

Ассоциация научно-технических организаций "Уральский профессиональный форум"
Автономная некоммерческая профессиональная образовательная организация
"Современный цифровой колледж при Западно-уральском институте экономики и права"
(АНПОО "СЦК при ЗУИЭП")



РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
ОПЦ.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

по специальности

09.02.07 Информационные системы и программирование
квалификация «Программист»

форма обучения: очно-заочная

Вводится с 01.09.2024

Пермь 2024

РАССМОТРЕНО

на заседании Педагогического совета
протокол от «26» февраля 2024 г. № 4

Рабочая программа учебной дисциплины «Основы алгоритмизации и программирования» разработана на основе требований федерального государственного образовательного стандарта по специальности 09.02.07 Информационные системы и программирование, утвержденного Приказом Министерства и науки Российской Федерации от 09.12.2016 № 1547 (в действующей редакции, далее по тексту – ФГОС СПО); примерной основной образовательной программы, утвержденной протоколом Федерального учебно-методического объединения по УГПС 09.00.00: от 15 июля 2021 г. № 3, зарегистрировано в государственном реестре примерных основных образовательных программ: регистрационный номер 6 Приказом ФГБОУ ДПО ИРПО № П-24 от 02.02.2022г., предъявляемым к структуре, содержанию, результатам освоения учебного предмета «Основы алгоритмизации и программирования», и является частью образовательной программы среднего профессионального образования - программы подготовки специалистов среднего звена АНПОО "СЦК при ЗУИЭП" по специальности 09.02.07 Информационные системы и программирование, квалификация «Программист».

СОДЕРЖАНИЕ

1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	4
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	4
3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ	10
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	11

1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

1.1. Место дисциплины в структуре основной профессиональной образовательной программы

Учебная дисциплина «Основы алгоритмизации и программирования» принадлежит к общепрофессиональному циклу.

1.2. Цель и планируемые результаты освоения дисциплины:

Цель изучения дисциплины:

научить студентов современным технологиям разработки программного продукта в условиях многократного использования созданных программ и работы вычислительных систем в реальном масштабе времени, обработке и хранению больших объемов информации, диалоговому режиму работы на ЭВМ.

Код ПК, ОК	Умения	Знания
ОК 01., ОК 02., ОК 03., ОК 05., ОК 09., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 11.1., ПК 11.2., ПК 11.3., ПК 11.4., ПК 2.1., ПК 2.2., ПК 2.3., ПК 2.4., ПК 2.5.	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.	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, инкапсуляция и полиморфизма, наследования и переопределения

2. СТРУКТУРА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем в часах
Объем образовательной программы	200
в том числе:	
теоретическое обучение	24

практические занятия	40
<i>Самостоятельная работа</i>	130
Промежуточная аттестация - экзамен	6

1.2. Тематический план и содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающегося, курсовая работа (проект)	Объем в часах	Коды компетенций, формированию которых способствует элемент программы Воспитательный компонент - ЛР
Раздел 1.	<i>Введение в программирование</i>	4	ОК 01., ОК 02., ОК 03., ОК 05., ОК 09., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 11.1., ПК 11.2., ПК 11.3., ПК 11.4., ПК 2.1., ПК 2.2., ПК 2.3., ПК 2.4., ПК 2.5.
Тема 1.1. Языки программирования	Содержание учебного материала 1. Развитие языков программирования. 2. Обзор языков программирования. Области применения языков программирования. Стандарты языков программирования. Среда проектирования. Компиляторы и интерпретаторы. 3. Жизненный цикл программы. Программа. Программный продукт и его характеристики. 4. Основные этапы решения задач на компьютере.	2	ЛР 5, ЛР 6, ЛР 7, ЛР 8, ЛР 9, ЛР 10, ЛР 13
Тема 1.2. Типы данных	Содержание учебного материала 1. Типы данных. Простые типы данных. Производные типы данных. Структурированные типы данных.	2	
Раздел 2.	Содержание учебного материала	4	ОК 01., ОК 02., ОК 03., ОК 05., ОК 09., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 11.1., ПК 11.2., ПК 11.3., ПК 11.4., ПК 2.1., ПК 2.2., ПК 2.3., ПК 2.4., ПК 2.5.
Тема 2.1. Операторы языка программирования	1. Операции и выражения. Правила формирования и вычисления выражений. Структура программы. Ввод и вывод данных. Оператор присваивания. Составной оператор. 2. Условный оператор. Оператор выбора. 3. Цикл с постусловием. Цикл с предусловием. Цикл с параметром. Вложенные циклы. 4. Массивы. Двумерные массивы. Строки. Стандартные процедуры и функции для работы со строками. 5. Структурированный тип данных – множество. Операции над множествами. 6. Комбинированный тип данных – запись. Файлы последовательного доступа. Файлы прямого доступа		ЛР 5, ЛР 6, ЛР 7, ЛР 8, ЛР 9, ЛР 10, ЛР 13
Раздел 3.	Содержание учебного материала	6	ОК 01., ОК 02., ОК 03.,

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающегося, курсовая работа (проект)	Объем в часах	Коды компетенций, формированию которых способствует элемент программы Воспитательный компонент - ЛР
Тема 3.1. Процедуры и функции	1. Общие сведения о подпрограммах. Определение и вызов подпрограмм. Область видимости и время жизни переменной. Механизм передачи параметров. Организация функций.	2	ОК 05., ОК 09., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 11.1., ПК 11.2., ПК 11.3., ПК 11.4., ПК 2.1., ПК 2.2., ПК 2.3., ПК 2.4., ПК 2.5.
	2. Рекурсия. Программирование рекурсивных алгоритмов.		
Тема 3.2. Структуризация в программировании	Содержание учебного материала	2	ЛР 5, ЛР 6, ЛР 7, ЛР 8, ЛР 9, ЛР 10, ЛР 13
	1. Основы структурного программирования. Методы структурного программирования.		
Тема 3.3. Модульное программирование	Содержание учебного материала	2	ЛР 5, ЛР 6, ЛР 7, ЛР 8, ЛР 9, ЛР 10, ЛР 13
	1. Модульное программирование. Понятие модуля. Структура модуля. Компиляция и компоновка программы.		
	2. Стандартные модули.		
Раздел 4	<i>Основные конструкции языков программирования</i>	4	ОК 01., ОК 02., ОК 03., ОК 05., ОК 09., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 11.1., ПК 11.2., ПК 11.3., ПК 11.4., ПК 2.1., ПК 2.2., ПК 2.3., ПК 2.4., ПК 2.5.
Тема 4.1. Указатели.	Содержание учебного материала	4	ЛР 5, ЛР 6, ЛР 7, ЛР 8, ЛР 9, ЛР 10, ЛР 13
	1. Указатели. Описание указателей. Основные понятия и применение динамически распределяемой памяти. Создание и удаление динамических переменных.		
	2. Структуры данных на основе указателей.		
	3. Задача о стеке.		
Раздел 5	Содержание учебного материала	6	ОК 01., ОК 02., ОК 03., ОК 05., ОК 09., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 11.1., ПК 11.2., ПК 11.3., ПК 11.4., ПК 2.1., ПК 2.2.,
Тема 5.1. Основные принципы объектно-ориентированного программирования	1. История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс.	1	ОК 01., ОК 02., ОК 03., ОК 05., ОК 09., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 11.1., ПК 11.2., ПК 11.3., ПК 11.4., ПК 2.1., ПК 2.2.,
	2. Основные принципы ООП: инкапсуляция, наследование, полиморфизм.		
	3. Классы объектов. Компоненты и их свойства.		
	4. Событийно-управляемая модель программирования. Компонентно-		

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающегося, курсовая работа (проект)	Объем в часах	Коды компетенций, формированию которых способствует элемент программы Воспитательный компонент - ЛР
(ООП)	ориентированный подход.		ПК 2.3., ПК 2.4., ПК 2.5.
Тема 5.2. Интегрированная среда разработчика.	Содержание учебного материала	1	ЛР 5, ЛР 6, ЛР 7, ЛР 8, ЛР 9, ЛР 10, ЛР 13
	1. Требования к аппаратным и программным средствам интегрированной среды разработчика.		
	2. Интерфейс среды разработчика: характеристика, основные окна, инструменты, объекты. Форма и размещение на ней управляющих элементов.		
	3. Панель компонентов и их свойства. Окно кода проекта.		
	4. Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта.		
	5. Панель компонентов и их свойства. Окно кода проекта. Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта.		
Тема 5.3. Визуальное событийно-управляемое программирование	Содержание учебного материала	1	
	1. Основные компоненты (элементы управления) интегрированной среды разработки, их состав и назначение.		
	2. Дополнительные элементы управления. Свойства компонентов. Виды свойств. Синтаксис определения свойств. Назначения свойств и их влияние на результат. Управление объектом через свойства.		
Тема 5.4. Разработка оконного приложения	Содержание учебного материала	1	
	1. Разработка функционального интерфейса приложения. Создание интерфейса приложения.		
	2. Разработка функциональной схемы работы приложения.		
Тема 5.5. Этапы разработки приложений	Содержание учебного материала	1	
	1. Разработка приложения.		
	2. Проектирование объектно-ориентированного приложения.		

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающегося, курсовая работа (проект)	Объем в часах	Коды компетенций, формированию которых способствует элемент программы Воспитательный компонент - ЛР
	3. Создание интерфейса пользователя.		
	4. Тестирование, отладка приложения.		
Тема 5.6. Иерархия классов.	Содержание учебного материала	1	
	1. Классы ООП: виды, назначение, свойства, методы, события.		
	2. Перегрузка методов.		
	3. Тестирование и отладка приложения.		
	4. Решение задач		
Тематика практических занятий и лабораторных работ:		40	ОК 01., ОК 02., ОК 03., ОК 05., ОК 09., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 11.1., ПК 11.2., ПК 11.3., ПК 11.4., ПК 2.1., ПК 2.2., ПК 2.3., ПК 2.4., ПК 2.5. ЛР 5, ЛР 6, ЛР 7, ЛР 8, ЛР 9, ЛР 10, ЛР 13
1. Знакомство со средой программирования.			
2. Составление программ линейной структуры.			
3. Составление программ разветвляющейся структуры.			
4. Составление программ циклической структуры			
5. Обработка одномерных массивов.			
6. Обработка двумерных массивов.			
7. Работа со строками.			
8. Работа с данными типа множество.			
9. Файлы последовательного доступа.			
10. Типизированные файлы.			
11. Нетипизированные файлы.			
12. Организация процедур.			
13. Организация функций.			
14. Применение рекурсивных функций.			
15. Программирование модуля.			
16. Создание библиотеки подпрограмм.			
17. Использование указателей для организации связанных списков.			
18. Изучение интегрированной среды разработчика.			
19. Создание проекта с использованием компонентов для работы с текстом.			
20. Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени.			
21. События компонентов (элементов управления), их сущность и назначение.			

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающегося, курсовая работа (проект)	Объем в часах	Коды компетенций, формированию которых способствует элемент программы Воспитательный компонент - ЛР
	22. Создание процедур на основе событий. 23. Создание проекта с использованием кнопочных компонентов. 24. Создание проекта с использованием компонентов стандартных диалогов и системы меню. 25. Разработка функциональной схемы работы приложения. 26. Разработка оконного приложения с несколькими формами. 27. Разработка игрового приложения. 28. Создание процедур обработки событий. Компиляция и запуск приложения. 29. Разработка интерфейса приложения. 30. Тестирование, отладка приложения. 31. Классы ООП: виды, назначение, свойства, методы, события. 32. Объявления класса. 33. Создание наследованного класса. 34. Программирование приложений. 35. Перегрузка методов.		
Самостоятельная работа	<i>Составление опорных конспектов, рефераты по заданной теме, изучение и закрепление теоретического материала, подготовка к практическим занятиям, написание курсовой работы</i>	130	
Промежуточная аттестация - экзамен		6	
Всего:		200	

Темы курсовых работ:

1. Написать программу, которая формирует двумерную матрицу чисел размерностью $n \times n$. Затем формирует два одномерных массива, причем в один записываются элементы, расположенные на главной диагонали и выше, в другой – элементы матрицы, лежащие ниже главной диагонали и выводит все массивы на экран. Осуществить бинарный поиск заданного элемента во втором массиве. Результаты поиска вывести на экран.
2. Дана квадратная матрица размерностью $n \times n$. Написать программу формирования последовательности В, элементами которой являются элементы таблицы А, расположенные над главной диагональю. В полученной последовательности найти три наименьших элемента. Если количество элементов между 1-м и 2-м элементами совпадает с количеством элементов между 2-м и 3-м элементами, то поменять их местами, сохранив порядок следования (без использования дополнительного массива). Преобразованный массив вывести на экран.
3. Написать программу, которая предлагает пользователю некоторый список функций для построения графиков: $y=x^4/(x^3+1)$; $y=ax^2+bx+c$; $y=\sin(x)+b$; $y=1/(x^2+x+1)$. После выбора функции, задания коэффициентов и отрезка, на котором выполняется построение, программа строит соответствующий график. Затем значение коэффициентов и положение графика можно изменить (например, с помощью клавиш управления курсором), после чего график перестраивается и записывается обновленное уравнение кривой.
4. Написать программу, позволяющую отсортировать массив двумя способами (обменная и выбором). Обеспечить определение и отображение времени сортировки каждым методом. Предусмотреть операции создания файла, записи данных в файл и чтения файла для обработки.
5. Написать программу тестирования знаний студентов по дисциплине Программирование и основы алгоритмизации. Вопросы для тестирования должны выбираться из файла случайным образом. Программа должна осуществлять подсчет правильных ответов и выводить на экран результаты тестирования (общее число вопросов, количество правильных ответов, оценку).
6. Написать программу быстрой сортировки массива, двоичного поиска повторяющихся элементов и замены их пользователем. Предусмотреть операции создания файла, записи данных в файл и чтения файла для обработки.
7. Написать программу, вычисляющую определитель квадратной матрицы размерностью $n \times n$. Содержимое матрицы должно отображаться на экране. Обеспечить редактирование элементов матрицы, автоматический подсчет определителя и вывод его значения на экран.
8. Написать программу шифрования и дешифрования текстовых файлов. В основу шифрования положить коэффициент смещения букв, в результате использования которого, например, буква А меняется на Д. Исходный и результирующий текст вывести на экран и в файл.
9. Написать программу, моделирующую тренажер по устному счету. Пользователь вводит разрядность операндов, тип операции (+ - * /) на множестве натуральных чисел и количество примеров. Компьютер генерирует случайным образом операнды, вычисляет результат операции и выводит пользователю серию примеров, в каждом из которых один из операндов или результат замаскирован. Пользователь вводит пропущенное число, компьютер проверяет правильность и ведет статистику ошибок.
10. Написать программу шифрования и дешифрования данных, содержащихся в текстовых файлах методом прямой замены. Зашифрованные данные сохранить в файле, исходный и результирующий отобразить на экране.

11. Написать программу, которая будет исправлять неточности редактирования текстового файла. Программа должна убирать знаки пробела перед знаком препинания, ставить один пробел после знака препинания и между словами, исправлять первую букву предложения на заглавную. Информация считывается из текстового файла и записывается в другой текстовый файл на жестком диске.
12. Написать программу, обеспечивающую создание списка абитуриентов. Предусмотреть ввод фамилии, имени и отчества, года рождения, оценок за вступительные экзамены (математика, физика, русский язык). Программа должна позволять сохранить список в файл, загрузить его и произвести сортировку по следующим полям в порядке убывания: общий балл за экзамен; фамилия, имя, отчество; год рождения.
13. Написать программу демонстрации графических возможностей Turbo C++ и обучающую работе с основными графическими процедурами и функциями. Программа должна контролировать усвоение изученного материала (в виде теста или в какой-либо другой форме).
14. Написать программу, которая по введенному пользователем вектору чисел строит диаграммы различных видов в графическом режиме. В интерфейсе предусмотреть ввод вектора чисел и вывод диаграммы (не более 5-ти видов).
15. Выбрать какой-либо школьный предмет (информатика, математика и т.д.), подобрать ребусы по нему и предложить их для решения. Программа должна позволить выбрать тот или иной ребус, проконтролировать его решение и подвести итоги при завершении работы.
16. Написать программу, обеспечивающую создание файла для хранения и обработки данных о студентах учебной группы. Обработка должна включать операции добавления, удаления, корректировки записей и просмотра данных в табличном отображении. Записи организовать в виде связанного списка. Перечисленные операции реализовать с помощью стандартных операторов для заданного абстрактного типа данных.
17. Написать программу, обеспечивающую создание файла для хранения и обработки данных о мониторах. Обработка должна включать операции добавления, удаления, корректировки записей и просмотра данных в табличном отображении. Записи организовать в виде стека. Перечисленные операции реализовать с помощью стандартных операторов для заданного абстрактного типа данных.
18. Написать программу для решения систем линейных уравнений методом Гаусса. Исходные данные и результаты решения вывести в файл. Исходные данные разместить в динамической области памяти.
19. Написать программу для решения систем линейных уравнений итерационным методом Зейделя. Исходные данные и результаты решения вывести в файл. Исходные данные разместить в динамической области памяти.
20. Написать программу для решения нелинейных уравнений методом половинного деления и методом Ньютона. Сравнить методы по быстродействию.
21. Написать программу вычисления интегралов методами прямоугольников, трапеций и Симпсона. Сравнить время реализации методов при заданной тонности.
22. Написать программу решения системы линейных уравнений методом Крамера. Количество переменных до 20. Программа должна выводить результаты решения задачи, а также время, затраченное на расчет с точностью до сотых долей секунд.
23. Составить программу, которая находит среднее арифметическое элементов непустого однонаправленного списка вещественных чисел, заменяет все вхождения числа x на число y , меняет местами первый и последний элементы списка, проверяет упорядочены ли числа в списке по возрастанию. Результирующий массив записать в файл.
24. Дана квадратная матрица размерностью $n \times n$. Написать программу формирования последовательности A , элементами которой являются элементы таблицы B , расположенные над второстепенной диагональю. В полученной последовательности найти три наибольших элемента. Если

количество элементов между 2-м и 3-м элементами совпадает с количеством элементов между 1-м и 2-м элементами, то поменять их местами, сохранив порядок следования (без использования дополнительного массива). Преобразованный массив вывести на экран и в файл.

25. Написать программу, позволяющую отсортировать массив двумя способами (вставками и выбором). Обеспечить определение и отображение времени сортировки каждым методом. Предусмотреть операции создания файла, записи данных в файл и чтения файла для обработки.

3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ:

3.1. Требования к минимальному материально-техническому обеспечению:

Реализация программы осуществляется с применением исключительно электронного обучения, дистанционных образовательных технологий с учетом требований федерального законодательства.

В колледже создана единая электронная информационно-образовательная среда (далее – ЭИОС), представляющая собой совокупность электронных информационных и образовательных ресурсов, информационных и телекоммуникационных технологий, соответствующих технических и технологических средств, обеспечивающая освоение обучающимися образовательных программ или их частей, а также взаимодействие всех субъектов образовательного процесса. Основным элементом ЭИОС – система управления обучением (LMS) «MOODLE», в которой внедрены личный кабинет студента и личный кабинет преподавателя. При обучении инвалидов и лиц с ограниченными возможностями здоровья дистанционные образовательные технологии предусматривают возможность приема-передачи информации в доступных для них формах. Образовательная платформа Moodle поддерживает функцию голосового чтения с экрана, экранную лупу и клавиатуру.

Для использования в образовательном процессе, в том числе для реализации программы библиотечный фонд образовательной организации имеет электронные образовательные и информационные ресурсы.

3.2. Учебно-методическое обеспечение обучения:

1. Ляхович, В. Ф., Основы информатики : учебник / В. Ф. Ляхович, В. А. Молодцов, Н. Б. Рыжикова. — Москва : КноРус, 2023. — 347 с. — ISBN 978-5-406-11093-5. — URL: <https://book.ru/book/947649>. — Текст : электронный.
2. Макарова, Н. В., Основы программирования : учебник и практикум / Н. В. Макарова, Ю. Н. Нилова, С. Б. Зеленина, Е. В. Лебедева. — Москва : КноРус, 2023. — 452 с. — ISBN 978-5-406-11053-9. — URL: <https://book.ru/book/947384>. — Текст : электронный.

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. <ul style="list-style-type: none"> • 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>«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>Примеры форм и методов контроля и оценки</p> <ul style="list-style-type: none"> • Тестирование • Контрольная работа • Самостоятельная работа. • Защита реферата • Наблюдение за выполнением практического задания. (деятельностью студента) • Оценка выполнения практического задания(работы)
<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. <ul style="list-style-type: none"> • 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, инкапсуляция и полиморфизма, наследования и переопределения. 	<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 ошибки.</p> <p>«Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.</p>	<ul style="list-style-type: none"> • Подготовка и выступление с докладом, сообщением, презентацией... • Решение ситуационной задачи. <p>Написание курсового проекта</p>

