

Государственное образовательное учреждение
среднего профессионального образования
Ярославской области
Угличский индустриально-педагогический колледж

ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
ОП.09 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ
для специальности **09.02.01**
Компьютерные системы и комплексы

Углич
2019 г.

УТВЕРЖДАЮ:

_____ Т.М. Смирнова

Заместитель директора

по учебной работе

Рабочая программа учебной дисциплины разработана на основе
Федерального государственного образовательного стандарта (далее – ФГОС)
по специальности среднего профессионального образования (далее СПО)
Компьютерные системы и комплексы.

Организация-разработчик: **ГОУ СПО ЯО Угличский индустриально-
педагогический колледж**

Разработчик:

**Павлов Александр Викторович, преподаватель методической цикловой
комиссии естественно-математических дисциплин**

СОДЕРЖАНИЕ

1. ПАСПОРТ ПРИМЕРНОЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	4
2. СТРУКТУРА И ПРИМЕРНОЕ СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	5
3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРИМЕРНОЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	12
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	13

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

Основы алгоритмизации и программирования

1.1. Область применения программы

Рабочая программа учебной дисциплины является частью основной профессиональной образовательной программы в соответствии с ФГОС по специальности (специальностям) СПО / профессии (профессиям) НПО *230113 Компьютерные системы и комплексы*

1.2. Место дисциплины в структуре основной профессиональной образовательной программы:

Дисциплина общепрофессионального цикла ОП.09

1.3. Цели и задачи дисциплины – требования к результатам освоения 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, принципы построения и использования.

1.4. Рекомендуемое количество часов на освоение программы дисциплины:

максимальной учебной нагрузки обучающегося 378 часа, в том числе:

обязательной аудиторной учебной нагрузки обучающегося 252 часов;
самостоятельной работы обучающегося 126 часов.

2. СТРУКТУРА И ПРИМЕРНОЕ СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	<i>Объем часов</i>
Максимальная учебная нагрузка (всего)	378
Обязательная аудиторная учебная нагрузка (всего)	252
в том числе:	
лабораторные занятия	*
практические занятия	170
контрольные работы	*
курсовая работа (проект) <i>(если предусмотрено)</i>	*
Самостоятельная работа обучающегося (всего)	126
в том числе:	
самостоятельная работа над курсовой работой (проектом) <i>(если предусмотрено)</i>	*
.....	*
.....	*
<i>Указываются другие виды самостоятельной работы при их наличии (реферат, расчетно-графическая работа, домашняя работа и т.п.).</i>	
Итоговая аттестация в форме (указать) зачета экзамена	

2.2. Примерный тематический план и содержание учебной дисциплины *Основы алгоритмизации и программирования*

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающихся, курсовая работа (проект) (если предусмотрены)	Объем часов	Уровень усвоения
Тема 1. Алгоритмы.	Содержание (указывается перечень дидактических единиц)	8	2
	1. Понятие Алгоритма. Машина Тьюринга.	2	
	2. Представления и свойства алгоритмов. Блок-схемы.	2	
	Практические занятия (при наличии, указываются темы)	4	
	1. Составление блок-схем простейших алгоритмов.	4	
Тема 2. Введение в программирование.	Содержание (указывается перечень дидактических единиц)	6	2
	1. Понятие, цели и задачи программирования.	1	
	2. Основные этапы разработки программ.	1	
	3. Понятия исходного кода, объектного и загрузочного модуля. Назначения и виды трансляторов. Компоновка.	1	
	4. Языки программирования и инструментальные средства разработки.	1	
	Практические занятия (при наличии, указываются темы)	2	
	1. Особенности современных языков высокого уровня.	1	
	2. Основные характеристики и свойства языков низкого уровня.	1	
	Тема 3. Основы языка программирования Си.	Содержание (указывается перечень дидактических единиц)	32
1. Алфавит и элементы языка Си. Типы данных. Константы и простые переменные. Идентификаторы.		2	
2. Оператора присваивания. Выражения.		2	
3. Арифметические операции и операторы.		1	
4. Логические и битовые операторы.		1	
5. Условный оператор if. Структурная конструкция if ... else. Тернарный оператор ?.		1	
6. Циклические операторы for, while, do. Операторы break и continue.		1	
7. Оператор switch.		2	
8. Понятие функций. Функция main. Оператор return. Библиотечные функции компилятора.		2	
Практические занятия (при наличии, указываются темы)		20	
1. Вычисление арифметических выражений.		3	

	2.	Приоритеты и правила неявного преобразования типов при вычислении арифметических выражений.	3		
	3.	Вычисление логических и битовых выражений.	3		
	4.	Особенности выполнения логических и битовых операторов.	3		
	5.	Использование условных операторов в процессе программирования.	3		
	6.	Организация циклов в процессе программирования.	3		
	7.	Использование функций и правила передачи параметров.	2		
Тема 4. Инструментальная система разработки Microsoft Visual Studio.	Содержание (указывается перечень дидактических единиц)		6	3	
	1.	Основные функции Microsoft Visual Studio. Понятие решений и проектов. Типы проектов.	1		
	2.	Создание и работа с проектами. Выполнение компиляции и компоновки. Основные функции кодового отладчика.	1		
	3.	Система электронной справки Microsoft Developer Network.	1		
	Практические занятия (при наличии, указываются темы)		3		
	1.	Особенности работы с проектами в Microsoft Visual Studio.	1		
	2.	Выполнение отладки программ в Microsoft Visual Studio	1		
	3.	Использование справочной библиотеки MSDN.	1		
Тема 5. Массивы, структуры, указатели.	Содержание (указывается перечень дидактических единиц)		14	2	
	1.	Понятие и создание массивов. Индексы элементов массива. Многомерные массивы. Размещение массивов в памяти.	1		
	2.	Понятие структуры данных. Оператор struct. Создание структур. Размещение элементов структуры в памяти.	1		
	3.	Понятие указателя. Вычисление адреса. Операции с указателями. Операция разадресации.	1		
	4.	Классы памяти. Область видимости и время жизни объектов.	1		
	Лабораторные работы (при наличии, указываются темы)				
	1.				
	Практические занятия (при наличии, указываются темы)		10		
	1.	Использование массивов в процессе программирования.	3		
	2.	Использование структур данных в процессе программирования.	3		
	3.	Особенности работы с указателями.	4		
	Тема 6. Функции обработки текстовых данных.	Содержание (указывается перечень дидактических единиц)		10	2
		1.	Особенности обработки строк в языке Си. Основные функции манипуляции со строками.	2	
2.		Функции классификации символов.	1		

	Лабораторные работы (при наличии, указываются темы)		
	Практические занятия (при наличии, указываются темы)	7	
	1. Операции со строками с использованием библиотечных функций.	4	
	2. Применение функций классификации символов при работе с тестовыми данными.	3	
Тема 7. Математические вычисления в языке Си.	Содержание (указывается перечень дидактических единиц)	10	2
	1. Особенности математических вычислений в языке Си.	1	
	2. Основные математические функции.	2	
	Практические занятия (при наличии, указываются темы)	7	
	1. Использование математических функций при численных вычислениях.	5	
	2. Задача о табулировании функции.	2	
Тема 8. Ввод-вывод на языке Си.	Содержание (указывается перечень дидактических единиц)	13	2
	1. Консольный ввод-вывод.	1	
	2. Форматирование данных при вводе-выводе	1	
	3. Внутренний ввод-вывод.	1	
	4. Файловый ввод-вывод потокового уровня. Текстовый и двоичный ввод-вывод.	1	
	5. Функции ввода-вывода потокового уровня.	1	
	Лабораторные работы (при наличии, указываются темы)	2	
	1. Решение задач с использованием файлового ввода-вывода.	2	
	Практические занятия (при наличии, указываются темы)	6	
	1. Форматирование текста при выполнении ввода-вывода.	1	
	2. Использование библиотечных функций при файловом вводе-выводе.	3	
	3. Обработка исключительных ситуаций ввода-вывода.	2	
Тема 9. Решение задач с программированием на языке Си.	Содержание (указывается перечень дидактических единиц)	13	2
	1. Функции генерации псевдослучайных чисел.	1	
	2. Метод половинного деления для вычисления нулевых значений функции.	1	
	3. Численное вычисление определённых интегралов.	1	
	4. Поиск и сортировка данных.	1	
	Практические занятия (при наличии, указываются темы)	9	
	1. Генерация псевдослучайных чисел.	1	
	2. Вычисление нулевых значений функции методом половинного деления.	2	
	3. Вычисление определённого интеграла методом трапеций.	2	
	4. Сортировка массивов с использованием функции qsort.	4	
Тема 10. Язык программирования C#.	Содержание (указывается перечень дидактических единиц)	15	2
	1. Понятие .NET-архитектуры. JIT-компиляция.	1	

	2.	Элементы языка C#. Типы данных. Константы. Переменные. Выражения.	1	
	3.	Арифметические, логические, битовые операторы языка C#. Особенности выполнения операторов в языке C#.	2	
	4.	Понятие пространства имён в языке C#. Ключевые слова namespace и using.	1	
	Практические занятия (при наличии, указываются темы)		10	
	1.	Использование MicrosoftVisualStudio при программировании на языке C#.	2	
	2.	Создание простейших консольных приложений на языке C#.	8	
Тема 11. Основы объектно-ориентированного программирования.	Содержание (указывается перечень дидактических единиц)		21	2
	1.	Понятие объектно-ориентированного программирования. Инкапсуляция. Наследование. Полиморфизм. Понятие класса. Свойства и методы класса.	2	
	2.	Конструкторы и деструкторы. Вызов методов и возврат из них. Неявный указатель this. Ключевое слово static.	2	
	3.	Перегрузка операторов и методов классов.	1	
	4.	Массивы в языке C#. Оператор foreach.	2	
	Практические занятия (при наличии, указываются темы)		14	
	1.	Использование классов в процессе объектно-ориентированного программирования.	2	
	2.	Особенности вызова классовых методов. Атрибуты ref и out.	2	
	3.	Использование массивов в языке C#	2	
	4.	Создание консольных приложений на основе объектно-ориентированного программирования.	8	
Тема 12. Основы классового наследования.	Содержание (указывается перечень дидактических единиц)		12	2
	1.	Понятие наследования. Создание наследующих классов.	1	
	2.	Вызов конструкторов и деструкторов при наследовании. Скрытие свойств и методов. Ключевое слово base.	1	
	3.	Виртуальные методы.	1	
	4.	Абстрактные классы. Ключевое слово sealed.	1	
	Практические занятия (при наличии, указываются темы)		8	
	1.	Особенности использования конструкторов при классовом наследовании.	2	
	2.	Использование виртуальных методов и абстрактных классов.	2	
	3.	Создание консольных приложений с полиморфными классами.	4	
	Тема 13. Классы языка C#.	Содержание (указывается перечень дидактических единиц)		18
1.		Класс Math. Свойства и методы класса.	1	
2.		Класс string. Свойства и методы класса.	1	
3.		Класс Random. Свойства и методы класса.	1	

	4.	Коллекционный класс ArrayList. Свойства и методы класса.	1	
	5.	Класс object. Упаковка и распаковка.	1	
	6.	Обработка исключительных ситуаций. Блоки try ... catch ...	1	
	Практические занятия (при наличии, указываются темы)		12	
	1.	Математические вычисления в языке C#	2	
	2.	Обработка текстовых данных на основе класса string	2	
	3.	Генерация псевдослучайных чисел на основе класса Random.	2	
	4.	Особенности работы с коллекциями ArrayList.	2	
	5.	Использование классов языка C# в объектно-ориентированном программировании.	4	
Тема 14. Основы технологии визуального программирования.	Содержание (указывается перечень дидактических единиц)		22	2
	1.	Особенности выполнения программ в операционных системах MicrosoftWindows. Основные понятия и особенности разработки Windows-приложений.	2	
	2.	Виды ресурсов. Окна. Понятие событий.	2	
	3.	Основные типы элементов управления. Программное меню.	2	
	Практические занятия (при наличии, указываются темы)		16	
	1.	Создание Windows-приложения на базе класса Form.	4	
	2.	Использование оконных элементов управления.	6	
3.	Программирование модальных диалогов.	6		
Тема 15. Программирование графики.	Содержание (указывается перечень дидактических единиц)		21	2
	1.	Системы координат. Понятие GDI-объектов. Классы Pen, SolidBrush, Bitmap, Font, Graphics. Графические примитивы.	2	
	2.	Событие Paint. Обработка события Paint.	2	
	3.	Понятие интервального таймера. Класс Timer.	1	
	Практические занятия (при наличии, указываются темы)		16	
	1.	Создание GDI-объектов и рисование графических примитивов.	6	
	2.	Построение графика функции.	5	
3.	Использование интервального таймера.	5		
Тема 16. Программирование на языке ассемблера.	Содержание (указывается перечень дидактических единиц)		35	1
	1.	Основные характеристики языка ассемблера.	1	
	2.	Регистры процессора.	2	
	3.	Арифметические команды.	2	
	4.	Битовые команды.	2	
	5.	Организация условных переходов.	2	
	6.	Организация циклов.	2	

	Практические занятия <i>(при наличии, указываются темы)</i>	24	
1.	Создание простейших программ на языке ассемблера.	4	
2.	Использование арифметических команд.	4	
3.	Использование битовых команд.	4	
4.	Использование условных команд в ассемблере.	4	
5.	Использование циклов.	4	
6.	Использование команд плавающей арифметики.	4	
Примерная тематика внеаудиторной самостоятельной работы		26	
<ul style="list-style-type: none"> • подготовка реферата по теме Основные направления развития современных языков программирования • подготовка выступления по теме Программирование с использованием разноязыковых модулей • Использование средств файлового ввода-вывода при работе с данными • Методы оптимизации программ 			
ВСЕГО (аудиторных часов):		378	

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ДИСЦИПЛИНЫ

3.1. Требования к минимальному материально-техническому обеспечению

Реализация программы дисциплины предполагает наличие учебного кабинета.

Оборудование рабочих мест учебного кабинета:

- компьютерный стол, интерактивная доска (или проектор) для преподавателя;
- компьютерный класс с локальной вычислительной сетью и выходом в Интернет;
- компьютерные столы для обучающихся;
- комплект учебно-методической документации.

Оборудование рабочих мест кабинета:

- компьютер, оборудование которого удовлетворяет требованиям операционной системой Microsoft Windows 7 Professional (x86/x64), с инструментальной системой разработки приложений Microsoft Visual Studio 2010 Express Edition и справочной документацией Microsoft Developer Network;
- комплект учебно-методической документации.

3.2. Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Основная литература:

1. Прата, Стивен. Язык программирования С. Лекции и упражнения. — Пер. с англ. - М.: Издательский дом «Вильямс», 2009. - 960 с.
2. Фленов, Михаил. Библия С#. СПб.: БХВ-Петербург, 2009. - 560 с.

Дополнительная литература:

1. Павловская Т.А. С#. Программирование на языке высокого уровня. Учебник для ВУЗов — СПб.: Питер, 2007. - 432 с.
2. Шилдт Г. Полный справочник по С++. Пер. с англ. - М.: Издательский дом «Вильямс», 2003. - 800 с.
3. Секунов Н. VisualC++ .NET. Самоучитель. - СПб.: БХВ-Петербург, 2002. - 736 с.

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ

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

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
<p>уметь:</p> <ul style="list-style-type: none"> • формализовать поставленную задачу; • применять полученные знания к различным предметным областям; • составлять и оформлять программы на языках программирования; 	<p>Наблюдение и оценка деятельности обучающихся на практических занятиях. Оценка результатов выполнения практических работ (составление вариантов документов).</p>
<p>знать:</p> <ul style="list-style-type: none"> • общие принципы построения и использования языков программирования, их классификацию; • современные интегрированные среды разработки программ; 	<p>Письменный опрос. Тестирование.</p>
<ul style="list-style-type: none"> • стандарты языков программирования; 	<p>Устный опрос. Письменный опрос.</p>
<ul style="list-style-type: none"> • общую характеристику языков ассемблера; 	<p>Письменный опрос.</p>
<ul style="list-style-type: none"> • назначение, принципы построения и использования; 	<p>Устный опрос. Письменный опрос. Оценка результатов выполнения практических.</p>