

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ МОСКОВСКОЙ ОБЛАСТИ
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ МОСКОВСКОЙ ОБЛАСТИ
«ЛУХОВИЦКИЙ АВИАЦИОННЫЙ ТЕХНИКУМ»**

УТВЕРЖДЕНО

приказом директора ГБПОУ МО
«Луховицкий авиационный техникум»
ОТ «__» _____ 2019Г № ____ /УР
Директор ГБПОУ МО
«Луховицкий авиационный техникум»
_____ О.В. Ларионова

**РАБОЧАЯ ПРОГРАММА
учебной дисциплины**

ОП.04. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

специальность 09.02.07 Информационные системы и программирование

РП.ОП.04.09.02.07/3

Программа учебной дисциплины разработана на основе Федерального государственного образовательного стандарта по специальности среднего профессионального образования 09.02.07 Информационные системы и программирование

Организация-разработчик: ГБПОУ МО «Луховицкий авиационный техникум»

Разработчик:

Головачёва Олеся Николаевна, преподаватель ГБПОУ МО «Луховицкий авиационный техникум»

РАССМОТРЕНА

цикловой комиссией специальности
09.02.07
председатель комиссии
_____ Головачёва О.Н
Протокол №__ от «__» _____ 2019г

СОГЛАСОВАНА

зам. директора по УР
ГБПОУ МО «ЛАТ»
_____ О.В.Рыбакова
«__» _____ 2019г

Рецензенты:

О.Н.Головачёва Председатель цикловой комиссии специальности
09.02.07 ГБПОУ МО «ЛАТ»

СОДЕРЖАНИЕ

	Стр.
1. Паспорт рабочей программы учебной дисциплины	4
2. Структура и содержание учебной дисциплины	6
3. Условия реализации рабочей программы учебной дисциплины	9
4. Контроль и оценка результатов освоения учебной дисциплины	11

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

ОП.04. Основы алгоритмизации и программирования

1.1. Область применения программы

Рабочая программа учебной дисциплины «Основы алгоритмизации и программирования» является частью программы подготовки специалистов среднего звена в соответствии с ФГОС СПО по специальности 09.02.07 Информационные системы и программирование, утвержденным приказом Министерством образования и науки Российской Федерации от 9.12.2016г. № 1547.

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

Дисциплина является обязательной и входит в общепрофессиональный учебный цикл (ОП.00).

Учебная дисциплина «ОП. 04 Основы алгоритмизации и программирования» обеспечивает формирование профессиональных и общих компетенций по всем видам деятельности ФГОС по профессии/специальности 09.02.07 Информационные системы и программирование. Особое значение дисциплина имеет при формировании и развитии ОК1., ОК2., ОК4., ОК5., ОК9., ОК10., ПК1.1- ПК1.5., ПК2.4, 2.5.

В рамках программы учебной дисциплины обучающимися осваиваются умения и знания

<i>Код ПК, ОК</i>	<i>Умения</i>	<i>Знания</i>
ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5	<p>Разрабатывать алгоритмы для конкретных задач.</p> <p>Использовать программы для графического отображения алгоритмов.</p> <p>Определять сложность работы алгоритмов.</p> <p>Работать в среде программирования.</p> <p>Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.</p> <p>Оформлять код программы в соответствии со стандартом кодирования.</p> <p>Выполнять проверку, отладку кода программы.</p>	<p>Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.</p> <p>Эволюцию языков программирования, их классификацию, понятие системы программирования.</p> <p>Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.</p> <p>Подпрограммы, составление библиотек подпрограмм</p> <p>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.</p>

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

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, инкапсуляции и полиморфизма, наследования и переопределения.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объём часов
Объем образовательной программы учебной дисциплины	165
в том числе	
лекции	95
практические занятия	66
Самостоятельная работа	4
Аттестация (Экзамен в 5 семестре)	4

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

Наименование разделов и тем	Содержание		Объем часов		
Тема № 1. Основные принципы алгоритмизации и программирования.	1.	Основные принципы алгоритмизации и программирования.	12		
	2.	Алгоритмы и величины.			
	3.	Линейные вычислительные алгоритмы.			
	4.	Ветвления и циклы в вычислительных алгоритмах.			
	5.	Логические основы алгоритмизации, вспомогательные алгоритмы и процедуры.			
	6.	Основы структурного программирования, Структура и способы описания языков высокого уровня.			
		В том числе, практических занятий и лабораторных работ		-	
	Самостоятельная работа обучающихся		-		
Тема № 2. Программирование на языке Паскаль	7.	Знакомство с языком Паскаль. Элементы языка Турбо Паскаль.	48		
	8.	Концепция типов данных.			
	9.	Арифметические операции, функции, выражения. Оператор присваивания.			
	10.	Ввод данных с клавиатуры и вывод на экран.			
	11.	Символьный вывод данных на экран.			
	12.	Логические величины, операции, выражения. Функции связывающие различные типы данных.			
	13.	Программирование ветвящихся алгоритмов.			
	14.	Программирование циклических алгоритмов.			
	15.	Массивы.			
	16.	Символьные строки.			
	17.	Множества.			
	18.	Файлы.			
	19.	Комбинированный тип данных			
	20.	Указатели и динамические структуры данных.			
	21.	Внешние подпрограммы и модули.			
	22.	Графические средства Турбо Паскаль.			
	23.	Подпрограммы.			
	24.	Рекурсивные подпрограммы.			
		В том числе, практических занятий и лабораторных работ		38	
	25.	Практическая работа №1. Знакомство со средой программирования. Составление программ линейной структуры.			
	26.	Практическая работа №2. Составление программ разветвляющейся структуры.			
	27.	Практическая работа №3. Составление программ циклической структуры.			
	28.	Практическая работа №4. Обработка одномерных массивов.			

	29.	Практическая работа №5. Обработка двумерных массивов.	
	30.	Практическая работа №6. Работа со строками.	
	31.	Практическая работа №7. Работа с данными типа множество.	
	32.	Практическая работа №8. Файлы последовательного доступа.	
	33.	Практическая работа №9. Типизированные файлы.	
	34.	Практическая работа №10. Не типизированные файлы.	
	35.	Практическая работа №11. Организация процедур.	
	36.	Практическая работа №12. Программирование модуля.	
	37.	Практическая работа №13. Создание библиотеки подпрограмм.	
	38.	Практическая работа №14. Использование процедур и функций модуля CRT.	
	39.	Практическая работа №15. Графические средства Турбо Паскаль.	
	40.	Практическая работа №16. Организация и использование функций.	
	41.	Практическая работа №17. Применение рекурсивных функций.	
		Самостоятельная работа обучающихся	2
Тема № 3. Методы построения алгоритмов.	42.	Метод последовательной детализации	6
	43.	Рекурсивные методы	
	44.	Методы перебора в задачах поиска	
	45.	Методы сортировки данных и сложность алгоритмов	
		В том числе, практических занятий и лабораторных работ	-
	Самостоятельная работа обучающихся	-	
Тема № 4. Объектно-ориентированное программирование.	46.	История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс.	29
	47.	Основные принципы ООП: инкапсуляция, наследование, полиморфизм.	
	48.	Классы объектов. Компоненты и их свойства.	
	49.	Событийно-управляемая модель программирования. Компонентноориентированный подход.	
	50.	Требования к аппаратным и программным средствам интегрированной среды разработчика.	
	51.	Интерфейс среды разработчика: характеристика, основные окна, инструменты, объекты. Форма и размещение на ней управляющих элементов.	
	52.	Панель компонентов и их свойства. Окно кода проекта.	
	53.	Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта.	
	54.	Интегрированная среда программирования Delphi.	
	55.	Компоненты Delphi. Свойства компонентов.	
	56.	Событийно-управляемое программирование.	
	57.	Массивы	
	58.	Технология создания приложений в Delphi.	
59.	Примеры разработки приложений Delphi.		

	60.	Иерархия классов	
		В том числе, практических занятий и лабораторных работ	
	61.	Практическая работа №1. Изучение интегрированной среды разработчика. Создание проекта с использованием компонентов для работы с текстом.	28
	62.	Практическая работа №2. Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени.	
	63.	Практическая работа №3. События компонентов (элементов управления), их сущность и назначение	
	64.	Практическая работа №4. Создание процедур на основе событий.	
	65.	Практическая работа №5. Создание проекта с использованием кнопочных компонентов, стандартных диалогов и системы меню.	
	66.	Практическая работа №6. Разработка интерфейса приложения. Тестирование, отладка приложения.	
	67.	Практическая работа №7. Классы ООП: виды, назначение, свойства, методы, события. Объявления класса. Создание наследованного класса.	
	68.	Практическая работа №8. Программирование приложений.	
	69.	Практическая работа №9. Получить навыки использования анимационных компонентов.	
	70.	Практическая работа №10. Разработка тестового приложения	
	71.	Практическая работа №11. Разработка индивидуального проекта	
	72.	Самостоятельная работа обучающихся	2
Аттестация в форме экзамена			4
Объем образовательной программы:			165

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Требования к минимальному материально-техническому обеспечению

Реализация программы дисциплины требует наличия учебного кабинета математических дисциплин, лаборатории программирования и баз данных.

Оборудование учебного кабинета:

- посадочные места по количеству обучающихся;
- рабочее место преподавателя;
- учебно-методический комплекс по дисциплине «Основы алгоритмизации и программирования».

Оборудование лаборатории и рабочих мест лаборатории:

- посадочные места по количеству учащихся, причем каждое рабочее место оборудовано персональным компьютером (компьютеры объединены в локальную сеть);
- рабочее место преподавателя;
- учебно-методический комплекс по дисциплине «Основы алгоритмизации и программирования».

Технические средства обучения:

- персональный компьютер с лицензионным программным обеспечением;
- проекционная аппаратура, управляемая компьютером;
- компьютерные программы тестирования;
- комплект методических указаний для выполнения практических заданий по дисциплине;
- презентационный материал.

3.2. Информационное обеспечение обучения.

Для реализации программы библиотечный фонд образовательной организации должен иметь печатные и/или электронные образовательные и информационные ресурсы, рекомендуемые для использования в образовательном процессе.

3.2.1 Печатные издания:

1. Голицина О. Л., Попов И. И. Основы алгоритмизации и программирования: Учебное пособие. – М.: ФОРУМ, 2015.
2. Зыков С.В. Программирование. Объектно-ориентированный подход. Учебник и практикум. – М.: Юрайт, 2017.
3. Керниган Б., Пайк Р. Практика программирования. – М.: Вильямс, 2017.
4. Кудрина, Е. В. Основы алгоритмизации и программирования на языке C# : учебное пособие для среднего профессионального образования / Е. В. Кудрина, М. В. Огнева. — Москва : Издательство Юрайт, 2019.

5. Подбельский, В. В. Программирование. Базовый курс C# : учебник для среднего профессионального образования / В. В. Подбельский. — Москва : Издательство Юрайт, 2019.

6. Семакин И.Г., Шестаков А.П. Основы алгоритмизации и программирования: Учебник. – М.: Академия, 2015.

7. Шень А. Программирование. Теоремы и задачи. – М.: МЦНМО, 2017.

3.2.2 Дополнительные источники:

1. Васильев А. Программирование на C# для начинающих. – М.: Эксмо, 2018.

2. Канцедал С.А. Алгоритмизация и программирование. – М.: Форум, 2010.

3. Кормен Т. Алгоритмы. Вводный курс. – М.: Вильямс, 2017.

4. Культин Н.Б. Microsoft Visual C# в задачах и примерах. – С-Пб.: БХВ-Петербург, 2014

5. Светозарова Г.И., Сигитов Е.В., Козловский А.В. Практикум по программированию на алгоритмических языках. М.: Книга по требованию, 2012.

3.2.3 Интернет- ресурсы:

1. Единое окно доступа к образовательным ресурсам [Офиц. сайт]. URL: <http://window.edu.ru> (дата обращения: 19.09.2019).

2. Учебный курс – Инструменты, алгоритмы и структуры данных [Электронный ресурс] // Интернет-Университет Информационных технологий – дистанционное образование [Офиц. сайт]. URL: <http://www.intuit.ru/department/se/ialgdate/> (дата обращения: 2.09.2019).

3. Учебники по C# | Microsoft Docs [Электронный ресурс] // Microsoft – официальная страница [Офиц. сайт]. URL: <https://docs.microsoft.com/ru-ru/dotnet/csharp/tutorials/> (дата обращения: 2.09.2019).

4. Федеральный центр информационно-образовательных ресурсов [Офиц. сайт]. URL: <http://fcior.edu.ru/> (дата обращения: 2.08.2018).

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ ОП.04. Основы алгоритмизации и программирования

Результаты обучения (освоенные умения, усвоенные знания)	Критерии оценки	Формы и методы контроля и оценки результатов обучения
Умения:		
Разрабатывать алгоритмы для конкретных задач.	«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.	Наблюдение за выполнением практического задания (деятельностью студента). Оценка выполнения практического задания
Использовать программы для графического отображения алгоритмов.		Наблюдение за выполнением практического задания (деятельностью студента). Оценка выполнения практического задания.
Определять сложность работы алгоритмов.		Наблюдение за выполнением практического задания (деятельностью студента). Оценка выполнения практического задания.
Работать в среде программирования.	«Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.	Оценка выполнения практического задания.
Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.		Наблюдение за выполнением практического задания (деятельностью студента). Оценка выполнения практического задания.
Оформлять код программы в соответствии со стандартом кодирования.		Наблюдение за выполнением практического задания (деятельностью студента). Оценка выполнения практического задания.
Выполнять проверку, отладку кода программы.		Наблюдение за выполнением практического задания (деятельностью студента). Оценка выполнения практического задания.
Знания:		
Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.	«Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в	Наблюдение за выполнением практического задания (деятельностью студента). Оценка выполнения практического задания.
Эволюцию языков программирования, их классификацию, понятие системы программирования.		Подготовка и выступление с докладом, презентацией, внеаудиторная самостоятельная работа.

Результаты обучения (освоенные умения, усвоенные знания)	Критерии оценки	Формы и методы контроля и оценки результатов обучения
Основные элементы языка, структура программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.	<p>основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.</p> <p>«Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки</p>	Наблюдение за выполнением практического задания (деятельностью студента). Подготовка и выступление с докладом, внеаудиторная самостоятельная работа.
Подпрограммы, составление библиотек подпрограмм		Наблюдение за выполнением практического задания (деятельностью студента).
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		Наблюдение за выполнением практического задания (деятельностью студента). Устный и письменный опрос; тестирование; подготовка и выступление с докладом, сообщением, презентацией
Итоговый контроль освоения дисциплины		Экзамен

