

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ ДАГЕСТАН  
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
РЕСПУБЛИКИ ДАГЕСТАН «ТЕХНИЧЕСКИЙ КОЛЛЕДЖ ИМЕНИ Р.Н. АШУРАЛИЕВА»

## **РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

### ОП.04. Основы алгоритмизации и программирования


Специальность: 09.02.07 Информационные системы и программирование

Квалификация выпускника: программист

ОДОБРЕНО

предметной (цикловой) комиссией УГС 09.00.00. Информатика и вычислительная техника и 10.00.00 Информационная безопасность

Председатель П(Ц)К

 Ш.М. Мусаева

Протокол №1 от 30 августа 2024 г.

Рабочая программа учебной дисциплины ОП.04. Основы алгоритмизации и программирования разработана на основе:

- Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование, утвержденного приказом Министерства образования и науки Российской Федерации № 1547 от 9 декабря 2016 г. (зарегистрирован Министерством юстиции РФ 26 декабря 2016 г. N 44936);

с учетом:

- Примерной образовательной программы по специальности 09.02.07 Информационные системы и программирование (Приказ ФГБОУ ДПО ИРПО № П-24 от 02.02.2022, реквизиты решения о включении ПООП в реестр: протокол № 3 от 15.07.2021);

в соответствии с рабочим учебным планом.

Разработчик:

- Полозкова Елена Николаевна, преподаватель ГБПОУ РД «Технический колледж имени Р.Н. Ашуралиева»

© Полозкова Елена Николаевна 2024

© ГБПОУ РД «Технический колледж Р.Н. Ашуралиева» 2024

## **СОДЕРЖАНИЕ**

<b>1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ ОП.04. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ.....</b>	<b>4</b>
1.1. Место дисциплины в структуре программы подготовки специалистов среднего звена: .....	4
1.2. Цель и планируемые результаты освоения дисциплины:.....	4
<b>2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ ОП.04. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ.....</b>	<b>5</b>
2.1. Объем учебной дисциплины и виды учебной работы .....	5
2.2. Тематический план и содержание учебной дисциплины ОП.04. Основы алгоритмизации и программирования .....	7
<b>3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ ОП.04. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ.....</b>	<b>12</b>
3.1. Материально-техническое обеспечение реализации программы .....	12
3.2. Информационное обеспечение реализации программы .....	12
3.3. Кадровое обеспечение образовательного процесса .....	13
<b>4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ ОП.04. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ.....</b>	<b>14</b>

# **1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ ОП.04. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ**

## **1.1. Место дисциплины в структуре программы подготовки специалистов среднего звена:**

Учебная дисциплина ОП.04 Основы алгоритмизации и программирования принадлежит общепрофессиональному циклу ОП.00. обязательной части ФГОС по специальности 09.02.07 Информационные системы и программирование.

## **1.2. Цель и планируемые результаты освоения дисциплины:**

Освоение дисциплины должно способствовать формированию общих компетенций:

- ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;
- ОК 02. Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности;
- ОК 04. Эффективно взаимодействовать и работать в коллективе и команде;
- ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста;
- ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках.

Освоение дисциплины должно способствовать овладению профессиональными компетенциями:

- ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием;
- ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием;
- ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств;
- ПК 1.4. Выполнять тестирование программных модулей;
- ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода;
- ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения
- ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования

В результате освоения дисциплины обучающийся должен уметь:

- Разрабатывать алгоритмы для конкретных задач;
- Использовать программы для графического отображения алгоритмов;
- Определять сложность работы алгоритмов;
- Работать в среде программирования;
- Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- Оформлять код программы в соответствии со стандартом кодирования;
- Выполнять проверку, отладку кода программы;

В результате освоения дисциплины обучающийся должен знать:

- Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.
- Эволюцию языков программирования, их классификацию, понятие системы программирования.
- Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.
- Подпрограммы, составление библиотек подпрограмм
- Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения

Код ПК, ОК	Умения	Знания
ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ПК 1.1- ПК 1.5 ПК 2.4, ПК 2.5,	<ul style="list-style-type: none"> <li>– Разрабатывать алгоритмы для конкретных задач.</li> <li>– Использовать программы для графического отображения алгоритмов.</li> <li>– Определять сложность работы алгоритмов.</li> <li>– Работать в среде программирования.</li> <li>– Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.</li> <li>– Оформлять код программы в соответствии со стандартом кодирования.</li> <li>– Выполнять проверку, отладку кода программы.</li> <li>– <i>Использовать методы и приемы формализации задач</i></li> <li>– <i>Применять выбранные языки программирования для написания программного кода</i></li> <li>– <i>Применять инструментарий для создания и актуализации исходных текстов программ</i></li> <li>– <i>Выявлять ошибки в программном коде</i></li> <li>– <i>Применять методы и приемы отладки программного кода</i></li> <li>– <i>Применять современные компиляторы, отладчики и оптимизаторы программного кода</i></li> </ul>	<ul style="list-style-type: none"> <li>– Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.</li> <li>– Эволюцию языков программирования, их классификацию, понятие системы программирования.</li> <li>– Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.</li> <li>– Подпрограммы, составление библиотек подпрограмм</li> <li>– Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения</li> <li>– <i>Методы и приемы формализации задач</i></li> <li>– <i>Синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования</i></li> <li>– <i>Инструментарий для создания и актуализации исходных текстов программ</i></li> <li>– <i>Методы повышения читаемости программного кода</i></li> <li>– <i>Методы и приемы отладки программного кода</i></li> <li>– <i>Типы и форматы сообщений об ошибках, предупреждений</i></li> <li>– <i>Современные компиляторы, отладчики и оптимизаторы программного кода</i></li> <li>– <i>Языки программирования и среды разработки</i></li> </ul>

## 2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ ОП.04. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

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

<b>Вид учебной работы</b>	<b>Объем в часах</b>
<b>Объем образовательной программы</b>	<b>232</b>
в том числе:	
Теоретическое обучение	140
Лабораторные занятия	62
Консультация	2
Самостоятельная работа	22
<b>Промежуточная аттестация в форме экзамена</b>	<b>6</b>

- Объем времени обязательной части ППСЗ 112 час.
- Объем времени вариативной части ППСЗ 92 час.

Вариативная часть используется на углубление подготовки по дисциплине. на 62 часа увеличено теоретическое и практическое обучение, 22 часов выделено на самостоятельную работу, 2 часа на консультацию перед экзаменом, 6 часов выделено на промежуточную аттестацию в форме экзамена.

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

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем в часах	Коды компетенций, ЛР, формированию которых способствует элемент программы		
1	2	3	4		
<b>Раздел 1. Введение в программирование</b>		<b>8</b>	ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ПК 1.1- ПК 1.5 ПК 2.4, 2.5		
<b>Тема 1.1.</b> Языки программирования	<b>Содержание учебного материала</b>				
	1.	Обзор языков программирования. Области применения языков программирования. Стандарты языков программирования.			
	2.	Среда проектирования. Понятие интегрированной среды программирования. Компиляторы и интерпретаторы.			
	3.	Жизненный цикл программы. Программа. Программный продукт и его характеристики. Основные этапы решения задач на компьютере			
<b>Тема 1.2</b> Типы данных	<b>Содержание учебного материала</b>		2		
	4.	Типы данных. Простые типы данных. Производные типы данных.			
<b>Раздел 2. Основы программирования на языке Pascal</b>		<b>54</b>	ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ПК 1.1- ПК 1.5 ПК 2.4, 2.5		
<b>Тема 2.1.</b> Операторы языка Pascal	<b>Содержание учебного материала</b>			16	
	5.	Алфавит и лексика языка. Структура программы.			
	6.	Классификация типов данных языка Pascal.			
	7.	Операции и выражения.			
	8.	Правила формирования и вычисления выражений. Приоритет операций.			
	9.	Ввод и вывод данных. Оператор присваивания. Составной оператор.			
	10.	Условный оператор.			
	11.	Оператор выбора языка Pascal. Безусловный переход			
	12.	Циклы в Pascal			
	<b>Лабораторные занятия</b>				8
	13.	Программирование алгоритмов линейной структуры			
	14.	Программирование алгоритмов разветвляющейся структуры: оператор if, составной оператор			
15.	Программирование алгоритмов разветвляющейся структуры: оператор выбора Case				
	16.	Программирование алгоритмов циклической структуры			
<b>Тема 2.2.</b> Структурированные типы в языке Pascal	<b>Содержание учебного материала</b>			10	
	17.	Массивы. Двумерные массивы			
	18.	Строки. Стандартные процедуры и функции для работы со строками			
	19.	Структурированный тип данных – множество. Операции над множествами			
	20.	Комбинированный тип данных – запись			
	21.	Файлы последовательного доступа. Файлы прямого доступа			
	<b>Лабораторные занятия</b>		6		
22.	Обработка одномерных массивов				

	23.	Обработка двумерных массивов		
	24.	Работа со строками		
<b>Тема 2.3.</b> Процедуры и функции	<b>Содержание учебного материала</b>		6	
	25.	Общие сведения о подпрограммах. Определение и вызов подпрограмм в языке Pascal		
	26.	Область видимости и время жизни переменной		
	27.	Механизм передачи параметров подпрограмм		
	<b>Лабораторные занятия</b>		2	
28.	Создание пользовательских подпрограмм			
<b>Тема 2.4.</b> Динамическое программирование в Pascal	<b>Содержание учебного материала</b>		4	
	29.	Указатели. Описание указателей		
	30.	Основные понятия и применение динамически распределяемой памяти. Создание и удаление динамических переменных		
	<b>Лабораторные занятия</b>		2	
	31.	Использование указателей для организации связанных списков		
<b>Раздел 3. Введение в язык C#</b>			<b>12</b>	
<b>Тема 3.1.</b> Интегрированная среда разработки	<b>Содержание учебного материала</b>		10	ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	32.	Язык C# и программная технология .NET. Требования к аппаратным и программным средствам среды Visual Studio.NET		
	33.	Интерфейс Visual Studio: характеристика, основные окна, инструменты, объекты.		
	34.	Панель компонентов и их свойства. Окно кода проекта.		
	35.	Состав и характеристика проекта в среде Visual Studio.NET. Выполнение проекта.		
	36.	Настройка среды и параметров проекта. Компиляция.		
	<b>Лабораторные занятия</b>		2	
	37.	Знакомство со средой Visual Studio.NET. Создание нового проекта.		
<b>Раздел 4. Основы программирования на языке C#</b>			<b>34</b>	
<b>Тема 4.1.</b> Операторы и операции языка C#	<b>Содержание учебного материала</b>		16	ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	38.	Алфавит и лексемы. Структура программы на C#.		
	39.	Правила и стандарты оформления исходного кода.		
	40.	Типы данных в C#. Преобразование базовых типов данных в C#		
	41.	Консольный ввод/вывод.		
	42.	Арифметические и логические операции. Линейные программы.		
	43.	Операторы ветвления в C#		
	44.	Организация циклов в C#.		
	45.	Методы		
	<b>Лабораторные занятия</b>			
	46.	Линейное программирование в C#		
	47.	Программирование ветвлений в C#. Оператор if. Логические выражения. Оператор Switch		
		48.	Программирование циклов в C#	
<b>Содержание учебного материала</b>		6		
49.	Элементы и понятие массива в C#. Одномерные массивы			



<b>Тема 4.2.</b> Структурированные типы в C#	50.	Прямоугольные массивы. Ступенчатые массивы. Класс System Array.	6	
	51.	Символьные строки в C#		
	<b>Лабораторные занятия</b>			
	52.	Одномерные массивы в C#		
	53.	Двумерные и многомерные массивы в C#		
	54.	Работа со строками в C#		
<b>Раздел 5. Основы объектно-ориентированного программирования (ООП)</b>			<b>18</b>	
<b>Тема 5.1.</b> Основные принципы объектно-ориентированного программирования (ООП)	<b>Содержание учебного материала</b>		14	ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	55.	История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс		
	56.	Основные принципы ООП: инкапсуляция, наследование, полиморфизм. Классы объектов. Компоненты и их свойства.		
	57.	Событийно-управляемая модель программирования. Компонентно-ориентированный подход		
	58.	Понятия: класс, объект, свойства объекта, методы. Синтаксис объявления класса. Элементы класса.		
	59.	Конструкторы и деструкторы		
	60.	XAML. Структура и пространства имен XAML. Пространства имен XAML. Элементы и их атрибуты.		
	61.	Взаимодействие кода C# и XAML		
	<b>Лабораторные занятия</b>			
	62.	Разработка приложения с использованием классов		
	63.	Создание XML-документации	4	
<b>Раздел 6. Визуальное событийно-управляемое программирование. Технология WPF (Windows Presentation Foundation)</b>			<b>66</b>	
<b>Тема 6.1.</b> Введение в WPF	<b>Содержание учебного материала</b>		4	ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	64.	Особенности WPF, новшества технологии WPF.		
	65.	Структура WPF приложения, Page и Frame.		
	<b>Лабораторные занятия</b>		2	
	66.	Начало работы с WPF. Создание калькулятора на WPF		
<b>Тема 6.2.</b> Компоновка WPF	<b>Содержание учебного материала</b>		6	
	67.	Понятие и правила компоновки WPF. Элемент Grid, установка размеров.		
	68.	StackPanel, WrapPanel и примеры их использования		
	69.	Ознакомление с Margin и Padding. Canvas, Z-index и примеры их использования. Свойства компоновки элементов.		
	<b>Лабораторные занятия</b>			
	70.	Создание приложения на WPF с использованием компоновки элементов: Grid	4	
	71.	Создание приложения на WPF с использованием компоновки элементов: StackPanel, WrapPanel		
<b>Тема 6.3.</b> Элементы управления WPF	<b>Содержание учебного материала</b>		26	
	72.	Обзор элементов управления и их свойств. Свойства настройки шрифтов. Цвета фона и шрифта.		
	73.	Элементы управления содержимым. Кнопки		
	74.	CheckBox и RadioButton. Контейнеры GroupBox и Expander		
	75.	Всплывающие подсказки ToolTip и Popup		
	76.	Прокрутка (ScrollViewer)		
	77.	Текстовые элементы управления: TextBlock, TextBox, Label (метка)		

	78.	Элементы управления списками		
	79.	ListBox, ComboBox, ListView		
	80.	Создание вкладок и TabControl		
	81.	Меню		
	82.	ToolBar, TreeView, DataGrid, ProgressBar и Slider		
	83.	Работа с датами. Calendar и DatePicker		
	84.	Работа с изображениями. Image и InkCanvas		
	<b>Лабораторные занятия</b>		16	
	85.	Создание приложения на WPF с использованием элементов управления содержимым (кнопки - Button, метки - Label).		
	86.	Создание приложения на WPF с использованием элементов: CheckBox, RadioButton, GroupBox, Expander		
	87.	Создание приложения на WPF с добавлением всплывающих подсказок.		
	88.	Создание приложения на WPF с использованием текстовых элементов.		
	89.	Создание приложения на WPF с использованием списков и столбцов.		
	90.	Создание приложения на WPF с использованием вкладок.		
	91.	Создание приложения на WPF с использованием компонентов ввода и отображения чисел, дат и времени		
	92.	Создание приложения на WPF с использованием изображений		
<b>Тема 6.4.</b> Стили в WPF	<b>Содержание учебного материала</b>		6	
	93.	Создание стиля. Настройка дизайна с помощью ресурсов.		
	94.	Наследование стилей. Свойства стилей. Задание фона кнопки с помощью стиля		
	95.	Триггеры в WPF.		
	<b>Лабораторные занятия</b>		2	
	96.	Разработка Windows-приложения на WPF с использованием стиля, шрифтов и фона		
<b>Раздел 7. Модульное программирование</b>			<b>12</b>	
<b>Тема 7.1.</b> Понятие модульного программирования	<b>Содержание учебного материала</b>		4	ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	97.	Модульное программирование как метод разработки программ. Программный модуль и его основные характеристики. Типовая структура программного модуля. Инкапсуляция в модулях		
	98.	Порядок разработки программного модуля. Связность модулей. Ошибки периода исполнения и логические ошибки в программах. Обработка ошибок. Исключительные ситуации. Организация обработки исключительных ситуаций		
<b>Тема 7.2.</b> Разработка приложений	<b>Содержание учебного материала</b>		4	ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	99.	Среда разработки приложений. Архитектура оконных приложений. Конфигурации для создания консольных и оконных приложений		
	100.	Разработка приложений как многомодульного проекта		
	<b>Лабораторные занятия</b>		2	
	101.	Разработка многомодульного приложения.		
<b>Консультация</b>			<b>2</b>	

<b>Самостоятельная работа обучающихся:</b>		<b>22</b>	
	<ol style="list-style-type: none"> <li>1) Решение вариативных задач на линейное программирование в Pascal</li> <li>2) Решение вариативных задач на бинарное и множественное ветвление в Pascal</li> <li>3) Решение вариативных задач на циклы в Pascal</li> <li>4) Решение вариативных задач на циклы для формирования числовых последовательностей в Pascal</li> <li>5) Решение вариативных задач по массивам в Pascal</li> <li>6) Решение вариативных задач на обработку строк текста в Pascal</li> <li>7) Решение вариативных задач с подпрограммами в Pascal</li> <li>8) Решение вариативных задач с указателями в Pascal</li> <li>9) Решение вариативных задач на попадание точки в заштрихованную область в C#</li> <li>10) Решение вариативных задач на вычисление значений функций в C#</li> <li>11) Решение вариативных задач на вычисление значения функции по графику в C#</li> <li>12) Решение вариативных задач на циклы в C#</li> <li>13) Решение вариативных задач с одномерными массивами в C#</li> <li>14) Решение вариативных задач с двумерными массивами в C#</li> <li>15) Решение вариативных задач по работе со строками в C#</li> <li>16) Разработка приложения с использованием классов в C#</li> <li>17) Создание многомодульного приложения в C#. Навигация по приложению</li> <li>18) Создание многомодульного приложения в C#. Разработка бизнес-логики</li> </ol>		
	<b>Промежуточная аттестация в форме экзамена</b>	<b>6</b>	
	<b>Всего:</b>	<b>232</b>	

### **3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ ОП.04. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ**

#### **3.1. Материально-техническое обеспечение реализации программы**

Для реализации программы учебной дисциплины предусмотрена лаборатория Программирования и баз данных, оснащенная оборудованием и техническими средствами обучения:

Оборудование учебного кабинета:

- Посадочные места по количеству обучающихся;
- Рабочее место преподавателя;
- Комплект учебно-наглядных пособий «технические средства информатизации».

Оснащение лаборатории программирования:

- Автоматизированные рабочие места на 12 обучающихся (процессор Intel H310 процессор Core, оперативная память 8 Гб);
- Автоматизированное рабочее место преподавателя (процессор Intel Core i7 8700, оперативная память объемом 16 Гб);
- Сервер CPU: Intel i7 8700k 3.7 GHz, RAM 32 Gb, HDD 1 Tb Toshiba.
- Проектор Epson EB-530;
- Интерактивная доска 78 Triumph Board Dry Erase;
- Маркерная доска;
- МФУ Kyocera ECOSYS M2040dn (1102S33NL0) (A4,40 ppm,1200 dpi, 512 Mb, USB 2.0, Network, цв. сканер);
- Программное обеспечение общего и профессионального назначения (ОС Microsoft Windows 10 x64, Adobe Reader 10, Microsoft Office 2016, Microsoft Visio Professional 2016, Notepad++ v7.5.8, ПО Git 2.19.1, Eclipse IDE for JavaEE Developers, .NET Framework JDK 8, Microsoft SQL Server Express Edition, Microsoft Visio Professional, Microsoft Visual Studio, MySQL Installer for Windows, NetBeans, SQLServer Management Studio, Microsoft SQLServer JavaConnector, Android Studio, IntelliJ IDEA.
- Комплект учебно-методической документации;
- Фонд оценочных средств по дисциплине.

#### **3.2. Информационное обеспечение реализации программы**

##### **3.2.1. Основные печатные издания**

1. Основы алгоритмизации и программирования / И.Г. Семакин, А.П. Шестаков. – Москва: Академия, 2021. – 304 с.

##### **3.2.2. Основные электронные издания**

Трофимов, В. В. Основы алгоритмизации и программирования: учебник для среднего профессионального образования / В. В. Трофимов, Т. А. Павловская; под редакцией В. В. Трофимова. — Москва: Издательство Юрайт, 2021. — 137 с. — (Профессиональное образование). — ISBN 978-5-534-07321-8. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/473347> (дата обращения: 13.12.2021).

### 3.2.3. Дополнительные источники

1. Кудрина, Е. В. Основы алгоритмизации и программирования на языке C#: учебное пособие для среднего профессионального образования / Е. В. Кудрина, М. В. Огнева. — Москв : Издательство Юрайт, 2021. — 322 с. — (Профессиональное образование). — ISBN 978-5-534-10772-2. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/475228> (дата обращения: 13.12.2021).
2. Семакин И.Г., Шестаков А.П. Основы алгоритмизации и программирования. Практикум: учеб. пособие для студ. учреждений сред. проф. образования. М.: Издательский центр «Академия», 2019
3. Павловская Т.А. C#. Программирование на языке высокого уровня: учебник для вузов. СПб.: Питер, 2020.
4. Павловская Т.А. Программирование на языке высокого уровня C#. М.: Национальный Открытый Университет «ИНТУИТ», 2016
5. Архангельский А.Я. Язык Pascal и основы программирования в Delphi. М.: Бинوم-Пресс, 2008
6. Агуров П.В. C#. Разработка компонентов в MS Visual Studio 2005/2008. СПб.: БВХ-Петербург, 2008
7. Эндрю Троелсен, Филипп Джепикс Язык программирования C# 6.0 и платформа .NET 4.6 7-е издание. Издательская группа "Диалектика-Вильямс", 2016.

Интернет ресурсы:

- <https://metanit.com/> - сайт о программировании
- [http://www.pascal.helrov.net/index/files\\_pascal\\_programming](http://www.pascal.helrov.net/index/files_pascal_programming) - сайт для начинающих программистов
- <http://www.helloworld.ru/> - Разделы: языки программирования, интернет-технологии, программирование игр, системное программирование, алгоритмы
- <http://www.iguania.ru> - основы программирования для начинающих
- <http://algotlist.manual.ru/> - Алгоритмы, методы, исходники. Олимпиадные задачи по программированию
- <http://www.ict.edu.ru> - федеральный образовательный портал <http://www.edu-it.ru> - ИТ-образование в России: сайт открытого е-консорциума <http://claw.ru/> - Образовательный портал
- <http://claw.ru/> - Образовательный портал
- <http://comp-science.narod.ru/> - Дидактические материалы по информатике и программированию
- <https://visualstudio.microsoft.com/ru/> - Visual Studio
- <https://docs.microsoft.com/ru-ru/dotnet/csharp/> - Документация по .NET
- <http://mycsharp.ru/> - Уроки программирования с нуля. Си-шарп для чайников C#
- <http://nullpro.info/> - Записная книжка программиста новичка, C#, SQL, PHP и все-все-все

### 3.3. Кадровое обеспечение образовательного процесса

Реализация программы учебной дисциплины обеспечивается педагогическими работниками образовательной организации, а также лицами, привлекаемыми к реализации образовательной программы на условиях гражданско-правового договора, в том числе из числа руководителей и работников организаций, направление деятельности которых соответствует области профессиональной деятельности Об Связь, информационные и коммуникационные технологии (имеющих стаж работы в данной профессиональной

области не менее 3 лет).

Квалификация педагогических работников должна отвечать квалификационным требованиям, указанным в квалификационных справочниках.

Требования к квалификации педагогических работников. Высшее профессиональное образование или среднее профессиональное образование по направлению подготовки "Образование и педагогика" или в области, соответствующей преподаваемой дисциплине, без предъявления требований к стажу работы, либо высшее профессиональное образование или среднее профессиональное образование и дополнительное профессиональное образование по направлению деятельности в образовательном учреждении без предъявления требований к стажу работы.

Педагогические работники, привлекаемые к реализации образовательной программы, должны получать дополнительное профессиональное образование по программам повышения квалификации, в том числе в форме стажировки в организациях направления деятельности которых соответствует области профессиональной деятельности Об Связь, информационные и коммуникационные технологии, не реже 1 раза в 3 года с учетом расширения спектра профессиональных компетенций.

#### 4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ ОП.04. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

Результаты обучения	Критерии оценки	Формы и методы оценки
<p>Знания:</p> <ul style="list-style-type: none"> <li>– Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.</li> <li>– Эволюцию языков программирования, их классификацию, понятие системы программирования.</li> <li>– Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.</li> <li>– Подпрограммы, составление библиотек подпрограмм</li> <li>– Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения.</li> <li>– Перечень умений, осваиваемых в рамках дисциплины:</li> <li>– Разрабатывать алгоритмы для конкретных задач.</li> <li>– Использовать программы для графического отображения алгоритмов.</li> <li>– Определять сложность работы алгоритмов.</li> <li>– Работать в среде программирования.</li> </ul>	<p>«<b>Отлично</b>» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.</p> <p>«<b>Хорошо</b>» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.</p> <p>«<b>Удовлетворительно</b>» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.</p>	<p>Устный опрос. Компьютерное тестирование на знание терминологии по теме. Тестирование. Оценка выполненной самостоятельной работы Экзамен по дисциплине.</p>

<ul style="list-style-type: none"> <li>– Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.</li> <li>– Оформлять код программы в соответствии со стандартом кодирования.</li> <li>– Выполнять проверку, отладку кода программы.</li> </ul>	<p><b>«Неудовлетворительно»</b> - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.</p>	
<p>Умения:</p> <ul style="list-style-type: none"> <li>– Разрабатывать алгоритмы для конкретных задач.</li> <li>– Использовать программы для графического отображения алгоритмов.</li> <li>– Определять сложность работы алгоритмов.</li> <li>– Работать в среде программирования.</li> <li>– Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.</li> <li>– Оформлять код программы в соответствии со стандартом кодирования.</li> <li>– Выполнять проверку, отладку кода программы.</li> </ul>		<p>Наблюдение за выполнением практического задания. (деятельностью студента).          Экспертная оценка результатов деятельности обучающегося при выполнении и защите результатов лабораторных занятий.          Экзамен по дисциплине.</p>