



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
КАРАЧАЕВО-ЧЕРКЕССКОЙ РЕСПУБЛИКИ  
КАРАЧАЕВО-ЧЕРКЕССКОЕ РЕСПУБЛИКАНСКОЕ  
ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
«МЕЖДУНАРОДНЫЙ КОЛЛЕДЖ «ПОЛИГЛОТ»

Утверждаю  
Директор Международного  
колледжа «Полиглот»  
\_\_\_\_\_ Ю.М.Дагужиев  
« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

**ОСНОВЫ ПРОГРАММИРОВАНИЯ**

09.02.03 Программирование в компьютерных системах  
по специальности среднего профессионального образования базовой  
подготовки для 2 курса

г. Черкесск, 2017 г.

При разработке программы в основу положен Федеральный государственный образовательный стандарт по специальности 09.02.03 «Программирование в компьютерных системах», утвержденный приказом Министерства образования и науки Российской Федерации № 804 от 28 июля 2014 г.

Организация – разработчик: Карачаево-Черкесское республиканское профессиональное образовательное учреждение «Международный колледж «Полиглот»

Разработчик: Айбазова Ф.Х., преподаватель Международного колледжа «Полиглот».

Рассмотрено на заседании предметно-цикловой комиссии «Общобразовательных и естественнонаучных дисциплин»

От «\_\_» \_\_\_\_\_ 20\_\_ г. протокол № \_\_\_\_\_

Председатель цикловой комиссии \_\_\_\_\_ Л.Б.Дураева

Рекомендовано Педагогическим советом Карачаево-Черкесского республиканского профессионального образовательного учреждения «Международный колледж «Полиглот»

Заключение Педагогического совета № \_\_\_\_\_ от «\_\_» \_\_\_\_\_ 20\_\_ г.

## **РЕЦЕНЗИЯ**

*на рабочую программу учебной дисциплины «Основы программирования» для специальности*

*09.02.03 «Программирование в компьютерных системах»*

*преподавателя*

*Международного колледжа «Полиглот» Айбазовой Ф.Х.*

Рабочая программа учебной дисциплины «Основы программирования» разработана для реализации программы подготовки специалистов среднего звена по специальности 09.02.03. Программирование в компьютерных системах.

Данная программа ориентирована на подготовку студентов в соответствии с компетентностным подходом освоения дисциплины, рекомендаций по организации образовательного процесса, описание назначения дисциплины, отражена роль дисциплины в подготовке специалиста среднего звена, проводится обоснование структуры дисциплины.

Рабочая программа учебной дисциплины «Операционные системы» составлена в соответствии с рекомендациями к разработке учебно-методической документации и может быть рекомендована к обучению студентов специальности 09.02.03. Программирование в компьютерных системах

Рецензент \_\_\_\_\_

## РЕЦЕНЗИЯ

*на рабочую программу учебной дисциплины «Основы программирования» для специальности*

*09.02.03 «Программирование в компьютерных системах»*

*преподавателя*

*Международного колледжа «Полиглот» Айбазовой Ф.Х.*

Рабочая программа предусматривает аудиторное и внеаудиторное самостоятельное изучение дисциплины. В программе приведены различные темы и формы выполнения самостоятельной внеаудиторной работы студентов, что позволяет развивать творческую активность студентов, стремление к углублению знаний.

В программе отражены разделы, темы и содержание учебного материала, наиболее важными являются: Архитектура операционных систем, процессы и потоки, синхронизация процессов, управление памятью, файловые системы, тенденции развития операционных систем.

Учебно-методический план рассчитан на 121 часов из них 47 часа лекций, 44 часов практических работ, 30 часа самостоятельных работ.

Содержание разделов программы способствует развитию знаний, умений и навыков, которым должен овладеть студент по изучаемой дисциплине. Отражена организация текущего, рубежного и итогового контроля в соответствии с балльно-рейтинговой системой.

Рецензент \_\_\_\_\_

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Рабочая программа учебной дисциплины разработана на основе ФГОС по программе подготовки специалистов среднего звена 09.02.03 Программирование в компьютерных средах.

Рабочая программа предназначена для реализации требований к результатам освоения изучаемой дисциплины по ФГОС СПО.

«Основы программирования» изучаются в разделе общепрофессиональных дисциплин, основными задачами которых являются: обучение студентов устанавливать и сопровождать операционные системы; учитывать особенности работы в конкретной операционной системе, организовывать поддержку приложений других операционных систем; пользоваться инструментальными средствами операционной системы.

Учебно-методический план рассчитан на 121 часов из них 47 часа лекций, 44 часов практических работ, 30 часа самостоятельных работ. Вариативная часть распределена следующим образом для изучения тем: Промежуточная форма аттестации осуществляется в форме зачета.

## СОДЕРЖАНИЕ

<b>1. ПАСПОРТ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	<b>стр. 7</b>
<b>2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	<b>8</b>
<b>3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	<b>14</b>
<b>4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	<b>15</b>

# **1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ ОСНОВЫ ПРОГРАММИРОВАНИЯ**

## **1.1. Область применения программы**

Рабочая программа учебной дисциплины является частью ППСЗ программы в соответствии с ФГОС по специальности СПО.

Рабочая программа учебной дисциплины может быть использована в дополнительном профессиональном образовании (в программах повышения квалификации и переподготовки) и в профессиональной подготовке работников в области программирования компьютерных систем

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

В результате освоения дисциплины обучающийся должен уметь:

- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;

В результате освоения дисциплины обучающийся должен знать:

- этапы решения задачи на компьютере;
- типы данных;
- базовые конструкции изучаемых языков программирования;
- принципы структурного и модульного программирования;
- принципы объектно-ориентированного программирования.

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

максимальной учебной нагрузки обучающегося 121 часов, в том числе:  
обязательной аудиторной учебной нагрузки 91 обучающегося часов;  
самостоятельной работы обучающегося 30 часов.

## 2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

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

<b>Вид учебной работы</b>	<b><i>Объем часов</i></b>
<b>Максимальная учебная нагрузка (всего)</b>	<i>121</i>
<b>Обязательная аудиторная учебная нагрузка (всего)</b>	<i>91</i>
в том числе:	
практические занятия	<i>44</i>
<b>Самостоятельная работа обучающегося (всего)</b>	<i>30</i>
<i>Итоговая аттестация в форме зачета</i>	



## 2.2. Тематический план и содержание учебной дисциплины Основы программирования

Наименование разделов и тем	Содержание учебного материала, лабораторные и практические работы, самостоятельная работа обучающихся	Объем часов	Уровень освоения	
1	2	3	4	
<b>Раздел 1. Основные принципы алгоритмизации и программирования</b>		36		
<b>Тема 1.1</b> Основные понятия алгоритмизации	<b>Содержание учебного материала</b>		4	
	1	Понятие алгоритма. Свойства алгоритмов. Формы записей алгоритмов. Общие принципы построения алгоритмов. Основные алгоритмические конструкции: линейные, разветвляющиеся, циклические.		1
	2.	Данные: понятие и типы. Основные базовые типы данных и их характеристика. Структурированные типы данных и их характеристика. Методы сортировки данных		2
	<b>Практические занятия</b> Составление блок-схем линейных алгоритмов Составление блок-схем разветвляющихся алгоритмов Составление блок-схем циклических алгоритмов Составление блок-схем алгоритмов сортировки данных		6	
	<b>Контрольные работы по теме «Составление блок-схем алгоритмов»</b>		2	
	<b>Самостоятельная работа обучающихся</b> Составление блок-схем линейных алгоритмов Составление блок-схем разветвляющихся алгоритмов Составление блок-схем циклических алгоритмов Составление блок-схем алгоритмов сортировки данных		4	
	<b>Тема 1.2.</b> Логические основы алгоритмизации	<b>Содержание учебного материала</b>		2
1.		Основы алгебры логики. Логические операции с высказываниями: конъюнкция, дизъюнкция, инверсия. Законы логических операций. Таблицы истинности.	2	
<b>Практические занятия</b> Составление таблиц истинности		4		
<b>Самостоятельная работа обучающихся</b> Типовой расчет «Логические основы программирования»		2		
<b>Тема 1.3.</b> Языки и системы программирования	<b>Содержание учебного материала</b>		4	
	1	Эволюция языков программирования. Классификация языков программирования. Элементы языков программирования. Понятие системы программирования.		2
	2.	Исходный, объектный и загрузочный модули. Интегрированная среда программирования.	1	
	<b>Самостоятельная работа обучающихся</b> Построение таблицы классификации		2	
<b>Тема 1.4</b> Методы программирования	<b>Содержание учебного материала</b>		2	
	1	Методы программирования: структурный, модульный, объектно-ориентированный. Достоинства и недостатки методов программирования.		1
	2.	Общие принципы разработки программного обеспечения. Жизненный цикл программного обеспечения. Типы приложений. Консольные приложения. Оконные Windows приложения. Web-приложения. Библиотеки. Web-сервисы.	2	

	<b>Контрольные работы по теме «Основы программирования»</b>	2	
	<b>Самостоятельная работа обучающихся</b> Творческая работа «Жизненный цикл программного продукта» (на примере любого программного продукта)	2	
<b>Раздел 2. Программирование на алгоритмическом языке</b>		60	
<b>Тема 2.1. Основные элементы языка</b>	<b>Содержание учебного материала</b>	2	
	1. История развития языка программирования. Структурная схема программы на алгоритмическом языке. Лексика языка. Переменные и константы.		1
	2. Типы данных. Выражения и операции		2
	<b>Самостоятельная работа обучающихся</b> Типовой расчет «Операции и выражения в алгоритмических языках»	2	
<b>Тема 2.2. Операторы языка</b>	<b>Содержание учебного материала</b>	10	
	1. Синтаксис операторов: присваивания, ввода-вывода.		2
	2. Синтаксис операторов: безусловного и условного переходов.		2
	3. Синтаксис операторов: циклов. Составной оператор.		2
	4. Вложенные условные операторы.		2
	5. Циклические конструкции. Циклы с предусловием и постусловием.	3	
	<b>Практические занятия</b> Составление программ линейной структуры. Составление программ разветвляющейся структуры. Составление программ разветвляющейся усложненной структуры. Составление программ циклической структуры. Составление программ усложненной структуры.	10	
<b>Самостоятельная работа обучающихся</b> Типовой расчет «Задачи линейной и ветвящейся конструкции» Типовой расчет «Задачи циклической конструкции»	4		
<b>Тема 2.3. Массивы</b>	<b>Содержание учебного материала</b>	2	
	1. Массивы, как структурированный тип данных. Объявление массива.		2
	2. Ввод и вывод одномерных массивов.		2
	3. Ввод и вывод двумерных массивов.		2
	4. Обработка массивов.		3
	5. Стандартные функции для массива целых и вещественных чисел.	3	
	<b>Практические занятия</b> Обработка одномерных массивов. Обработка двумерных массивов. Использование стандартных функций для работы с массивами.	2	
	<b>Самостоятельная работа обучающихся</b> Типовой расчет «Одномерные массивы» Типовой расчет «Двумерные массивы»	2	
<b>Тема 2.4. Строки и множества</b>	<b>Содержание учебного материала</b>	2	
	1. Структурированные типы данных: строки и множества. Объявление строковых типов данных. Поиск, удаление, замена и добавление символов в строке. Операции со		2

		строками. Стандартные функции и процедуры для работы со строками. Объявление множества. Операции над множествами.		
		<b>Практические занятия</b> Работа со строковыми переменными. Использование стандартных функций и процедур для работы со строками. Работа с данными типа множество. Разработка программ со структурированными типами данных. Разработка усложненных программ со структурированными типами данных.	2	
		<b>Самостоятельная работа обучающихся</b> Типовой расчет «Строки и символы» Типовой расчет «Множества» Типовой расчет «Комбинированный тип»	2	
<b>Тема 2.5.</b> Процедуры и функции	<b>Содержание учебного материала</b>		2	
	1.	Понятие подпрограммы. Процедуры и функции, их сущность, назначение, различие.		2
	2.	Организация процедур, стандартные процедуры.		2
	3.	Процедуры, определенные пользователем: синтаксис, передача аргументов.		2
	4.	Формальные и фактические параметры. Процедуры с параметрами, описание процедур.		2
	5.	Функции: способы организации и описание.		2
	6.	Вызов функций, рекурсия.		2
	7.	Программирование рекурсивных алгоритмов.		3
	8.	Стандартные функции.		2
		<b>Практические занятия</b> Организация процедур. Использование процедур. Организация функций. Использование функций. Применение рекурсивных функций. Использование процедур и функций		2
	<b>Самостоятельная работа обучающихся</b> Типовой расчет «Подпрограммы»	2		
<b>Тема 2.6.</b> Организация ввода-вывода данных. Работа с файлами	<b>Содержание учебного материала</b>		2	
	1.	Типы файлов. Организация доступа к файлам.		2
	2.	Файлы последовательного доступа. Открытие и закрытие файла последовательного доступа. Запись в файл и чтение из файла последовательного доступа.		2
	3.	Файлы произвольного доступа. Порядок работы с файлами произвольного доступа. Создание структуры записи. Открытие и закрытие файла произвольного доступа.		2
	4.	Запись и считывание из файла произвольного доступа. Использование файла произвольного доступа.		2
	5.	Стандартные процедуры и функции для файлов разного типа.		2
		<b>Практические занятия</b> Выполнение операций с файлом последовательного доступа. Выполнение операций с файлом произвольного доступа.		2

	Разработка программ с чтением и записью файлов разных типов. Использование стандартных процедур и функций для работы с файлами.		
	<b>Самостоятельная работа обучающихся</b> Типовой расчет «Типизированные файлы» Типовой расчет «Текстовые файлы»	2	
<b>Тема 2.7.</b> Библиотеки подпрограмм	<b>Содержание учебного материала</b>	2	
	1. Программирование модулей. Модуль: синтаксис, заголовок, разделы.		2
	2. Библиотеки подпрограмм: понятие и виды.		2
	3. Схемы вызова библиотек.		2
	4. Статическое и динамическое связывание.		2
	5. Использование библиотек подпрограмм.		2
	<b>Практические занятия</b> Программирование модуля. Создание библиотеки подпрограмм. Использование библиотеки подпрограммы.	2	
<b>Контрольные работы по теме «Программирование на алгоритмическом языке»</b>	2		
	<b>Самостоятельная работа обучающихся</b> Типовой расчет «Создание модуля»	2	
<b>Раздел 3.</b> Программирование в объектно-ориентированной среде		25	
<b>Тема 3.1.</b> Основные принципы объектно-ориентированного программирования (ООП)	<b>Содержание учебного материала</b>	2	
	1. История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс. Основные принципы ООП: инкапсуляция, наследование, полиморфизм.		2
	2. Событийно-управляемая модель программирования. Компонентно-ориентированный подход. Классы объектов. Компоненты и их свойства.		2
<b>Тема 3.2.</b> Интегрированная среда разработчика	<b>Содержание учебного материала</b>	3	
	1. Требования к аппаратным и программным средствам интегрированной среды разработчика. Интерфейс среды разработчика: характеристика, основные окна, инструменты, объекты.		2
	2. Форма и размещение на ней управляющих элементов. Панель компонентов и их свойства.		2
	3. Окно кода проекта. Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта.		2
	<b>Практические занятия</b> Изучение интегрированной среды разработчика. Создание простого проекта.	2	
<b>Тема 3.3.</b> Этапы разработки приложения	<b>Содержание учебного материала</b>	2	
	1. Проектирование объектно-ориентированного приложения. Создание интерфейса пользователя. Программирование приложения.		2
	2. Тестирование, отладка приложения. Создание документации.		2
	<b>Самостоятельная работа обучающихся</b> Творческая работа «Создание программного продукта»	2	

<b>Тема 3.4.</b> Иерархия классов	<b>Содержание учебного материала</b>		2	
	1.	Классы объектно-ориентированного языка программирования: виды, назначение, свойства, методы, события.		2
	2.	Объявление класса, свойств и методов экземпляра класса. Наследование. Перегрузка методов.	2	2
	<b>Практические занятия</b> Объявление класса, создание экземпляров класса. Создание наследованного класса. Перегрузка методов.		2	
<b>Тема 3.5.</b> Визуальное событийно-управляемое программирование	<b>Содержание учебного материала</b>		2	
	1.	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.		2
	2.	Назначение свойств и их влияние на результат. Управление объектом через свойства. События компонентов (элементов управления), их сущность и назначение.		2
	3.	Создание процедур на основе событий. Процедуры, определенные пользователем: синтаксис, передача аргументов. Вызов событий.	2	2
	<b>Практические занятия</b> 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.		2	
	<b>Самостоятельная работа обучающихся</b> Создание программного продукта: калькулятор Создание программного продукта: текстовый редактор Создание программного продукта: графический редактор		2	
<b>Тема 3.6.</b> Разработка оконного приложения	<b>Содержание учебного материала</b>		2	
	1.	Разработка функционального интерфейса приложения. Создание интерфейса приложения. Разработка функциональной схемы работы приложения.		2
	2.	Создание процедур обработки событий.		2
	3.	Компиляция и запуск приложения.	2	
	<b>Практические занятия</b> Разработка оконного приложения. Разработка оконного приложения с несколькими формами. Разработка многооконного приложения. <b>Контрольные работы по теме «Объектно-ориентированное программирование»</b>		2	
<b>Всего:</b>			121	

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 теоретического обучения; лаборатории системного и прикладного программирования.

Оборудование учебного кабинета: посадочные места по количеству обучающихся, рабочее место преподавателя, комплект учебно-методической документации.

Технические средства обучения: ноутбук, экран, мультимедийный проектор, доска.

Оборудование лаборатории и рабочих мест лаборатории: персональные компьютеры (монитор, системный блок, клавиатура, мышь), комплект учебно-методической документации, программное обеспечение (оболочки языков программирования).

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

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

Основные источники:

1. Семакин, И.Г., Шестаков, А.П. Основы программирования: Учебник. – М.: Мастерство, 2006.
2. Колдаев, В.Д. Основы алгоритмизации и программирования: Учебное пособие/под ред. проф. Л.Г.Гагариной.-М.: ИД «ФОРУМ»: ИНФРА-М,2009.-416с.

Дополнительные источники:

1. Голицына, О.Л., Попов, И.И. Основы алгоритмизации и программирования: Учебное пособие.– М.: Форум: Инфра-М, 2006.
2. Дэн Кларк. Объектно-ориентированное программирование в Visual Basic .NET. – СПб.: Питер, 2007.
3. Немнюгин С.А. Turbo Pascal. – СПб.: Питер, 2008.

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

**Контроль и оценка** результатов освоения 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
<b>умения:</b>	
работать в среде программирования	Выполнение практического задания в соответствии с требованиями к нему; Наблюдение за деятельностью обучающихся на практических занятиях.
реализовывать построенные алгоритмы в виде программ на конкретном языке программирования	Выполнение практического задания в соответствии с требованиями к нему; Наблюдение за деятельностью обучающихся на практических занятиях.
<b>знания:</b>	
этапы решения задачи на компьютере	Контрольная тестовая работа.
типы данных	Контрольная тестовая работа; Выполнение практического задания в соответствии с требованиями к нему; Зачет по окончании дисциплины.
базовые конструкции изучаемых языков программирования	Контрольная тестовая работа; Выполнение практического задания в соответствии с требованиями к нему; Зачет по окончании дисциплины.
принципы структурного и модульного программирования	Контрольная тестовая работа; Выполнение практического задания в соответствии с требованиями к нему; Зачет по окончании дисциплины.
принципы объектно-ориентированного программирования	Контрольная тестовая работа; Выполнение практического задания в соответствии с требованиями к нему; зачет по окончании дисциплины.