

БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ОРЛОВСКОЙ ОБЛАСТИ
«ОРЛОВСКИЙ ТЕХНИКУМ ПУТЕЙ СООБЩЕНИЯ ИМЕНИ В.А. ЛАПОВКИНА»

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
ОП.09 Основы алгоритмизации и программирования
для специальности СПО

09.02.01
Компьютерные системы и комплексы.

*Базовая подготовка
среднего профессионального образования*

Программа учебной дисциплины разработана на основе Федерального государственного образовательного стандарта (далее – ФГОС) по специальности среднего профессионального образования (далее - СПО)

09.02.01

Компьютерные системы и комплексы.

Организация-разработчик: БПОУ ОО «Орловский техникум путей сообщения им. В.А. Лапочкина»

Разработчики:

Курашова В.В., председатель предметно-цикловой комиссии электротехнических дисциплин;

Подрез Н.А., преподаватель информатики;

Долиненко Н.Л., преподаватель спецдисциплин.

Рассмотрено, одобрено и рекомендовано к использованию на заседании предметно-цикловой комиссии электротехнических дисциплин
Протокол № 10 от «20» 06 2020 г.

Проверено:

методист

Киселева Е.П.



Согласовано:

зам. директора

Симонова Г.Н.

СОДЕРЖАНИЕ

1. ПАСПОРТ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	стр. 4
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	5
3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ	13
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	15

1. ПАСПОРТ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

«Основы алгоритмизации и программирования».

1. Область применения программы

Программа учебной дисциплины является частью программы подготовки специалистов среднего звена в соответствии с ФГОС по специальности СПО 09.02.01 Компьютерные системы и комплексы.

Рабочая программа учебной дисциплины может быть использована в дополнительном профессиональном образовании в рамках реализации программ переподготовки кадров в учреждениях СПО.

2. Место учебной дисциплины в структуре программы подготовки специалистов среднего звена:

Учебная дисциплина «Основы алгоритмизации и программирования» относится к профессиональному циклу общепрофессиональных дисциплин.

3. Цели и задачи учебной дисциплины – требования к результатам освоения учебной дисциплины:

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

- формализовать поставленную задачу;
- применять полученные знания к различным предметным областям;
- составлять и оформлять программы на языках программирования;
- тестировать и отлаживать программы.

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

- общие принципы построения и использования языков программирования их классификацию;
- современные интегрированные среды разработки программ;
- процесс создания программ;
- стандарты языков программирования;
- общую характеристику языков ассемблера: назначение, принцип построения и использование.

4. Рекомендуемое количество часов на освоение программы учебной дисциплины:

максимальной учебной нагрузки обучающегося - 150 часа,
в том числе:

обязательной аудиторной учебной нагрузки обучающегося - 100 часов;
самостоятельной работы обучающегося - 50 часа.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Вид учебной работы	Объем часов
Максимальная учебная нагрузка (всего)	150
Обязательная аудиторная учебная нагрузка (всего)	100
в том числе:	
практические занятия	60
Самостоятельная работа учащегося (всего)	50
в том числе:	
- решение задач по темам	10
- работа со справочной и дополнительной литературой	6
- работа с учебными пособиями, в том числе электронными	6
- составление программ по темам	12
- подготовка к практической работе	16
Итоговая аттестация в форме дифференцированного зачёта.	

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

Наименование разделов и тем	Содержание учебного материала, лабораторные и практические работы, самостоятельная работа обучающихся	Объем часов	Уровень освоения
1	2	3	4
Введение	Учебная дисциплина «Основы алгоритмизации и программирования», ее основные задачи и связь с другими дисциплинами. Тенденции развития программного обеспечения вычислительной техники.	1	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы.	2	2
РАЗДЕЛ 1. ОСНОВНЫЕ ПРИНЦИПЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ.		34	
Тема 1.1 Алгоритмы и величины.	Понятие алгоритма. Свойства алгоритмов. Формы записей алгоритмов. Общие принципы построения алгоритмов. Основные алгоритмические конструкции: линейные, разветвляющиеся, циклические. Данные: понятие и типы. Основные базовые типы данных и их характеристика. Структурированные типы данных и их характеристика. Методы сортировки данных	8	2
	Практическое занятие Составление блок-схем алгоритмов.	3	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	4	2
Тема 1.2 Логические основы алгоритмизации	Основы алгебры логики. Логические операции с высказываниями: конъюнкция, дизъюнкция, инверсия. Законы логических операций. Таблицы истинности.	4	2
	Практическое занятие Составление таблиц истинности	3	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	2	2

Тема 1.3 Основы структурного программирования	Методы программирования: структурный, модульный, объектно-ориентированный. Достоинства и недостатки методов программирования. Общие принципы разработки программного обеспечения. Жизненный цикл программного обеспечения. Структурная методика алгоритмизации. Базовые алгоритмические структуры. Метод последовательной детализации. Сборочный метод.	8	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	2	2
Тема 1.4 Языки и системы программирования	Эволюция языков программирования. Классификация языков программирования. Элементы языков программирования. Понятие системы программирования. Структура и способы описания языков программирования высокого уровня.	6	2
	Практическая работа: Типы алгоритмов	2	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	2	2
Раздел 2 ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ ПАСКАЛЬ.		45	
Тема 2.1 Основы языка Турбо Паскаль	История развития языка программирования. Структурная схема программы на языке Турбо Паскаль. Лексика языка. Переменные и константы. Типы данных. Выражения и операции.	2	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	3	2
Тема 2.2 Операторы языка	Арифметические операции, функции, выражения. Оператор присваивания. Ввод данных с клавиатуры и вывод на экран. Управление символьным выводом на экран. Логические величины, операции, выражения. Программирование ветвящихся и циклических алгоритмов. Процедуры и функции. Вычисление рекуррентных последовательностей.	8	2

	Практические занятия. Транслирование алгоритмов. Арифметические выражения на языке Паскаль. Ввод-вывод данных. Логические выражения на языке Паскаль. Составление программ линейной структуры. Составление программ разветвляющейся структуры. Составление программ циклической структуры. Организация процедур. Использование процедур. Организация функций. Использование функций. Вычисление рекуррентных последовательностей.	12	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	4	2
Тема 2.3 Графические средства Турбо Паскаля.	Графические режимы экрана. Цвет фона и цвет рисунка. Графические координаты. Графическое окно. Графические примитивы. Закраски и заполнения. Построение графика функции.	2	2
	Практические занятия Графические средства Турбо Паскаля.	2	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	2	2
Тема 2.4 Символьные строки	Строка. Строковая константа. Операции над строковыми данными.	1	2
	Практические занятия Символьные строки	2	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	2	2

Тема 2.5 Массивы	Массивы как структурированный тип данных. Описание массива. Ввод и вывод одномерных массивов. Ввод и вывод двумерных массивов. Обработка массивов. Стандартные функции для массива целых и вещественных чисел.	3	2
	Практические занятия Обработка одномерных массивов. Обработка двумерных массивов.	2	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	2	2
Тема 2.6 Множества	Структурированные типы данных: множества. Объявление множества. Операции над множествами.	3	2
	Практические работы Работа с данными типа множество.	1	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	2	2
Тема 2.7 Файлы	Файловый тип переменной. Файловый ввод/вывод языка Паскаль. Внешние файлы. Текстовые файлы.	1	2
	Практические работы: Работа с файлом последовательного доступа. Работа с файлом произвольного доступа.	2	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	3	2
Тема 2.8 Внешние подпрограммы и модули	Введение внешних подпрограмм. Создание и использование модулей.	1	2
	Практические работы: Разработка многофайлового проекта	1	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	2	2

Тема 2.9 Методы построения алгоритмов.	Метод последовательной детализации. Рекурсивные методы. Методы перебора в задачах поиска. Методы сортировки данных и сложность алгоритма.	2	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	2	2
РАЗДЕЛ 3. ПРОГРАММИРОВАНИЕ В ОБЪЕКТНО-ОРИЕНТИРОВАННОЙ СРЕДЕ		20	
Тема 3.1 Основные принципы объектно-ориентированного программирования	История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс. Основные принципы ООП: инкапсуляция, наследование, полиморфизм. Событийно-управляемая модель программирования. Компонентно-ориентированный подход. Классы объектов. Компоненты и их свойства.	1	2
	Практическое занятие. Описание объектов.	1	2
	Самостоятельная работа студента: Составление программ по теме ООП Подготовка к практическим работам.	4	2
Тема 3.2 Интегрированная среда разработчика	Требования к аппаратным и программным средствам интегрированной среды разработчика. Интерфейс среды разработчика: характеристика, основные окна, инструменты, объекты. Форма и размещение на ней управляющих элементов. Панель компонентов и их свойства. Окно кода проекта. Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта. Объекты в Турбо Паскале. Интегрированная среда программирования DELPHI.	2	2
	Практические занятия Изучение интегрированной среды разработчика. Создание простого проекта	1	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	2	2

Тема 3.3 Визуальное событийно- управляемое программирование	Основные компоненты (элементы управления) интегрированной среды разработки, их состав и назначение. Дополнительные элементы управления. Свойства компонентов (элементов управления). Виды свойств. Синтаксис определения свойств. Категория свойств. Назначение свойств и их влияние на результат. Управление объектом через свойства. События компонентов (элементов управления), их сущность и назначение. Создание процедур на основе событий. Процедуры, определенные пользователем: синтаксис, передача аргументов. Вызов событий.	2	2
	Практические занятия Создание проекта с использованием кнопочных компонентов. Создание проекта с использованием компонентов для работы с текстом. Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени. Создание проекта с использованием компонентов стандартных диалогов и системы меню.	4	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	4	2
Тема 3.4 Этапы разработки приложения в DELPHI.	Проектирование объектно-ориентированного приложения. Создание интерфейса пользователя. Программирование приложения. Тестирование, отладка приложения. Создание документации. Примеры разработки приложений Delphi.	1	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	2	2
Тема 3.5 Иерархия классов	Классы объектно-ориентированного языка программирования: виды, назначение, свойства, методы, события. Объявление класса, свойств и методов экземпляра класса. Наследование. Перегрузка методов.	1	2
	Практические занятия Объявление класса, создание экземпляров класса. Создание наследованного класса. Перегрузка методов	3	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	2	2

Тема 3.6 Разработка оконного приложения	Разработка функционального интерфейса приложения. Создание интерфейса приложения. Разработка функциональной схемы работы приложения. Создание процедур обработки событий. Компиляция и запуск приложения.	1	2
	Практические занятия Разработка оконного приложения. Разработка оконного приложения с несколькими формами. Разработка многооконного приложения.	3	2
	Самостоятельная работа обучающихся Проработка конспекта занятия, повторение пройденного на занятии материала, изучение дополнительной литературы. Подготовка к практическому занятию	2	2
	Всего:	100	

1. – ознакомительный (узнавание ранее изученных объектов, свойств);
2. – репродуктивный (выполнение деятельности по образцу, инструкции или под руководством)
3. – продуктивный (планирование и самостоятельное выполнение деятельности, решение проблемных задач)

3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Требования к минимальному материально-техническому обеспечению

Реализация учебной дисциплины требует наличия учебной лаборатории (кабинета) системного и прикладного программирования.

Оборудование лаборатории и рабочих мест лаборатории:

- Учебные ПК
- Стационарный комплект интерактивного оборудования (проектор, экран)
- Принтер лазерный

Программное обеспечение:

- Microsoft Windows XP Professional
- MS Visio Professional
- Microsoft Office Professional 2007
- Антивирусное ПО
- Microsoft Visual Studio 2008
- PascalABC.NET
- Delphi 2010

3.2. Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Основные источники:

1. Гуриков С.Р Программирование в среде Lazarus: учебное пос.,-М.:НИЦ ИНФРА, 2019
2. Кистрин А.В. Проектирование цифровых устройств: учебник, - М.:НИЦ ИНФРА, 2019
3. Колдаев В.Д.Гагарина Л.Г.Павловская Т.А. Основы алгоритмизации и программирования: учебное пос., - М.:НИЦ ИНФРА, 2019

Дополнительные источники:

Цветкова М.С. Информатика. Учебник, Академия, 2018

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Контроль и оценка результатов освоения учебной дисциплины осуществляется преподавателем в процессе проведения практических занятий и лабораторных работ, тестирования, а также выполнения обучающимися индивидуальных заданий, проектов, исследований.

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
<i>1</i>	<i>2</i>
Умения:	
работать в среде программирования	практические работы, домашняя контрольная работа, работа со справочными системами
реализовывать построенные алгоритмы в виде программ на конкретном языке программирования	практические работы, экспертная оценка программного кода, контрольные работы
Знания:	
этапов решения задачи на компьютере;	практические работы, контрольная работа
типов данных;	
базовых конструкций изучаемых языков программирования;	практические работы, контрольная работа, экспертная оценка программного кода
принципов структурного и модульного программирования;	практические работы, работа со справочной и дополнительной литературой
принципов объектно-ориентированного программирования	практические работы, работа со справочниками, экспертная оценка программного кода