

**Автономная некоммерческая организация
профессионального образования
«Пятигорский техникум экономики и инновационных технологий»
(АНО ПО «ПТЭИТ»)**



РАБОЧАЯ ПРОГРАММА ПО ДИСЦИПЛИНЕ

ОП.04 Основы алгоритмизации и программирования

для студентов специальности

09.02.07 Информационные системы и программирование
Квалификация: Специалист по информационным системам

г. Пятигорск, 2024г.

СОДЕРЖАНИЕ

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

1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ «ОП.04.ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

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:

В результате изучения дисциплины обучающиеся должны освоить следующие общие (ОК) и профессиональные (ПК) компетенции:

ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.

ОК 09. Использовать информационные технологии в профессиональной деятельности.

ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке.

ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.

ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.

ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.

ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

Код ПК, ОК	Умения	Знания
ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5 ПК 2.4, 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.	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

	<p>Выполнять проверку, отладку кода программы.</p>	<p>классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения</p>
--	--	--

2. СТРУКТУРА УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем в часах
Объем образовательной программы	183
в том числе:	
теоретическое обучение	86
практические занятия	86
<i>Самостоятельная работа</i>	-
Консультации	2
Промежуточная аттестация в форме экзамена	9

В результате освоения рабочей программы обучающийся должен достичь следующих личностных результатов:

ЛР 1 Осознающий себя гражданином и защитником великой страны.

ЛР 2 Проявляющий активную гражданскую позицию, демонстрирующий приверженность принципам честности, порядочности, открытости, экономически активный и участвующий в студенческом и территориальном самоуправлении, в том числе на условиях добровольчества, продуктивно взаимодействующий и участвующий в деятельности общественных организаций.

ЛР 3 Соблюдающий нормы правопорядка, следующий идеалам гражданского общества, обеспечения безопасности, прав и свобод граждан России. Лояльный к установкам и проявлениям представителей субкультур, отличающий их от групп с деструктивным и девиантным поведением. Демонстрирующий неприятие и предупреждающий социально опасное поведение окружающих.

ЛР 4 Проявляющий и демонстрирующий уважение к людям труда, осознающий ценность собственного труда. Стремящийся к формированию в сетевой среде лично и профессионального конструктивного «цифрового следа».

ЛР 5 Демонстрирующий приверженность к родной культуре, исторической памяти на основе любви к Родине, родному народу, малой родине, принятию традиционных ценностей многонационального народа России.

ЛР 6 Проявляющий уважение к людям старшего поколения и готовность к участию в социальной поддержке и волонтерских движениях.

ЛР 7 Осознающий приоритетную ценность личности человека; уважающий собственную и чужую уникальность в различных ситуациях, во всех формах и видах деятельности.

ЛР 8 Проявляющий и демонстрирующий уважение к представителям различных этнокультурных, социальных, конфессиональных и иных групп. Сопричастный к сохранению, преумножению и трансляции культурных традиций и ценностей многонационального российского государства.

ЛР 9 Соблюдающий и пропагандирующий правила здорового и безопасного образа жизни, спорта; предупреждающий либо преодолевающий зависимости от алкоголя, табака, психоактивных веществ, азартных игр и т.д. Сохраняющий психологическую устойчивость в ситуативно сложных или стремительно меняющихся ситуациях.

ЛР 10 Заботящийся о защите окружающей среды, собственной и чужой безопасности, в том числе цифровой.

ЛР 11 Проявляющий уважение к эстетическим ценностям, обладающий основами эстетической культуры.

ЛР 12 Принимающий семейные ценности, готовый к созданию семьи и воспитанию детей; демонстрирующий неприятие насилия в семье, ухода от родительской ответственности, отказа от отношений со своими детьми и их финансового содержания.

ЛР 13 Демонстрирующий умение эффективно взаимодействовать в команде, вести диалог, в том числе с использованием средств коммуникации

ЛР 14 Демонстрирующий навыки анализа и интерпретации информации из различных источников с учетом нормативно-правовых норм

ЛР 15 Демонстрирующий готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности.

2.1 Тематический план и содержание учебной дисциплины «ОП.04.ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

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	
1	2	3	4	
Раздел 1.	<i>Введение в программирование</i>	12	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5	
Тема 1.1. Языки программирования	Содержание учебного материала	6		
	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. В том числе практических занятий и лабораторных работ			
Тема 1.2. Типы данных	Содержание учебного материала	6		
	1. Типы данных. Простые типы данных. Производные типы данных. Структурированные типы данных. В том числе практических занятий и лабораторных работ			
Раздел 2.	Содержание учебного материала	36		ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10
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.		ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	5. Структурированный тип данных – множество. Операции над множествами.		
	6. Комбинированный тип данных – запись. Файлы последовательного доступа. Файлы прямого доступа		
	В том числе практических занятий и лабораторных работ		
Раздел 3.	Содержание учебного материала	28	OK 1 OK 2 OK 4 OK 5 OK 9 OK 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
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.	12	
	2. Рекурсия. Программирование рекурсивных алгоритмов.		
	В том числе практических занятий и лабораторных работ		
Тема 3.2. Структуризация в программировании	Содержание учебного материала	4	
	1. Основы структурного программирования. Методы структурного программирования.		
	В том числе практических занятий и лабораторных работ		
Тема 3.3. Модульное программирование	Содержание учебного материала	12	
	1. Модульное программирование. Понятие модуля. Структура модуля. Компиляция и компоновка программы.		
	2. Стандартные модули.		
	В том числе практических занятий и лабораторных работ		
Раздел 4	<i>Основные конструкции языков программирования</i>	12	OK 1 OK 2 OK 4 OK 5 OK 9 OK 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
Тема 4.1 Указатели.	Содержание учебного материала	12	
	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.		
	В том числе практических занятий и лабораторных работ		
Раздел 5	Содержание учебного материала	60	OK 1 OK 2
Тема 5.1 Основные принципы	1. История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс.	12	

объектно-ориентированного программирования (ООП)	2. Основные принципы ООП: инкапсуляция, наследование, полиморфизм.		ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	3. Классы объектов. Компоненты и их свойства.		
	4. Событийно-управляемая модель программирования. Компонентно-ориентированный подход.		
	В том числе практических занятий и лабораторных работ		
Тема 5.2 Интегрированная среда разработчика.	Содержание учебного материала	12	
	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.		
	6. Настройка среды и параметров проекта.		
	В том числе практических занятий и лабораторных работ		
Тема 5.3. Визуальное событийно-управляемое программирование	Содержание учебного материала	10	
	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.		
	3. События компонентов (элементов управления), их сущность и назначение. Создание процедур на основе событий.		
	В том числе практических занятий и лабораторных работ		
Тема 5.4 Разработка оконного приложения	Содержание учебного материала	12	
	1. Разработка функционального интерфейса приложения. Создание интерфейса приложения.		
	2. Разработка функциональной схемы работы приложения.		
	3. Разработка игрового приложения.		
	В том числе практических занятий и лабораторных работ		

	Самостоятельная работа обучающихся		
Тема 5.5 Этапы разработки приложений	Содержание учебного материала	10	
	1. Разработка приложения.		
	2. Проектирование объектно-ориентированного приложения.		
	3. Создание интерфейса пользователя.		
	4. Тестирование, отладка приложения.		
	В том числе практических занятий и лабораторных работ		
Тема 5.6 Иерархия классов.	Содержание учебного материала	4	
	1. Классы ООП: виды, назначение, свойства, методы, события.		
	2. Перегрузка методов.		
	3. Тестирование и отладка приложения.		
	4. Решение задач		
	В том числе практических занятий и лабораторных работ		
Тематика практических занятий и лабораторных работ:			
Знакомство со средой программирования.			
Составление программ линейной структуры.			
Составление программ разветвляющейся структуры.			
Составление программ циклической структуры			
Обработка одномерных массивов.			
Обработка двумерных массивов.			
Работа со строками.			
Работа с данными типа множество.			
Файлы последовательного доступа.			
Типизированные файлы.			
Нетипизированные файлы.			
Организация процедур.			
Организация функций.			
Применение рекурсивных функций.			
Программирование модуля.			
Создание библиотеки подпрограмм.			
Использование указателей для организации связанных списков.			

<p>Изучение интегрированной среды разработчика.</p> <p>Создание проекта с использованием компонентов для работы с текстом.</p> <p>Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени.</p> <p>События компонентов (элементов управления), их сущность и назначение.</p> <p>Создание процедур на основе событий.</p> <p>Создание проекта с использованием кнопочных компонентов.</p> <p>Создание проекта с использованием компонентов стандартных диалогов и системы меню.</p> <p>Разработка функциональной схемы работы приложения.</p> <p>Разработка оконного приложения с несколькими формами.</p> <p>Разработка игрового приложения.</p> <p>Создание процедур обработки событий. Компиляция и запуск приложения.</p> <p>Разработка интерфейса приложения.</p> <p>Тестирование, отладка приложения.</p> <p>Классы ООП: виды, назначение, свойства, методы, события.</p> <p>Объявления класса.</p> <p>Создание наследованного класса.</p> <p>Программирование приложений.</p> <p>Перегрузка методов.</p>		
Итого во взаимодействии с преподавателем	172	
Консультации	2	
Промежуточная аттестация	9	
Всего:	183	

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ «ОП.04.ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

3.1. Для реализации программы учебной дисциплины должны быть предусмотрены следующие специальные помещения:

Лаборатория «**Программирование баз данных**», оснащенная:

- Автоматизированные рабочие места обучающихся (процессор Core i3, оперативная память объемом 8 Гб) – 12 шт.,
- Автоматизированное рабочее место преподавателя (процессор Core i3, оперативная память 8 Гб) - 1 шт.
- Сервер в лаборатории (8-х ядерный процессор с частотой 3 ГГц, оперативная память объемом 16 Гб, жесткие диски общим объемом 1 Тб, программное обеспечение: WindowsServer 2012)
- Проектор – 1 шт.,
- Экран- 1 шт.,
- Маркерная доска- 1 шт.
- **Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО:**
 - EclipseIDEforJavaEEDevelopers
 - NETFrameworkJDK 8,
 - MicrosoftSQLServerExpressEdition,
 - MicrosoftVisioProfessional,
 - MicrosoftVisualStudio,
 - MySQLInstallerforWindows
 - NetBeans
 - SQLServerManagementStudio
 - MicrosoftSQLServerJavaConnector
 - AndroidStudio
 - IntelliJIDEA.

3.2. Информационное обеспечение реализации программы

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

3.2.1. Печатные и электронные изданияиздания

Окулов, С.М. Основы программирования : учебник / Окулов С.М. 10-е изд. — Москва : Лаборатория знаний, 2020. — 337 с. — ISBN 978-5-00101-759-2. — URL: <https://book.ru/book/936416> (дата обращения: 24.02.2021). — Текст : электронный.

Коренская, И. Н. Основы алгоритмизации и программирования на языке Паскаль. Лабораторный практикум : учебное пособие для спо / И. Н. Коренская. — Санкт-Петербург : Лань, 2021. — 128 с. — ISBN 978-5-8114-6521-7. — Текст : электронный // Лань : электронно-библиотечная

TML, CSS, Scratch, Python. Моя первая книга по программированию : руководство / С. В. Голиков, Е. В. Дубовик, Г. С. Русин, Ю. А. Иркова. — Санкт-Петербург : Наука и Техника, 2018. — 336 с. — ISBN 978-5-94387-754-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/108276> (дата обращения: 24.02.2021)..

Конова, Е. А. Алгоритмы и программы. Язык C++ : учебное пособие / Е. А. Конова, Г. А. Поллак. — Санкт-Петербург : Лань, 2020. — 384 с. — ISBN 978-5-8114-5521-8. —

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ «ОП.04.ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

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>«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</p>	<p>Примеры форм и методов контроля и оценки</p> <ul style="list-style-type: none"> • Компьютерное тестирование на знание терминологии по теме; • Тестирование.... • Контрольная работа • Самостоятельная работа. <ul style="list-style-type: none"> • Защита реферата.... • Семинар • Защита курсовой работы (проекта) <ul style="list-style-type: none"> • Выполнение проекта; • Наблюдение за выполнением практического задания. (деятельностью студента) <ul style="list-style-type: none"> • Оценка выполнения практического задания(работы) • Подготовка и выступление с докладом, сообщением, презентацией... <ul style="list-style-type: none"> • Решение ситуационной задачи....
<p><i>Перечень знаний, осваиваемых в рамках дисциплины:</i></p> <ul style="list-style-type: none"> • Понятие алгоритмизации, свойства алгоритмов, общие принципы построения 	<p>характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий</p>	

<p>алгоритмов, основные алгоритмические конструкции.</p> <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>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>	
---	---	--

**Автономная некоммерческая организация
профессионального образования
«Пятигорский техникум экономики и инновационных технологий»
(АНО ПО «ПТЭИТ»)**



УТВЕРЖДАЮ:
Директор АНО ПО «ПТЭИТ»

Ш.М.Исаев
«31» мая 2024 г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ПО ДИСЦИПЛИНЕ**

ОП.04 Основы алгоритмизации и программирования

для студентов специальности

09.02.07 Информационные системы и программирование

Квалификация: Специалист по информационным системам

г. Пятигорск, 2024г.

ФОС учебной дисциплины разработана на основе Федерального государственного образовательного стандарта (ФГОС) по специальности среднего профессионального образования (СПО) 09.02.07 Информационные системы и программирование (Приказ Минобрнауки России от 09.12.2016 № 1547)

Организация-разработчик: Автономная некоммерческая организация профессионального образования «Пятигорский техникум экономики и инновационных технологий»

Разработчик: Кононюк Т.Д., преподаватель базовой квалификационной категории АНО ПО «ПТЭИТ»

РАССМОТРЕНА
отделением информационно-технических
дисциплин
Протокол №9 от 24.05.2024г.
Зав.отделением _____ Кононюк Т.Д.

СОГЛАСОВАНА
на заседании УМС
Протокол № 6 от 30.05.2024
_____ Кодякова О.А.

Рецензенты

Аваков А.А.- преподаватель АНО ПО «ПТЭИТ»

Баранская М.Ф. – преподаватель информационных дисциплин АЧОУ ВО «Институт Управления, Бизнеса и Права», г. Пятигорск

2. Паспорт фонда оценочных средств УД Основы программирования

2.1 ФОС позволяет оценивать ОК и ПК:

В результате изучения дисциплины обучающиеся должны освоить следующие общие (ОК) и профессиональные (ПК) компетенции:

ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.

ОК 09. Использовать информационные технологии в профессиональной деятельности.

ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке.

ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.

ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.

ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.

ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

<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	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
--	--	---

У1. – работать в среде программирования;

У2. – реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.

2.3 ФОС позволяет оценивать усвоение **знаний**:

31. - этапы решения задачи на компьютере;

32. - типы данных;

33. - базовые конструкции изучаемых языков программирования; 34. - принципы структурного и модульного программирования;

35. - принципы объектно-ориентированного программирования.

2.4 Кодификатор оценочных средств:

Например:

2.5. Распределение оценочных средств по элементам освоенных умений, усвоенных знаний и их использование в практической деятельности для контроля сформированности компетенций в рамках тем/разделов УД по видам аттестации

Контролируемые разделы (темы) в порядке поэтапного освоения УД в рамках ППССЗ	Текущий контроль							
	Компетенции	Результаты обучения						
		Освоенные умения:		Усвоенные знания				
		У-1	У-2	3-1	3-2	3-3	3-4	3-5
Раздел 1. <i>Введение в программирование</i>								
Тема 1.1. Языки программирования	ОК.1, ОК.2, ОК.3, ОК.4 ПК.1.1 ПК.1.2				1		1	
Тема 1.2. Типы данных	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 , ПК 1.1- ПК 1.5, ПК 2.4, 2.5	1	1	1	1	1	1	
Раздел 2.								
Тема 2.1. Операторы языка программирования	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 , ПК 1.1- ПК 1.5, ПК 2.4, 2.5		1	1	1	1	1	
Раздел 3.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 , ПК 1.1- ПК 1.5, ПК 2.4, 2.5							
Тема 3.1. Процедуры и функции	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10		1	1	1	1	1	

	, ПК 1.1- ПК 1.5, ПК 2.4, 2.5							
Тема 3.2. Структуризация в программировании	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 , ПК 1.1- ПК 1.5, ПК 2.4, 2.5		1	1	1	1	1	
Тема 3.3. Модульное программирование	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 , ПК 1.1- ПК 1.5, ПК 2.4, 2.5	1		1	1	1	1	
Раздел 4								
Тема 4.1 Указатели.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 , ПК 1.1- ПК 1.5, ПК 2.4, 2.5	1	1	1	1	1		1
Раздел 5	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 , ПК 1.1- ПК 1.5, ПК 2.4, 2.5							
Тема 5.1 Основные принципы объектно-ориентированного программирования (ООП)	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 , ПК 1.1- ПК 1.5, ПК 2.4, 2.5	1	1	1	1	1	1	
Тема 5.2 Интегрированная среда разработчика.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10	1	1	1	1	1	1	

	, ПК 1.1- ПК 1.5, ПК 2.4, 2.5							
Тема 5.3. Визуальное событийно-управляемое программирование	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 , ПК 1.1- ПК 1.5, ПК 2.4, 2.5	1	1	1	1	1	1	1
Тема 5.4 Разработка оконного приложения	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 , ПК 1.1- ПК 1.5, ПК 2.4, 2.5	1	1	1	1	1	1	
Тема 5.5 Этапы разработки приложений	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 , ПК 1.1- ПК 1.5, ПК 2.4, 2.5	1	1	1	1	1	1	1
Тема 5.6 Иерархия классов.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 , ПК 1.1- ПК 1.5, ПК 2.4, 2.5	1	1	1	1	1	1	1

3. Комплекты контрольно - оценочных средства по видам аттестации

3.1 КОС/КИМ для текущего контроля

Оценочные средства	Комплекты контрольных заданий или иные материалы, необходимые для оценки знаний, умений, навыков и опыта практической деятельности, характеризующие этапы формирования компетенций
Практическая работа	Методические рекомендации по выполнению практических работ

3.2 КОС/КИМ для промежуточной аттестации

Форма проведения например	Комплекты контрольных заданий или иные материалы, необходимые для оценки знаний, умений, навыков и опыта практической деятельности, характеризующие этапы формирования компетенций
Дифференцированный зачет	- вопросы и задания для подготовки к диф.зачету; - перечень билетов; - критерии и шкала оценивания ответа обучающегося.

1. Перечень практических работ и вариантов заданий.

№ раздела дисциплины	Наименование практической работы	Цель работы	Формы текущего контроля
1	2	3	4
Раздел 1.			
Тема 1.1.	Практическая работа № 1. «Составление блоксхем линейных, разветвляющихся и циклических алгоритмов».	Повторить и закрепить навыки составления блок схем линейных, разветвляющихся и циклических алгоритмов	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.
	Практическая работа № 2. «Составление блоксхем алгоритма сортировки данных методом пузырька. Анализ трудоемкости алгоритма»	Повторить навыки построения блок-схем алгоритмов. Научиться составлять блоксхемы алгоритмов сортировки и анализировать их трудоемкость с заданными начальными условиями.	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.

Тема 1.2.	Практическая работа № 3. «Описание жизненного цикла программного продукта».	Научиться применять на практике сведения о моделях жизненного цикла программного продукта и определять подходящую модель жизненного цикла для разработки ПО.	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.
Раздел 2. Тема 2.1.	Практическая работа № 4. «Написание программ по составленным блок-схемам».	Ознакомиться с средой разработки. Научиться писать код программы по составленной блок-схеме.	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.
Тема 2.2.	Практическая работа № 5. «Написание программы с использованием указателей и ссылок».	Сформировать основы работы с указателями и ссылками. Закрепить сведения об указателях и ссылках, полученные на лекции. Проанализировать на практике механизм работы ссылок и указателей.	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.
Тема 2.3.	Практическая работа № 6. «Работа с одномерными массивами».	Закрепить на практике сведения об одномерных массивах, полученные на лекции. Научиться выполнять операции с одномерными массивами.	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.
	Практическая работа № 7. «Работа с двумерными массивами».	Закрепить на практике сведения об двумерных массивах, полученные на лекции. Научиться выполнять операции с двумерными массивами.	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.
Тема 2.4	Практическая работа № 8. «Работа со строками».	Закрепить на практике сведения о строках, полученные на лекции. Научиться выполнять операции со строками. Изучить особенности работы со строками в С.	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.

Тема 2.5.	Практическая работа № 9. «Работа с файлами».	Сформировать основы работы с файлами в С. Научиться писать программы для работы с файлами.	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.
Раздел 3.			Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.
Тема 3.1.	Практическая работа № 10. «Создание и использование объектов»	Закрепить теоретические сведения об ООП, полученные на лекции. Научиться создавать объекты и работать с ними в среде программирования.	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.
	Практическая работа № 11. «Изучение интегрированной среды разработчика».	Закрепить знания о создании объектов и работе с ними в среде программирования. Сформировать навыки работы с отладчиком.	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.
	Практическая работа №12. «Создание простого проекта».	Закрепить навыки работы в среде разработчика. Улучшить навыки работы с отладчиком и компилятором. Создать проект в выбранной среде программирования и проанализировать его работу.	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.
	Практическая работа № 13. «Тестирование, отладка проекта. Создание документации».	Улучшить навыки работы с отладчиком. Выполнить тестирование разработанного проекта. Проанализировать результаты тестирования и составить отчет о проделанной работе.	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.
	Практическая работа № 14. «Создание проекта с использованием кнопок и текстовых полей».	Закрепить навыки работы со средой разработки. Сформировать навыки работы с кнопками и текстовыми полями.	Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.

<p>Практическая работа № 15. «Создание проекта с меню и справкой для пользователя».</p>	<p>Закрепить навыки работы с кнопками и текстовыми полями. Научиться разрабатывать меню и справку пользователя средствами среды разработки.</p>	<p>Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.</p>
<p>Практическая работа № 16. «Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени».</p>	<p>Закрепить навыки работы с кнопками и текстовыми полями. Сформировать навыки работы с компонентами ввода, отображения чисел, даты и времени.</p>	<p>Оценка за выполнение практического задания, оценка за устный дифференцированный опрос.</p>

Варианты заданий

Практическая работа № 1

Вариант 1

- 1) Составить блок-схему алгоритма для нахождения суммы двух целых чисел.
- 2) Составить блок-схему алгоритма для нахождения наибольшего из двух чисел.
- 3) Составить блок-схему алгоритма для нахождения суммы чисел от 1 до 10, используя цикл с предусловием.
- 4) Составить блок-схему алгоритма для нахождения факториала числа 6, используя цикл с параметром.
- 5) Составить блок-схему алгоритма для нахождения разности чисел от 10 до 0, используя цикл с постусловием.

Вариант 2

- 1) Составить блок-схему алгоритма для нахождения разности двух целых чисел.
- 2) Составить блок-схему алгоритма для нахождения наименьшего из двух чисел.
- 3) Составить блок-схему алгоритма для нахождения суммы чисел от 1 до 10, используя цикл с постусловием.
- 4) Составить блок-схему алгоритма для нахождения факториала числа 7, используя цикл с параметром.
- 5) Составить блок-схему алгоритма для нахождения разности чисел от 10 до 0, используя цикл с предусловием.

Вариант 3

- 1) Составить блок-схему алгоритма для нахождения произведения двух целых чисел.

- 2) Составить блок-схему алгоритма для определения равенства двух чисел.
- 3) Составить блок-схему алгоритма для нахождения суммы чисел от 1 до 10, используя цикл с параметром.
- 4) Составить блок-схему алгоритма для нахождения факториала числа 5, используя цикл с предусловием.
- 5) Составить блок-схему алгоритма для нахождения разности чисел от 10 до 0, используя

цикл с постусловием.

Вопросы для самоконтроля

1. Перечислить блоки для построения блок-схем.
2. Описать принцип работы цикла с постусловием.
3. Описать принцип работы цикла с предусловием.
4. Описать принцип работы цикла с параметром.
5. Сравнить циклы с предусловием и постусловием.
6. Сравнить цикл с параметром и цикл с предусловием.

Практическая работа № 2.

Вариант 1

- 1) Составить блок-схему алгоритма сортировки методом пузырька.
- 2) Используя составленную блок-схему, отсортируйте последовательности чисел: 1 2 3 4 5 6 7 8 9 10 и 2 3 6 5 1 8 9 12 0 -1. Проанализировать сложность алгоритма для обеих последовательностей.

Вариант 2

- 1) Составить блок-схему алгоритма сортировки методом пузырька.
- 2) Используя составленную блок-схему, отсортируйте последовательности чисел: 10 9 8 7 6 5 4 3 2 1 и 9 8 3 5 1 0 7 2 4 5. Проанализировать сложность алгоритма для обеих последовательностей.

Вариант 3

- 1) Составить блок-схему алгоритма сортировки методом пузырька.
- 2) Используя составленную блок-схему, отсортируйте последовательности чисел: 1 2 3 4 5 6 7 8 9 10 и 10 9 8 7 6 5 4 3 2 1. Проанализировать сложность алгоритма для обеих последовательностей.

Вопросы для самоконтроля

1. Описать принцип работы пузырьковой сортировки.
2. Описать принцип работы сортировки вставками.
3. Сравнить сортировку вставками и пузырьковую сортировку.
4. Проанализировать результаты практической работы и определить, в каких случаях сортировка методом пузырька наименее эффективна.

Практическая работа № 3.

Вариант 1

- 1) Проанализировать стандарты ISO 15504 и ISO/IEC12207, определить актуальность, сходства и различия.
- 2) Проанализировать стандарт ГОСТ Р ИСО/МЭК 12207 и выписать его основные положения.
- 3) Описать процесс разработки ПО при использовании каскадной модели жизненного цикла.

Вариант 2

- 1) Проанализировать стандарты ISO 15504 и ISO/IEC12207, определить актуальность, сходства и различия.
- 2) Проанализировать стандарт ГОСТ Р ИСО/МЭК 12207 и выписать его основные положения.
- 3) Описать процесс разработки ПО при использовании спиральной модели жизненного цикла.

Вариант 3

- 1) Проанализировать стандарты ISO 15504 и ISO/IEC12207, определить актуальность, сходства и различия.
- 2) Проанализировать стандарт ГОСТ Р ИСО/МЭК 12207 и выписать его основные положения.
- 3) Описать процесс разработки ПО при использовании итерационной модели жизненного цикла.

Вопросы для самоконтроля

1. Дать определение понятию «жизненный цикл программного продукта».
2. Сравнить каскадную и спиральную модели жизненного цикла. Определить проекты, для которых использование каскадной модели будет лучше.
3. Сравнить спиральную и итерационную модели. Определить проекты, для которых применение спиральной модели будет лучше.
4. Сравнить каскадную и итерационную модели. Определить проекты, для которых применение итерационной модели будет лучше.
5. Описать этап проектирования программного продукта.
6. Описать этап постановки задачи.
7. Описать этап анализа требований.

Практическая работа № 4.

Вариант 1

- 1) Написать программу для нахождения суммы двух целых чисел, используя блок-схему, составленную на Практической работе №1.
- 2) Написать программу для нахождения наибольшего из двух чисел, используя блок-схему, составленную на Практической работе №1.
- 3) Написать программу для нахождения суммы чисел от 1 до 10, используя цикл с предусловием, с помощью блок-схемы, составленной на Практической работе №1.

- 4) Написать программу для нахождения факториала числа 6, используя цикл с параметром, с помощью блок-схемы, составленной на Практической работе №1.
- 5) Написать программу для нахождения разности чисел от 10 до 0, используя цикл с постусловием, с помощью блок-схемы, составленной на Практической работе №1.

Вариант 2

- 1) Написать программу для нахождения разности двух целых чисел, используя блок-схему, составленную на Практической работе №1.
- 2) Написать программу для нахождения наименьшего из двух чисел, используя блок-схему, составленную на Практической работе №1.
- 3) Написать программу для нахождения суммы чисел от 1 до 10, используя цикл с постусловием, с помощью блок-схемы, составленной на Практической работе №1.
- 4) Написать программу для нахождения факториала числа 7, используя цикл с параметром, с помощью блок-схемы, составленной на Практической работе №1.
- 5) Написать программу для нахождения разности чисел от 10 до 0, используя цикл с предусловием, с помощью блок-схемы, составленной на Практической работе №1.

Вариант 3

- 1) Написать программу для нахождения произведения двух целых чисел, используя блок-схему, составленную на Практической работе №1.
- 2) Написать программу для нахождения равенства двух чисел, используя блок-схему, составленную на Практической работе №1.
- 3) Написать программу для нахождения суммы чисел от 1 до 10, используя цикл с параметром, с помощью блок-схемы, составленной на Практической работе №1.
- 4) Написать программу для нахождения факториала числа 5, используя цикл с предусловием, с помощью блок-схемы, составленной на Практической работе №1.
- 5) Написать программу для нахождения разности чисел от 10 до 0, используя цикл с постусловием, с помощью блок-схемы, составленной на Практической работе №1.

Вопросы для самоконтроля

1. Описать синтаксис функции для ввода значений с клавиатуры.
2. Дать определение понятию «инициализация переменной».
3. Дать определение понятию «объявление переменной».
4. Описать синтаксис циклов с предусловием, постусловием и параметром.
5. Описать синтаксис условного оператора.

Практическая работа № 5.

- 1) Написать программу, выводящую на экран адрес ячейки памяти, занятой переменной.
- 2) Написать программу, передающую адрес ячейки памяти, занятой одной переменной, другой.
- 3) Написать программу, выводящую на экран значение переменной с помощью указателя.
- 4) Написать программу, передающую значение переменной по указателю.
- 5) Написать программу, выводящую на экран адрес первого элемента массива.

Вопросы для самоконтроля

1. Дать определение понятиям «ссылка» и «указатель».
2. Описать назначение указателей и ссылок.
3. Объяснить механизм индексации массива.

Практическая работа № 6.

- 1) Написать программу для создания одномерного массива из десяти элементов. Заполнить его значениями, равными индексу элемента массива.
- 2) Изменить написанную программу так, чтобы пользователь мог вводить значения элементов массива с клавиатуры.
- 3) Вывести заполненный с клавиатуры массив на экран.
- 4) Написать программу для нахождения суммы элементов массива.
- 5) Написать программу для сортировки элементов массива методом пузырька.

Вопросы для самоконтроля

1. Дать определение понятию «массив».
2. Описать механизм создания массивов в С.

Практическая работа № 7.

- 1) Написать программу, инициализирующую два массива размерностями 4x4 (значения вводит пользователь с клавиатуры).
- 2) Написать программу для вывода введенных элементов массива на экран.
- 3) Написать программу для нахождения суммы элементов введенных массивов (согласно правилам сложения матриц).
- 4) Написать программу для нахождения произведения элементов массивов (согласно правилу умножения двух матриц).
- 5) Изменить программу так, чтобы пользователь мог выбрать желаемое действие с массивами (сложение или умножение).

Вопросы для самоконтроля

1. Описать синтаксис вложенного цикла с параметром.
2. Оценить сложность программы для нахождения произведения элементов массивов.

Практическая работа № 8.

- 1) Написать программу для создания строковой константы.
- 2) Написать программу для сравнения строк, введенных с клавиатуры.
- 3) Написать программу для подсчета количества символов в введенной строке.

Вопросы для самоконтроля

1. Объяснить особенности работы со строками в С.
2. Описать синтаксис и принцип работы функций, использовавшиеся в ходе выполнения практической работы.

Практическая работа № 9.

- 1) Написать программу для вывода информации из файла на экран.

- 2) Написать программу для подсчета строк в файле.
- 3) Написать программу для подсчета количества символов в каждой строке файла.
- 4) Написать программу для сортировки строк файла по алфавиту и сохранения результата в другом файле.

Вопросы для самоконтроля

1. Объяснить особенности работы с файлами в С.
2. Описать синтаксис и принцип работы функций, использовавшиеся в ходе выполнения практической работы.

Практическая работа № 10.

- 1) Создать проект в среде разработки, изменив положение формы при его открытии.
- 2) Изучить возможности вкладок «Панель элементов» и «Свойства». В созданный проект добавить кнопки, и текстовые поля, а также поля для ввода текстовых сообщений и графической информации. Изменить стандартные названия добавленных объектов.
- 3) Написать программу для вывода приветственного сообщения при запуске проекта.

Вопросы для самоконтроля

1. Описать основные принципы ООП.
2. Описать свойства и методы созданных объектов.

Практическая работа № 11.

- 1) Изучить главное меню среды разработки. Во вкладке «Создание проекта» ознакомиться с доступными для работы языками и видами приложений, доступных для создания.
- 2) Изучить возможности вкладок «Обозреватель решений» и «Отладка».
- 3) Используя точки останова, запустить созданный ранее проект.
- 4) Использовать «Шаг с заходом» для отладки проекта.
- 5) Использовать «Шаг с обходом» для отладки проекта.

Вопросы для самоконтроля

1. Дать определение понятию «отладка»
2. Дать определение точке останова.
3. Описать и охарактеризовать разные способы отладки.

Практическая работа № 12.

Написать программу-тест, состоящую из десяти вопросов:

четыре вопроса с одним правильным ответом;

четыре вопроса с более, чем одним, правильным ответом; два

вопроса со свободным вариантом ответа.

Результат тестирования выводится в отдельном сообщении с указанием количества правильных ответов.

Вопросы для самоконтроля

1. Описать свойства radiobutton.
2. Описать свойства checkbox.

Практическая работа № 13.

Написать программу для выполнения сортировки вставками одномерного массива, заданного пользователем. Выполнить отладку написанного кода. Провести тестирование программы. Результаты тестирования вместе с тестовыми наборами включить в отчет.

Вопросы для самоконтроля

1. Описать этап кодирования программного продукта.
2. Описать этап тестирования программного продукта. Перечислить виды тестирования.

Практическая работа № 14.

Написать программу-калькулятор с графическим интерфейсом для работы с целыми числами. Реализовать функции сложения, вычитания, умножения, деления, основные тригонометрические функции, нахождения корня, возведения в степень, нахождения натурального логарифма.

Вопросы для самоконтроля

1. Описать реализацию основных тригонометрических функций в программе.
2. Перечислить исключительные ситуации, которые могут возникнуть в ходе работы приложения.

Практическая работа № 15.

Изменить программу с тестированием, добавив в нее еще один тест. Создать контекстное меню, в котором пользователь сможет выбрать тест, который желает пройти. В контекстном меню добавить также вкладку со справкой по программе.

Вопросы для самоконтроля

1. Описать свойства элементов button и textbox.
2. Перечислить и охарактеризовать возможные способы создания справки.

Практическая работа № 16.

Написать программу-анкету, состоящую из 10 вопросов, среди которых должны быть следующие: фамилия, имя, отчество; рождения; время начала заполнения анкеты.

Для завершения анкетирования пользователь должен нажать на кнопку с надписью «Выход», после чего в сообщении должно выводиться затраченное на анкетирование время в следующем виде: Имя Отчество Фамилия – часы:минуты:секунды.

Вопросы для самоконтроля

1. Описать свойства элемента monthcalendar.
2. Описать свойства элемента datetimerpicker.
3. Описать свойства элемента timer.

2. Методические указания к выполнению практической работы по дисциплине Основы программирования

3. Критерии и шкала оценивания

<i>Оценка</i>	<i>Критерии оценки</i>
<i>Отлично</i>	Правильность выполнения задания на практическую работу в соответствии с вариантом; высокая степень усвоения теоретического материала по теме практической работы. Способность продемонстрировать преподавателю навыки работы в инструментальной программной среде, а также применить их к решению типовых задач, отличных от варианта задания. Высокое качество подготовки отчета по практической работе. Правильность и полнота ответов на вопросы преподавателя при защите работы.
<i>Хорошо</i>	Демонстрирует достаточно высокий/выше среднего уровень выполнения задания на практическую работу в соответствии с вариантом и хорошую степень усвоения теоретического материала по теме лабораторной/практической работы. Все требования, предъявляемые к работе, выполнены.
<i>Удовлетворительно</i>	Демонстрирует средний уровень выполнения задания на практическую работу в соответствии с вариантом. Большинство требований, предъявляемых к заданию, выполнены.
<i>Неудовлетворительно</i>	Демонстрирует низкий/ниже среднего уровень знаний, умений, навыков в соответствии с критериями оценивания. Многие требования, предъявляемые к заданию, не выполнены.

1. Перечень методических материалов для проведения промежуточной аттестации по дисциплине «Основы программирования»

Оценочные средства	Методические материалы
Дифференцированный зачет	Вопросы для подготовки Перечень билетов Критерии оценки

2. Вопросы к дифференцированному зачету по дисциплине «Основы программирования». Теоретическая часть

1. Дать определение понятиям: информация, данные.
2. Дать общую характеристику языка программирования С.
3. Охарактеризовать процессы тестирования и отладки программ.
4. Перечислить и охарактеризовать свойства алгоритма.
5. Дать определение понятию язык программирования. Перечислить критерии классификации языков программирования и охарактеризовать их.
6. Перечислить и оценить типовые алгоритмы для работы с линейными массивами.
7. Перечислить и охарактеризовать основные объекты языка С.
8. Описать задачу сортировки массива. Оценить алгоритм сортировки линейного массива методом "пузырька"
9. Рассказать, что такое среда программирования, дать характеристику компонентам, ее составляющим.
10. Дать определение и краткую характеристику синтаксических и логических ошибок, привести примеры.
11. Оценить использование динамической памяти: выделение и освобождение памяти при работе с матрицами. Перечислить особенности обработки матриц при работе с динамической памятью.
12. Охарактеризовать модификаторы типов.
13. Оценить обработку массивов символов и строк в языке Си: ввод и вывод, простейшие алгоритмы сканирования и обработки строки.
14. Дать характеристику приоритетам операций в языке С.
15. Перечислить и оценить библиотечные функции для обработки строк. Привести примеры использования.
16. Охарактеризовать оператор и операцию присваивания в языке Си.
17. Рассказать про переменные и константы в языке С. Перечислить скалярные типы данных.
18. Охарактеризовать конструкцию ветвления: представление в блок схемах и языках программирования. Привести примеры полной и неполной формы.
19. Дать определение понятиям: компиляция, компоновка программы.
20. Дать характеристику локальным и глобальным переменным: область действия и область видимости.

21. Описать процесс объявления данных числовых типов в языке C и перечислить операции, допустимые для данных числовых типов.
22. Охарактеризовать типовые алгоритмы частичной обработки двумерного массива (по строкам или по столбцам).
23. Описать механизм использования библиотечных функций: заголовочные файлы, подключение библиотек.
24. Описать двумерный массив: объявление, инициализация двумерного массива, индексация элементов, формирование и вывод двумерного массива.
25. Охарактеризовать конструкцию выбора: представление в блок схемах и языках программирования. Привести пример на языке C.
26. Описать конструкцию цикла: представление в блок схемах и языках программирования.

Привести пример на языке C с использованием оператора for.

27. Охарактеризовать типовые алгоритмы для обработки двумерного массива (целиком).
28. Описать линейный массив: понятие массива, объявление, инициализация массива, индексация элементов, формирование и вывод массива.
29. Рассказать, что такое указатели, как они объявляются и инициализируются в языке C.

Перечислить операции, применимые к указателям.

30. Описать механизм использования указателей при работе с массивами в C.
31. Охарактеризовать функции в языке Си: понятие, объявление, прототипы функций. Вызов функции. Типы возвращаемых значений.
32. Описать конструкцию цикла: представление в блок схемах и языках программирования.

Привести пример на языке C с использованием оператора while-do.

33. Дать общую характеристику языка программирования C.
34. Описать конструкцию цикла: представление в блок схемах и языках программирования.

Привести пример на языке C с использованием оператора do-while.

35. Дать определение понятию «структура». Описать принцип работы со структурой.
36. Описать типы integer и его модификации (short и long).
37. Описать оператор условного перехода if-else. Перечислить и описать его компоненты.
38. Описать тип char и функции для работы с ним.
39. Перечислить основные алгоритмические конструкции. Описать оператор вложенного цикла. Привести пример.
40. Дать определение понятиям интерпретация и компоновка.
41. Дать определение понятию «модуль». Перечислить и охарактеризовать методы разработки структурированных программ.
42. Дать определение понятиям «функция» и «процедура» функции. Сравнить функцию и

процедуру. Привести пример объявления функции, описать структуру функции.

Практическая часть

1. Написать программу для вычисления суммы массивов 4x4, введенных с клавиатуры.
2. Написать программу для вычисления произведения массивов, введенных с клавиатуры, с реализацией проверки на возможность выполнения операции.
3. Написать программу для нахождения матрицы, транспонированной заданной.
4. Написать программу для сортировки одномерного массива методом пузырька.
5. Написать программу для сортировки вставками одномерного массива.
6. Написать программу-калькулятор для вычисления суммы, разности, произведения и частного двух чисел с реализацией меню.
7. Написать программу для вычисления факториала введенного числа.
8. Написать программу для подсчета количества строк в файле.
9. Написать программу для подсчета количества символов в каждой строке файла.
10. Написать программу для замены табуляции в файле пробелами.
11. Написать программу для сортировки строк в файле.
12. Написать программу для копирования содержимого файла.
13. Написать программу-калькулятор для основных тригонометрических функций.
14. Написать программу для вычисления чисел последовательности Фибоначчи до числа с индексом, заданным пользователем.
15. Программа просит пользователя ввести a и b, если $a > b > 5$, то программа выводит на экран слово "осень", если $a = b$, то выводит значение выражения $\frac{a+b}{a^2+b^2+1}$, во всех остальных

случаях ввести c и вывести значение выражения $\sqrt{c} * \sin(c^2 + a^2 b) - \frac{1}{7}$.

16. Напишите программу нахождения гипотенузы и площади прямоугольного треугольника по двум катетам.
17. Напишите программу вычисления среднего арифметического и среднего геометрического двух чисел.
18. Составить программу подсчета общего количества цифр и знаков "+", "-", ".", "*" в строке s, введенной с клавиатуры,
19. Составить программу печать элементов данного множества в алфавитном порядке
20. Составить программу формирования множества строчных латинских букв, входящих в строку, введенную с клавиатуры, и подсчета количества знаков препинания в ней.

3. Критерии и шкала оценивания ответа обучающегося на дифференцированном зачете по дисциплине «Основы программирования».

Оценка	Критерии оценки
--------	-----------------

<i>Отлично</i>	Обучающийся владеет знаниями и умениями дисциплины в полном объеме рабочей программы, достаточно глубоко осмысливает дисциплину; самостоятельно, в логической последовательности и исчерпывающе отвечает на все вопросы зачетного билета, умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать, и систематизировать изученный материал, выделять в нем главное; устанавливать причинно-следственные связи; четко формирует ответы, решает задачи повышенной сложности.
<i>Хорошо</i>	Обучающийся владеет знаниями и умениями дисциплины почти в полном объеме программы (имеются пробелы знаний только в некоторых, особенно сложных разделах); самостоятельно и отчасти при наводящих вопросах дает полноценные ответы на вопросы билета; не всегда выделяет наиболее существенное, не допускает вместе с тем серьезных ошибок в ответах; умеет решать средней сложности задачи.
<i>Удовлетворительно</i>	Обучающийся владеет обязательным объемом знаний по дисциплине; проявляет затруднения в самостоятельных ответах, оперирует неточными формулировками; в процессе ответов допускаются ошибки по существу вопросов. Обучающийся способен решать лишь наиболее легкие задачи, владеет только обязательным минимумом знаний.
<i>Неудовлетворительно</i>	Обучающийся не освоил обязательного минимума знаний по дисциплине, не способен ответить на вопросы билета даже при дополнительных наводящих вопросах.