

## **РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

### **ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ**

---

*название дисциплины*

Для специальности: 09.02.01 Компьютерные системы и комплексы

09.02.01 ОП.09

2023 г.

Рабочая программа учебной дисциплины разработана на основе Федерального государственного образовательного стандарта (ФГОС) по специальности среднего профессионального образования 09.02.01 «Компьютерные системы и комплексы» и учебного плана по указанной специальности.

Организация-разработчик: государственное бюджетное профессиональное образовательное учреждение Архангельской области «Мирнинский промышленно-экономический техникум»

Разработчики:

Кузнецова С.П., заведующий дневным отделением МПЭТ;

<p style="text-align: center;"><b>ОДОБРЕНА</b></p> <p>Цикловой комиссией дисциплин 09.02.01 и 13.02.11</p>	<p>Составлена в соответствии с требованиями Федерального государственного стандарта среднего профессионального образования по специальности 09.02.01 «Компьютерные системы и комплексы»</p>
<p>Председатель цикловой комиссии  А.Е. Мысова <u>«01» сентября 2023</u></p>	<p>Зам. директора по учебной работе  М.Н. Венедиктова <u>«01» сентября 2023</u></p>

## **СОДЕРЖАНИЕ**

<b>1. ПАСПОРТ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	<b>4</b>
<b>2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	<b>5</b>
<b>3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	<b>9</b>
<b>4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	<b>10</b>

# **1. ПАСПОРТ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ**

## **1.1. Область применения программы**

Программа учебной дисциплины является частью основной профессиональной образовательной программы в соответствии с ФГОС по специальности СПО 270113 Компьютерные системы и комплексы базовый уровень.

## **1.2. Место дисциплины в структуре основной профессиональной образовательной программы**

В соответствии с Федеральным государственным образовательным стандартом (третьего поколения) учебным планом предусмотрено изучение дисциплины «Основы алгоритмизации и программирования» в цикле общепрофессиональных дисциплин.

## **1.3. Цели и задачи дисциплины – требования к результатам освоения дисциплины**

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

- использовать языки программирования, строить логически правильные и эффективные программы.

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

- общие принципы построения алгоритмов, основные алгоритмические конструкции;
- понятие системы программирования;
- основные элементы процедурного языка программирования, структура программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти;
- подпрограммы, составление библиотек программ;
- объектно-ориентированная модель программирования, понятие классов и объектов, их свойств и методов.

## **1.4. Рекомендуемое количество часов на освоение программы дисциплины:**

максимальной учебной нагрузки обучающегося **156** часов, в том числе:

обязательной аудиторной учебной нагрузки обучающегося **112** часов;

самостоятельной работы обучающегося **44** часов.

## 2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

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

<b>Вид учебной работы</b>	<b>Объем часов</b>
<b>Максимальная учебная нагрузка (всего)</b>	156
<b>Обязательная аудиторная учебная нагрузка (всего)</b>	112
в том числе:	
практические занятия	20
контрольные работы	
<b>Самостоятельная работа обучающегося (всего)</b>	44
в том числе:	
Подготовка к практическим работам с использованием методических рекомендаций преподавателя, оформление практических работ, отчетов и подготовка к их защите.	24
Работа над дополнительными заданиями повышенной сложности	20
Итоговая аттестация в форме экзамена, дифференцированного зачета	

## 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, курсовая работ (проект)	Объем часов	Уровень освоения
1	2	3	4
<b>Раздел 1.</b> Изучение основных принципов алгоритмизации и программирования		<b>24</b>	
<b>Тема 1.1.</b> Общие принципы построения алгоритмов.	<i>Содержание учебного материала</i> Определение и свойства алгоритма. Виды и способы записи Общие принципы построения алгоритмов. Исполнители алгоритмов	2	2
	Самостоятельная работа обучающихся Конспект по теме «Логические основы алгоритмизации». Решение задач по составлению таблиц истинности.	2	
<b>Тема 1.2.</b> Основные алгоритмические конструкции.	<i>Содержание учебного материала</i> Линейные алгоритмические конструкции. Разветвляющиеся алгоритмические конструкции Циклические алгоритмические конструкции	4	2
	Практические занятия 1. Составление блок-схем линейных и разветвляющихся алгоритмов 2. Составление блок-схем циклических алгоритмов	4	
	Самостоятельная работа обучающихся Составление блок-схем алгоритмов для решения задач различных видов.	6	
<b>Тема 2. Языки и методы программирования</b>		<b>10</b>	
	Содержание учебного материала Языки программирования. Эволюция языков программирования. Классификация языков программирования. Понятие системы программирования. Исходный, объектный и загрузочный модули. Интегрированная среда программирования. Методы программирования: структурный, модульный, объектно-ориентированный. Достоинства и недостатки методов программирования. Общие принципы разработки программного обеспечения. Жизненный цикл программного обеспечения. Типы приложений. Консольные приложения.	4	2,3
	Контрольные работы Контрольная работа по темам 1-2	2	
	Самостоятельная работа обучающихся Презентация по теме «Достоинства и недостатки методов программирования»	4	
<b>Тема 3.</b>	Содержание учебного материала	<b>18</b>	

<b>Программирование на процедурном языке C#</b>	Основные элементы языка. Структурная схема программы на алгоритмическом языке. Лексика языка. Переменные и константы. Типы данных. Выражения и операции. Операторы языка. Синтаксис операторов. Составной оператор. Вложенные условные операторы. Циклические конструкции. Массивы как структурированный тип данных. Объявление массива. Ввод и вывод одномерных и двумерных массивов. Обработка массивов. Стандартные функции для массива целых и вещественных чисел. Структурированные типы данных: строки и множества. Объявление строковых типов данных. Поиск, удаление, замена и добавление символов в строке. Операции со строками. Стандартные функции и процедуры работы со строками. Объявление множества. Операции над множествами.	4	
	Практические занятия 1. Составление программ линейной структуры. 2. Составление программ разветвляющейся структуры. 3. Составление программ циклической структуры. 4. Обработка одномерных и двумерных массивов. 5. Работа со строковыми переменными. 6. Работа с данными типа множество.	10	2,3
	Самостоятельная работа обучающихся Решение задач	4	
<b>Тема 4. Процедуры и функции. Работа с файлами</b>	Содержание учебного материала Понятие подпрограммы. Процедуры и функции, их сущность, назначение и различие. Организация процедур, стандартные процедуры. Процедуры, определенные пользователем: синтаксис, передача аргументов. Формальные и фактические параметры. Процедуры с параметрами, описание процедур. Функции: способы организации и описание. Вызов функций, рекурсия. Программирование рекурсивных алгоритмов. Типы файлов. Организация доступа к файлам. Файлы последовательного доступа. Открытие и закрытие файла последовательного доступа. Файлы произвольного доступа. Порядок работы с файлами произвольного доступа. Создание структуры записи. Открытие и закрытие файла произвольного доступа. Запись и считывание из файла произвольного доступа. Использование файла произвольного доступа. Стандартные процедуры и функции для файлов разного типа.	20 2 2 2	2,3
	Практические занятия 1. Организация и использование процедур. 2. Организация и использование функций. 3. Работа с файлом последовательного доступа. 4. Работа с файлом произвольного доступа.	8	
	Контрольные работы Контрольная работа по темам 3-4	2	
	Самостоятельная работа обучающихся Решение задач	4	
<b>Тема 5. Библиотеки подпрограмм</b>	Содержание учебного материала Программирование модулей. Модуль: синтаксис, заголовок, разделы. Библиотеки подпрограмм: понятие и виды. Схема вызова библиотек. Статическое и динамическое связывание. Использование библиотек подпрограмм.	14 2	2
	Практические занятия 1. Программирование модуля. 2. Создание библиотеки подпрограмм. 3. Использование библиотеки подпрограмм.	6	

	Контрольные работы Контрольная работа по теме 5	2	
	Самостоятельная работа обучающихся Решение задач	4	
<b>Тема 6. Основные принципы объектно-ориентированного программирования</b>	Содержание учебного материала: История развития ООП. Базовые понятия: 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.	4 2	1
	Самостоятельная работа обучающихся Реферат по теме «История развития ООП»	2	
<b>Тема 7. Знакомство с интегрированной средой С#.</b>	Содержание учебного материала: 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. Практические занятия 1. Создание простого проекта.	8 2  2	2
	Самостоятельная работа обучающихся Сообщение по теме «Интегрированная среда разработчика»	4	
<b>Тема 8. Этапы разработки приложения.</b>	Содержание учебного материала: 1 Проектирование объектно-ориентированного приложения. Создание интерфейса пользователя. Программирование приложения. Тестирование, отладка приложения. Создание документации.	6 2	1
	Самостоятельная работа обучающихся Сообщение на тему «Юзабилити»	4	
<b>Тема 9. Иерархия классов</b>	Содержание учебного материала:	12	2
	Классы объектно-ориентированного языка программирования: виды, назначение, свойства, методы, события. Объявление класса, свойств и методов экземпляра класса. Наследование. Перегрузка методов	2	
	Практические занятия 1. Объявление класса, создание экземпляров класса. 2. Создание наследованного класса. 3. Перегрузка методов.	6	
	Самостоятельная работа обучающихся Решение задач	4	
<b>Тема 10. Визуальное событийно-управляемое программирование</b>	Содержание учебного материала:	10	2
	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	
	Практические занятия 1. Создание проекта с использованием компонентов для работы с текстом. 2. Создание проекта с использованием компонентов стандартных диалогов и системы меню.	4	
	Самостоятельная работа обучающихся Подготовить текстуальный конспект по основным визуальным событиям и их обработки в системе интегрированной среды разработчика	4	
<b>Тема 11. Разработка</b>	Содержание учебного материала:	14	

<b>оконного приложения</b>	Разработка функционального интерфейса приложения. Создание интерфейса приложения. Разработка функциональной схемы работы приложения. Создание процедур обработки событий. Компиляция и запуск приложения.	2	2
	Практические занятия 1. Разработка оконного приложения. 2. Разработка оконного приложения с несколькими формами. 3. Разработка многооконного приложения.	6	
	Контрольные работы Контрольная работа по темам 7-11	2	
	Самостоятельная работа обучающихся Решение задач	4	
	<b>Всего</b>	<b>112</b>	
	<b>Самостоятельная работа</b>	<b>44</b>	
	<b>Максимальная нагрузка</b>	<b>156</b>	

Для характеристики уровня освоения учебного материала используются следующие обозначения:

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 ПРОГРАММЫ ДИСЦИПЛИНЫ**

#### **3.1. Требования к минимальному материально-техническому обеспечению**

Реализация программы дисциплины требует наличия учебного кабинета «Программирования и баз данных»; лабораторий информационных систем, инструментальных средств разработки.

Оборудование учебного кабинета на 30 посадочных мест: проектор, комплект учебно-методической документации.

Оборудование лаборатории и рабочих мест лаборатории: 15 персональных компьютеров, принтер, сканер, проектор, программное обеспечение общего и профессионального назначения, комплект учебно-методической документации.

#### **3.2. Информационное обеспечение обучения**

##### **Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы**

Основные источники:

1. Голицына О. Л., Попов И. И. Основы алгоритмизации и программирования: Учеб. пособие. – М.: ФОРУМ: ИНФРА-М, 2005. – 432 с.
2. Культин Н. Б. Turbo Pascal в задачах и примерах. – СПб.: БХВ-Петербург, 2006. – 256 с.

Дополнительные источники:

1. Аляев Ю., Козлов О. Алгоритмизация и языки программирования Pascal, C++, Visual Basic. – М.: Финансы и статистика, 2003.
2. Грызлов В.И., Грызлова Т.П. Турбо Паскаль 7.0. – М.: ДМК, 2000.
3. Немнюгин С.А. Turbo Pascal. – СПб.: Питер, 2000.
4. Семакин И.Г., Шестаков А.П. Основы программирования: Учебник. – М.: Мастерство, 2001.

Интернет-ресурсы:

1. Авторские уроки по информатике (Паскаль)  
<http://uchinfo.com.ua//exclusive/pascal/index.htm>
2. Андреева Т. А. Программирование на языке Паскаль  
<http://www.intuit.ru/department/pl/plpascal/lit.html>

#### 4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Образовательное учреждение, реализующее подготовку по учебной дисциплине, обеспечивает организацию и проведение промежуточной аттестации и текущего контроля индивидуальных образовательных достижений – демонстрируемых обучающимися знаний, умений и навыков.

Текущий контроль проводится преподавателем в процессе проведения практических занятий, тестирования, а также выполнения обучающимися индивидуальных заданий, проектов, исследований.

Обучение по учебной дисциплине завершается промежуточной аттестацией в форме экзамена, который проводит преподаватель дисциплины.

Формы и методы промежуточной аттестации и текущего контроля по учебной дисциплине доводятся до сведения обучающихся не позднее начала двух месяцев от начала обучения по основной профессиональной образовательной программе.

Для промежуточной аттестации и текущего контроля образовательными учреждениями создаются фонды оценочных средств (ФОС).

ФОС включают в себя педагогические контрольно-измерительные материалы, предназначенные для определения соответствия (или несоответствия) индивидуальных образовательных достижений основным показателям оценки результатов подготовки (таблицы).

<b>Результаты обучения (освоенные умения, усвоенные знания)</b>	<b>Формы и методы контроля и оценки результатов обучения</b>
<p>В результате освоения дисциплины обучающийся должен уметь:</p> <ul style="list-style-type: none"><li>использовать языки программирования, строить логически правильные и эффективные программы;</li></ul> <p>В результате освоения дисциплины обучающийся должен знать:</p> <ul style="list-style-type: none"><li>общие принципы построения алгоритмов, основные алгоритмические конструкции;</li></ul>	<p>Практические занятия по всем темам дисциплины. Самостоятельная работа по составлению и тестированию программ различного назначения и структуры</p> <p>Самостоятельная работа по составлению базовых алгоритмов. Практическая работа «Блок-схемы алгоритмов»</p>

<ul style="list-style-type: none"> <li>• понятие системы программирования;</li> </ul>	<p>Тестирование «Методы и системы программирования»</p>
<ul style="list-style-type: none"> <li>• основные элементы процедурного языка программирования, структура программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти;</li> </ul>	<p>Практические работы по составлению программ. Контрольные работы по темам раздела Выполнение творческих заданий по составлению программ</p>
<ul style="list-style-type: none"> <li>• подпрограммы, составление библиотек программ;</li> </ul>	<p>Самостоятельная работа «Использование библиотек подпрограмм»</p>
<ul style="list-style-type: none"> <li>• объектно-ориентированная модель программирования, понятие классов и объектов, их свойств и методов.</li> </ul>	<p>Тестирование «Объектно-ориентированное программирование» Контрольная работа Итоговый экзамен по дисциплине</p>

<b>Результаты (освоенные общие компетенции)</b>	<b>Основные показатели результатов подготовки</b>	<b>Формы и методы контроля</b>
ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.	- демонстрация интереса к будущей профессии.	<i>Экспертное наблюдение и оценка на практических занятиях при выполнении заданий</i>
ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.	- демонстрация эффективности и качества выполнения учебных задач.	<i>Экспертное наблюдение и оценка на практических занятиях при выполнении заданий различной направленности</i>
ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.	- демонстрация способности принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.	<i>Экспертное наблюдение и оценка на практических занятиях при выполнении учебных заданий</i>
ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.	- нахождение и использование информации для эффективного выполнения поставленных задач	<i>Экспертное наблюдение и оценка на практических занятиях при выполнении поставленных задач</i>
ОК 5. Использовать информационно-коммуникационные технологии в профессиональной деятельности.	- демонстрация навыков использования информационно-коммуникационные технологии в профессиональной деятельности.	<i>Экспертное наблюдение и оценка на семинарских и зачётных занятиях при демонстрации итогов самостоятельной работы</i>
ОК 6. Работать в коллективе и команде, эффективно общаться с коллегами, руководством, потребителями.	- взаимодействие с обучающимися и преподавателями в ходе обучения.	<i>Экспертное наблюдение и оценка на практических занятиях</i>
ОК 7. Брать на себя ответственность за работу членов команды (подчиненных), результат выполнения заданий.	- проявление ответственности за работу подчиненных, результат выполнения заданий.	<i>Экспертное наблюдение и оценка на семинарских занятиях при выполнении работ в мини-коллективах.</i>
ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием,	- планирование обучающимся повышения личностного и квалификационного уровня.	<i>Экспертное наблюдение и оценка на практических занятиях при выполнении индивидуальных заданий.</i>

осознанно планировать повышение квалификации.		
ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.	- проявление интереса к инновациям в области профессиональной деятельности.	<i>Экспертное наблюдение и оценка на практических занятиях при выполнении заданий.</i>
ПК 2.1. Создавать программы на языке ассемблера для микропроцессорных систем.	Умение создавать алгоритмы решения задач и реализовывать их с помощью ЯВУ, проводить аналогию решения на языке ассемблера для микропроцессорных систем	<i>Практические занятия, самостоятельная работа, публичное представление презентаций.</i>
ПК 2.3. Проводить тестирование и отладку микропроцессорных систем.	Умение создавать алгоритмы решения задач и реализовывать их с помощью ЯВУ, проводить аналогию решения на языке ассемблера для микропроцессорных систем	<i>Практические занятия, самостоятельная работа, доклады на заданные темы, решение индивидуальных задач</i>
ПК 3.3. Принимать участие в отладке и технических испытаниях компьютерных систем и комплексов; инсталляции, конфигурировании и настройке операционной системы, драйверов, резидентных программ..	Владение базовыми навыками работы с алгоритмическими языками.	<i>Практические занятия, самостоятельная работа, создание проектов решения задач.</i>