

Комитет образования и науки Курской области
областное бюджетное профессиональное образовательное учреждение
«Курский государственный политехнический колледж»

УТВЕРЖДАЮ

Директор ОБПОУ «КГПК»

 О.И. Морозова

«29»  2018 г.



Рабочая программа учебной дисциплины
Основы алгоритмизации и программирования
для специальности 09.02.07 Информационные системы и программирование

Курск
2018

Рабочая программа учебной дисциплины разработана на основе федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование, утвержденного приказом Министерства образования и науки Российской Федерации от 09 декабря 2016 года, № 1547 (зарегистрирован Министерством юстиции Российской Федерации 26 декабря 2016 года, регистрационный №44936)

Разработчик: Яковлев И. В., преподаватель ОБПОУ «КГПК»


Рабочая программа учебной дисциплины «Основы алгоритмизации и программирования» рассмотрена и одобрена на заседании П(Ц)К «Информатика и ВТ»

Протокол № 11 от « 10 » _____ 2018 г.

Председатель П(Ц)К  Л. А. Тарасова

Рабочая программа учебной дисциплины «Основы алгоритмизации и программирования» рассмотрена и одобрена на заседании Методического совета
Протокол № 3 от « 21 » нояб 2018 г.

Председатель Методического совета  И.Н.Толмачева

Заместитель директора по учебной работе  Н.Ю. Тарасова

СОДЕРЖАНИЕ

	стр.
1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	4
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	11
3. УСЛОВИЯ РЕАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	20
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	21

1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

1.1. Область применения рабочей программы

Рабочая программа учебной дисциплины «Основы алгоритмизации и программирования» является частью основной образовательной программы среднего профессионального образования – программы подготовки специалистов среднего звена по специальности 09.02.07 Информационные системы и программирование, входящей в укрупненную группу специальностей 09.00.00 Информатика и вычислительная техника.

1.2. Место дисциплины в структуре основной профессиональной образовательной программы:

Учебная дисциплина «Основы алгоритмизации и программирования» относится к общепрофессиональному циклу основной профессиональной образовательной программы.

1.3. Цели и планируемые результаты 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 учебной 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.

В результате изучения дисциплины обучающийся осваивает элементы общих и профессиональных компетенций.

Код	Наименование общих компетенций	Умения	Знания
ОК 1.	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.	<p>Распознавать задачу и/или проблему в профессиональном и/или социальном контексте.</p> <p>Анализировать задачу и/или проблему и выделять её составные части.</p> <p>Правильно определить и найти информацию, необходимую для решения задачи и/или проблемы.</p> <p>Составить план действия,</p> <p>Определить необходимые ресурсы.</p> <p>Владеть актуальными методами работы в профессиональной и смежных сферах.</p> <p>Реализовать составленный план.</p> <p>Оценить результат и последствия своих действий (самостоятельно или с помощью наставника).</p>	<p>Знать актуальный профессиональный и социальный контекст, в котором приходится работать и жить.</p> <p>Знать основные источники информации и ресурсов для решения задач и проблем в профессиональном и/или социальном контексте.</p> <p>Знать актуальные стандарты выполнения работ в профессиональной и смежных областях.</p> <p>Знать актуальные методы работы в профессиональной и смежных сферах.</p>
ОК 2.	Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.	<p>Определять задачи поиска информации</p> <p>Определять необходимые источники информации</p> <p>Планировать процесс поиска</p> <p>Структурировать получаемую информацию</p> <p>Выделять наиболее значимое в перечне информации</p> <p>Оценивать практическую значимость результатов поиска</p> <p>Оформлять результаты поиска</p>	<p>Номенклатуру информационных источников применяемых в профессиональной деятельности</p> <p>Приемы структурирования информации</p> <p>Формат оформления результатов поиска информации</p>
ОК 4.	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.	<p>Организовывать работу коллектива и команды</p> <p>Взаимодействовать с коллегами, руководством, клиентами.</p>	<p>Психология коллектива</p> <p>Психология личности</p> <p>Основы проектной деятельности</p>

ОК 5.	Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.	Излагать свои мысли на государственном языке Оформлять документы	Особенности социального и культурного контекста Правила оформления документов.
ОК 9.	Использовать информационные технологии в профессиональной деятельности.	Применять средства информационных технологий для решения профессиональных задач Использовать современное программное обеспечение	Современные средства и устройства информатизации Порядок их применения и программное обеспечение в профессиональной деятельности
ОК 10.	Пользоваться профессиональной документацией на государственном и иностранном языке.	Понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые), понимать тексты на базовые профессиональные темы участвовать в диалогах на знакомые общие и профессиональные темы строить простые высказывания о себе и о своей профессиональной деятельности кратко обосновывать и объяснить свои действия (текущие и планируемые) писать простые связные сообщения на знакомые или интересующие профессиональные темы	Правила построения простых и сложных предложений на профессиональные темы основные общеупотребительные глаголы (бытовая и профессиональная лексика) лексический минимум, относящийся к описанию предметов, средств и процессов профессиональной деятельности особенности произношения правил чтения текстов профессиональной направленности
ПК 1.1.	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.	Основные этапы разработки программного обеспечения.

		Оформлять документацию на программные средства.	Основные принципы технологии структурного и объектно-ориентированного программирования.
ПК 1.2.	Разрабатывать программные модули в соответствии с техническим заданием.	Создавать программу по разработанному алгоритму как отдельный модуль; Оформлять документацию на программные средства.	Основные этапы разработки программного обеспечения. Основные принципы технологии структурного и объектно-ориентированного программирования.
ПК 1.3.	Выполнять отладку программных модулей с использованием специализированных программных средств.	Выполнять отладку и тестирование программы на уровне модуля. Оформлять документацию на программные средства.	Основные принципы отладки и тестирования программных продуктов. Инструментарий отладки программных продуктов.
ПК 1.4.	Выполнять тестирование программных модулей.	Выполнять отладку и тестирование программы на уровне модуля. Оформлять документацию на программные средства.	Основные виды и принципы тестирования программных продуктов.
ПК 1.5.	Осуществлять рефакторинг и оптимизацию программного кода.	Выполнять оптимизацию и рефакторинг программного кода. Работать с системой контроля версий.	Способы оптимизации и приемы рефакторинга. Инструментальные средства анализа алгоритма. Методы организации рефакторинга и оптимизации кода. Принципы работы с системой контроля версий.
ПК 1.6.	Разрабатывать модули программного обеспечения для мобильных платформ.	Осуществлять разработку кода программного модуля на современных языках программирования. Оформлять документацию на программные средства.	Основные этапы разработки программного обеспечения. Основные принципы технологии структурного и объектно-ориентированного программирования.

<p>ПК 2.4.</p>	<p>Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.</p>	<p>Использовать выбранную систему контроля версий. Анализировать проектную и техническую документацию. Выполнять тестирование интеграции. Организовывать постобработку данных. Использовать приемы работы в системах контроля версий. Оценивать размер минимального набора тестов. Разрабатывать тестовые пакеты и тестовые сценарии. Выполнять ручное и автоматизированное тестирование программного модуля. Выявлять ошибки в системных компонентах на основе спецификаций.</p>	<p>Модели процесса разработки программного обеспечения. Основные принципы процесса разработки программного обеспечения. Основные подходы к интегрированию программных модулей. Основы верификации и аттестации программного обеспечения. Методы и способы идентификации сбоев и ошибок при интеграции приложений. Методы и схемы обработки исключительных ситуаций. Основные методы и виды тестирования программных продуктов. Приемы работы с инструментальными средствами тестирования и отладки. Стандарты качества программной документации. Основы организации инспектирования и верификации. Встроенные и основные специализированные инструменты анализа качества программных продуктов. Методы организации работы в команде разработчиков.</p>
--------------------	--	---	--

ПК 2.5.	Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.	Использовать выбранную систему контроля версий. Использовать методы для получения кода с заданной функциональностью и степенью качества. Анализировать проектную и техническую документацию. Организовывать постобработку данных. Приемы работы в системах контроля версий. Выявлять ошибки в системных компонентах на основе спецификаций.	Модели процесса разработки программного обеспечения. Основные принципы процесса разработки программного обеспечения. Основные подходы к интегрированию программных модулей. Основы верификации и аттестации программного обеспечения. Стандарты качества программной документации. Основы организации инспектирования и верификации. Встроенные и основные специализированные инструменты анализа качества программных продуктов. Методы организации работы в команде разработчиков.
ПК 3.1.	Осуществлять ревьюирование программного кода в соответствии с технической документацией.	Работать с проектной документацией, разработанной с использованием графических языков спецификаций.	Технологии решения задачи планирования и контроля развития проекта. Принятые стандарты обозначений в графических языках моделирования. Типовые функциональные роли в коллективе разработчиков, правила совмещения ролей. Методы организации

			работы в команде разработчиков.
ПК 3.3.	Производить исследование созданного программного кода с использованием специализированных программных средств с целью выявления ошибок и отклонения от алгоритма.	Выполнять оптимизацию программного кода с использованием специализированных программных средств. Использовать методы и технологии тестирования и ревьюирования кода и проектной документации.	Принципы построения диаграмм деятельности программного продукта. Приемы работы с инструментальными средами проектирования программных продуктов.
ПК 3.4.	Проводить сравнительный анализ программных продуктов и средств разработки, с целью выявления наилучшего решения согласно критериям, определенным техническим заданием.	Проводить сравнительный анализ программных продуктов. Проводить сравнительный анализ средств разработки программных продуктов. Разграничивать подходы к менеджменту программных проектов.	Основные методы сравнительного анализа программных продуктов и средств разработки. Основные подходы к менеджменту программных продуктов. Основные методы оценки бюджета, сроков и рисков разработки программ.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Объем образовательной программы	298
Самостоятельная работа	14
Объем работы обучающихся во взаимодействии с преподавателем	278
в том числе:	
теоретические занятия	140
лабораторные занятия (если предусмотрено)	-
практические занятия (если предусмотрено)	136
курсовая работа (проект) (если предусмотрено)	-
контрольная работа	-
консультации	2
Промежуточная аттестация	6
<i>Формы самостоятельной работы (работа с интернет-ресурсами, подготовка к устным опросам, практическим занятиям, контрольной работе, работа с дополнительными источниками, подготовка докладов и презентаций.)</i>	
Промежуточная аттестация проводится в форме экзамена	

2.2. Тематический план и содержание учебной дисциплины

<i>Наименование разделов и тем</i>	<i>Содержание учебного материала и формы организации деятельности обучающихся</i>	<i>Объем часов</i>	<i>Осваиваемые элементы компетенций</i>
<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>
Тема 1. Основы алгоритмизации	<i>Содержание учебного материала</i>	14	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.6 ПК 2.4, 2.5 ПК 3.1, 3.3, 3.4
	<i>Теоретические занятия</i>	8	
	Понятие алгоритма, его свойства и виды. Критерии «хорошего» алгоритма. Способы описания алгоритмов: псевдокоды, блок-схема, программа. Базовые алгоритмические конструкции: линейная, разветвляющаяся, циклическая. Блок-схема: основные элементы, правила составления.	2	
	Различные комбинации алгоритмических конструкций. Тестовые данные. Математическая модель алгоритма. Классические алгоритмы Евклида,	2	
	Разработка линейных алгоритмов	2	
	Разработка нелинейных алгоритмов	2	
	<i>Практические занятия</i>	6	
	1 Практическое занятие № 1 Составление блок-схем простого линейного алгоритма	2	
	2 Практическое занятие №2 Составление блок-схем нелинейного алгоритма	2	
3 Практическое занятие №3 Составление блок-схем алгоритма с конструкцией цикла	2		
Тема 2. Основы программирования	<i>Содержание учебного материала</i>	88	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.6 ПК 2.4, 2.5 ПК 3.1, 3.3, 3.4
	<i>Теоретические занятия</i>	34	
	История языков программирования. Генеалогическое древо C#. Преимущества и недостатки языка C#. Платформа .NET Framework: назначение, составные элементы, тонкости компиляции.	2	
	Обзор интегрированной среды разработки Visual Studio .NET. Создание, сохранение и закрытие проектов и решений. Структура программы. Точка входа.	2	
	Базовый синтаксис C#. Пространство имен. Классы.	2	

<p>Переменные: определение, правила именования. Типы данных: значимые и ссылочные. Объявление переменных и их инициализация. Область действия и время существования переменных.</p> <p>Преобразование типов: явное и неявное.</p> <p>Константы: определение, виды и правила записи в программе.</p> <p>Суффиксы целых и вещественных констант. Escape – последовательности.</p>	2
<p>Понятие выражения. Математические операторы. Старшинство операторов.</p> <p>Математические функции (класс Math).</p> <p>Ввод – вывод данных. Операторы присваивания.</p>	2
<p>Операторы отношения. Оператор if-else: назначение и правила записи.</p> <p>Проверка простых и сложных условий. Вложенные условные операторы.</p> <p>Использование поразрядных (!, &, , ^) и условных (&&,) логических операторов. Отличие в использовании операторов & и &&, и . Тернарный оператор: правила использования. Механизм получения случайного числа.</p>	2
<p>Оператор выбора switch-case: назначение и правила записи.</p> <p>Операторы перехода: break, goto, return.</p>	2
<p>Операторы цикла (while; do.. while; for). Правила записи, отличия в применении.</p> <p>Стандартные операции при работе с циклическими алгоритмами: получение таблицы значений, нахождение суммы и произведения элементов ряда.</p> <p>Принудительный выход из цикла: операторы break и continue.</p> <p>Бесконечные циклы.</p>	2
<p>Массивы: определение, виды. Объявление одномерного массива.</p> <p>Варианты инициализации. Ввод и вывод одномерных массивов.</p>	2
<p>Стандартные операции для работы с массивами: заполнение случайными значениями значениями по формуле, нахождение</p>	2

суммы и произведения, нахождение максимума (минимума), подсчет количества элементов, удовлетворяющих определенному условию, сортировка и поиск элементов.		
Цикл foreach.		2
Двумерные массивы: объявление, ввод и вывод.		2
Работа с двумерными массивами.		2
Понятие потока. Механизм буферизации. Классы библиотеки .NET для работы с потоками.		2
Виды доступа к файлам. Объект FileStream. Классы StreamWriter и StreamReader. Основные операции при работе с файлами.		2
Способы работы с текстовыми файлами.		2
Обработка текстовых файлов.		2
Практические занятия		54
1	Практическое занятие №4 Разработка приложения, выполняющего преобразование типов данных.	2
2	Практическое занятие №5 Разработка приложения, использующего операторы сложения, вычитания, умножения и деления.	2
3	Практическое занятие №6 Разработка приложения, использующего математический класс Math.	4
4	Практическое занятие №7 Разработка приложения с форматированным выводом текста.	2
5	Практическое занятие №8 Разработка приложения, осуществляющего конкатенацию строк.	2
6	Практическое занятие №9 Разработка приложения, использующего операторы if..else.	4
7	Практическое занятие №10 Разработка приложения, использующего оператор switch.	4
8	Практическое занятие №11 Разработка приложения,	4

	использующего оператор цикла с предусловием.	
9	Практическое занятие №12 Разработка приложения, использующего оператор цикла с постусловием.	4
10	Практическое занятие №13 Разработка приложения, использующего одномерные массивы.	4
11	Практическое занятие №14 Разработка приложения, использующего оператор цикла со счетчиком с нетипичными условиями цикла.	4
12	Практическое занятие №15 Разработка приложения, использующего оператор цикла foreach.	4
13	Практическое занятие №16 Разработка приложения, использующего двумерные массивы.	4
14	Практическое занятие №17 Разработка приложения, сортирующего массив.	4
15	Практическое занятие №18 Разработка приложения, работающего с файлами.	4
16	Контрольно-практическое занятие №19 Разработка приложения, генерирующего случайные числа.	2
Тема 3. Объектно-ориентированная модель программирования	<i>Содержание учебного материала</i>	174
	<i>Теоретические занятия</i>	98
	Понятие класса и объекта. Характеристики объекта: поля, свойства, методы, события.	2
	Основные принципы объектно-ориентированного программирования: наследование, полиморфизм, инкапсуляция.	4
	Общая форма определения класса. Модификаторы доступа к элементам класса: public, private, protected, internal.	2
	Объявление классов и создание экземпляров классов.	4
	Переменные ссылочного типа и присваивание. Побочные эффекты множественных ссылок.	2
	Метод: понятие, правила записи. Вызов метода.	2
	Передача параметров по значению. Правило триединого	2

ОК 1
ОК 2
ОК 4
ОК 5
ОК 9
ОК 10
ПК 1.1- ПК 1.6
ПК 2.4, 2.5
ПК 3.1, 3.3, 3.4

соответствия параметров и аргументов: по количеству, типам и по порядку следования.	
Создание методов, возвращающих значения.	2
Способы размещения методов. Конструкторы.	2
Инкапсуляция как управление доступом к данным.	4
Свойства класса: понятие, виды, правила записи.	2
Решение задач на создание классов со свойствами.	4
Наследование и полиморфизм. Иерархия классов: понятие, преимущества. Синтаксис наследования.	4
Скрытие и перекрытие методов. Ключевые слова virtual, override.	2
Вызов методов базового класса («родителя»): ключевое слово base.	2
Тонкости использования конструкторов в иерархически связанных между собой классах.	2
Интерфейсы: назначение, правила написания. Способы реализации интерфейсов.	2
Работа с объектами через интерфейсы. Операторы is и as.	2
Восходящее и нисходящее приведение. Наследование в интерфейсах.	2
Сходства и различия интерфейсов, абстрактных классов и обычных классов.	2
Стандартные интерфейсы .NET: IComparable, ICloneable, IEnumerable.	2
Примеры реализации интерфейсов.	2
Назначение и синтаксис структуры. Элементы тела структуры.	4
Сходства и различия структур и классов. Плюсы и минусы	2

использования структур.		
Решение задач на создание структур.		4
Тип данных enum: объявление и тонкости использования.		2
Операции с перечислениями.		2
Понятие коллекции. Класс List: возможности, правила работы, встроенные методы.		4
Работа с ArrayList.		2
Делегаты: понятие, правила описания. Примеры и тонкости использования делегатов.		4
Обработка события: автоматическое создание обработчиков.		2
Решение задач на создание событий.		2
Проекты WPF.		2
Построение графического интерфейса пользователя средствами XAML.		4
Элементы управления и компоненты проекта WPF.		4
Привязка данных в проектах WPF.		2
Классы DateTime и TimeSpan.		2
Таймеры .Net.		2
Практические занятия		76
1	Практическое занятие № 20 Разработка классов и приложения использующего разработанные классы.	4
2	Практическое занятие № 21 Разработка статических классов и приложения использующего разработанные классы.	4
3	Практическое занятие № 22 Разработка приложения, использующего перегрузку методов класса.	4
4	Практическое занятие № 23 Разработка приложения, использующего механизм наследования классов.	4
5	Практическое занятие № 24 Разработка приложения, использующего аксессоры свойств класса.	4
6	Практическое занятие № 25 Разработка приложения,	4

	использующего структуры.	
7	Практическое занятие № 26 Разработка приложения, использующего перечисления.	4
8	Практическое занятие № 27 Разработка делегатов и событий, и приложения использующего описанные делегаты и события.	4
9	Практическое занятие № 28 Разработка приложения, использующего класс DateTime.	4
10	Практическое занятие № 29 Разработка приложения, использующего класс TimeSpan.	4
11	Практическое занятие № 30 Разработка приложения, использующего класс TimeSpan.	4
12	Практическое занятие № 31 Разработка приложения, использующего классы таймеров .Net.	4
13	Практическое занятие № 32 Разработка приложения, использующего обработку исключений.	4
14	Практическое занятие № 33 Разработка приложения, использующего обработку исключений.	4
15	Практическое занятие № 34 Разработка класса возбуждающего исключения и приложения, использующего разработанный класс.	4
16	Практическое занятие № 35 Разработка простейшего приложения WPF.	4
17	Практическое занятие № 36 Разработка приложения WPF, использующего адаптивную компоновку XAML.	4
18	Практическое занятие № 37 Разработка приложения WPF, использующего привязку данных.	4
19	Практическое занятие № 38 Разработка приложения WPF, использующего элементы управления для отображения табличных данных.	4
	Самостоятельная работа обучающихся Создание докладов на темы: 3. «Перспективы языков программирования» 4. «Кроссплатформенная разработка»	14

	<p>5. <i>«Мобильная разработка»</i></p> <p>6. <i>«Интегрированные среды разработки»</i></p>		
	<i>Консультация</i>	<i>2</i>	
	<i>Экзамен</i>	<i>6</i>	
<i>Всего:</i>		<i>298</i>	

3. УСЛОВИЯ РЕАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Материально-техническое обеспечение

Реализация программы дисциплины осуществляется в лаборатории программирования и баз данных.

Оборудование лаборатории:

- рабочие места по количеству обучающихся;
- рабочее место преподавателя;
- комплект учебно-методической документации;
- наглядные пособия: раздаточный материал.

Технические средства обучения:

- Интерактивная доска SMART
- Ионизатор AIRION
- Лазерный принтер HP LaserJet 3015
- Персональный компьютер – 13 шт
- Пара колонок SVEN
- Программируемое устройство на универсальной последовательной шине
- Табло бегущая строка – 10 шт
- Сканер Canon
- Проектор EPSON
- Сет. адаптер – 11 шт
- Управляемый коммутатор D-Link DIR-8XX
- Установка для изучения логических схем

3.2. Информационное обеспечение обучения

Перечень используемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Основные источники (печатные издания):

1. Подбельский В.В. Язык C#. Базовый курс. – М: Инфра, 2015.- 384 с.
2. Голицына О.Л., Попов И.И. Основы алгоритмизации и программирования: Учебное пособие.– М.: Форум: Инфра-М, 2015.

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Результаты обучения	Критерии оценки	Формы и методы оценки
<p><i>Перечень умений, осваиваемых в рамках дисциплины:</i></p> <ul style="list-style-type: none"> - 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. <p><i>Перечень знаний, осваиваемых в рамках дисциплины:</i></p> <ul style="list-style-type: none"> - 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. 	<p>«Отлично» теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.</p> <p>«Хорошо» теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.</p> <p>«Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат</p>	<p>Результатов выполнения самостоятельной работы Экспертное наблюдение и оценивание выполнения практических работ</p>

	ошибки. «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.	
--	---	--