчитать в нём количество строк и количество символов (включая пробелы). Результат вывести на экран.
  34. Программа добавляет в конец существующего файла «log.txt» текущую дату и время, а также строку, введённую пользователем.
  35. Скопировать содержимое одного текстового файла в другой, заменяя все пробелы на символ подчёркивания «\_».
  36. Написать функцию, которая принимает два целых числа и возвращает их наибольший общий делитель (НОД). Вызвать её для двух введённых чисел.
  37. Создать процедуру, выводящую на экран прямоугольник из звёздочек заданной ширины и высоты. Параметры передаются в процедуру.
  38. Реализовать рекурсивную функцию вычисления числа Фибоначчи  $F(n)$ , где  $F(0)=0$ ,  $F(1)=1$ ,  $F(n)=F(n-1)+F(n-2)$ . Вычислить  $F(N)$  для введённого  $N$ .
  39. Написать рекурсивную функцию возведения числа  $a$  в натуральную степень  $n$  ( $a^n$ ). Запрещено использовать встроенную операцию возведения в степень.
  40. Разработать модуль (библиотеку), содержащий функции для работы с геометрическими фигурами: площадь круга, площадь прямоугольника. В основной программе импортировать модуль и использовать эти функции.
- Тема 2.2. Основные принципы объектно-ориентированного программирования.
41. Создать класс «Студент» с полями: фамилия, группа, средний балл. Добавить конструктор для инициализации полей. В главной программе создать два объекта и вывести их данные.
  42. Реализовать класс «Счёт» (BankAccount) с полями: номер счёта, владелец, баланс. Предусмотреть методы: пополнение (deposit), снятие (withdraw) и вывод баланса. Создать объект и выполнить несколько операций.

43. Добавить в класс «Студент» инкапсуляцию: поля сделать закрытыми (private), а доступ к ним – через свойства (get/set). В свойстве для среднего балла проверять, чтобы значение было от 2 до 5.

44. Создать базовый класс «Фигура» с методом «Площадь» (возвращает 0). От него унаследовать классы «Круг» (радиус) и «Прямоугольник» (стороны). Переопределить метод площади в каждом наследнике. Продемонстрировать полиморфизм: массив фигур и вызов их площади.

45. Создать проект с двумя переключателями (RadioButton): «Цельсий → Фаренгейт» и «Фаренгейт → Цельсий». Пользователь вводит число в текстовое поле, нажимает кнопку «Конвертировать» – результат выводится в другом поле.

46. Реализовать приложение с компонентом DataGridView (таблица) и кнопкой «Загрузить». При нажатии из текстового файла (числа через запятую) заполняется таблица размером 3×3. Добавить кнопку для вычисления суммы всех элементов.

47. Создать форму с компонентом DateTimePicker (выбор даты) и кнопкой «Рассчитать». При нажатии выводится количество дней, прошедших с выбранной даты до сегодняшнего дня. Использовать компонент Label для вывода.

48. Разработать интерфейс для калькулятора: кнопки цифр и операций, текстовое поле для ввода/вывода. Реализовать базовые операции (+, -, \*, /). Обработать деление на ноль.

49. Создать приложение, которое отображает таблицу успеваемости студентов (колонки: ФИО, Оценка). Пользователь может добавлять новые строки через отдельные поля ввода и кнопку «Добавить». Также предусмотреть кнопку «Сохранить» – записать данные в текстовый файл.

#### **Критерии оценивания:**

- 5 баллов выставляется, если правильные ответы даны на 85-100% практических заданий
- 4 балла выставляется студенту, если правильные ответы даны на 65-84% практических заданий
- 3 балла выставляется студенту, если правильные ответы даны на 50-64% практических заданий
- 2 балла выставляется студенту, если правильные ответы даны на менее 50% практических заданий

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

Процедура оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций состоит из текущего контроля.

**Текущий контроль** успеваемости проводится с использованием оценочных средств, представленных в п. 2 данного приложения. Результаты текущего контроля доводятся до сведения студентов до промежуточной аттестации и учитываются при оценивании знаний, умений, навыков и (или) опыта деятельности.

## МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

### ОП. 07 Основы алгоритмизации и программирования

Методические указания для студентов по освоению дисциплины ОП.07 Основы алгоритмизации и программирования являются частью рабочей программы дисциплины (РПД) (приложением к рабочей программе).

РПД – рабочая программа, утвержденная директором колледжа для изучения дисциплины ОП. 07 Основы алгоритмизации и программирования. Она определяет цели и задачи дисциплины, формируемые в ходе ее изучения компетенции и их компоненты, содержание изучаемого материала, виды занятий и объем выделяемого учебного времени, а также порядок изучения и преподавания дисциплины.

Для самостоятельной учебной работы студента важное значение имеют разделы «Структура и содержание дисциплины (модуля)» и «Учебно-методическое и информационное обеспечение дисциплины (модуля)». В первом указываются разделы и темы изучаемой дисциплины, а также виды занятий и планируемый объем (в академических часах), во втором – рекомендуемая литература и перечень ресурсов информационно-телекоммуникационной сети «Интернет».

Для подготовки к текущему контролю студенты могут воспользоваться оценочными средствами, представленными в Приложении 1 к рабочей программе дисциплины.

#### 1. Описание последовательности действий студента

Приступая к изучению дисциплины необходимо в первую очередь ознакомиться с содержанием РПД, где в разделе «Структура и содержание дисциплины (модуля)» приведено общее распределение часов аудиторных занятий и самостоятельной работы по темам дисциплины.

Залогом успешного освоения дисциплины является регулярное посещение занятий и выполнение предусмотренных программой заданий. Пропуск одного, а тем более нескольких занятий может осложнить освоение разделов курса.

Лекции имеют целью дать систематизированные основы научных знаний по содержанию дисциплины. При изучении и проработке теоретического материала необходимо:

- повторить законспектированный на лекционном занятии материал и дополнить его с учетом рекомендованной по данной теме литературы;
- при самостоятельном изучении теоретической темы подготовить конспект, используя рекомендованные в РПД литературные источники и электронные образовательные ресурсы.

Практические занятия проводятся с целью углубления и закрепления знаний, полученных на лекциях и в процессе самостоятельной работы с учебной литературой.

В процессе практического занятия, как вида учебных занятий, обучающиеся выполняют одно или несколько практических заданий под руководством преподавателя в соответствии с изучаемым содержанием учебного материала.

Выполнение обучающимся практических работ проводится с целью:

- систематизации и закрепления полученных теоретических знаний и практических умений;
- углубления теоретических знаний в соответствии с заданной темой;
- формирования умений применять теоретические знания при решении поставленных задач;
- развития профессиональных компетенций у обучающихся;
- развития творческой инициативы, самостоятельности, ответственности и организованности.

Выполнение обучающимися практических заданий направлено на:

- обобщение, систематизацию, углубление, закрепление полученных теоретических знаний по конкретным темам дисциплины;

- формирование умений применять полученные знания на практике, реализацию единства интеллектуальной и практической деятельности;
- выработку при решении поставленных задач таких профессионально значимых качеств, как самостоятельность, ответственность, точность, творческая инициатива.

При подготовке к практическому занятию необходимо изучить или повторить лекционный материал по соответствующей теме.

## 2. **Самостоятельная работа студента**

Самостоятельная работа студента – самостоятельная учебная деятельность студента, организуемая колледжем и осуществляемая без непосредственного руководства педагога, но по его заданиям и под его контролем.

Цели самостоятельной работы:

- систематизация и закрепление полученных теоретических знаний и практических умений студентов;
- углубление и расширение теоретических знаний;
- формирование умений использовать нормативную, правовую, справочную документацию и специальную литературу;
- развитие познавательных способностей и активности студентов: творческой инициативы, самостоятельности, ответственности и организованности;
- формирование самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;
- воспитание самостоятельности, как личностного качества будущего специалиста.

Самостоятельная работа студента по дисциплине выполняется:

- самостоятельно вне расписания учебных занятий;
- с использованием современных образовательных технологий;
- работа со специальной литературой для подготовки к тестовым, практическим заданиям.

## 3. **Рекомендации по работе с литературой и источниками**

Работу с литературой следует начинать с анализа РПД, содержащей список основной и дополнительной литературы, а также знакомства с учебно-методическими разработками.

В случае возникновения затруднений в понимании учебного материала следует обратиться к другим источникам, где изложение может оказаться более доступным.

Работа с литературой не только полезна как средство более глубокого изучения дисциплины, но и является неотъемлемой частью профессиональной деятельности будущего выпускника.