

**Автономная некоммерческая профессиональная
образовательная организация
«КАЛИНИНГРАДСКИЙ КОЛЛЕДЖ УПРАВЛЕНИЯ»**

Утверждено
Учебно-методическим советом Колледжа
протокол заседания
№ 35 от 11 ноября 2021 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ
(ОП.04)**

По специальности	09.02.07 «Информационные системы и программирование»
Квалификация	«Специалист по информационным системам»
Форма обучения	Очная

Рабочий учебный план по
специальности утвержден
директором 01 октября 2021 г.

Калининград

Автономная некоммерческая профессиональная образовательная организация
«Калининградский колледж управления»

Лист актуализации ОП 04 Основы алгоритмизации и программирования
(наименование РПД с шифром)

Специальность: 09.02.07 Информационные системы и программирование

В целях актуализации основной образовательной программы внесены следующие изменения/ дополнения:

1. п. 3.2 – Информационное обеспечение обучения - обновлен список основной учебной литературы и дополнительной литературы.
2. п.3.4 – Лицензионное программное обеспечение - проведена актуализация лицензионного программного обеспечения.

Разработчик: Воробейкина И.В.
(ФИО, ученая степень, ученое звание)

22.05.2025
(дата)

Изменения (дополнения) в рабочую программу рассмотрены и утверждены на заседании учебно-методического совета, протокол №78 от 22 мая 2025 г.

СОГЛАСОВАНО:

Руководитель ОПОП

Шульгина Н.В.

Начальник

отдела оценки качества образования

Переляева А.М.

26.05. 2025 г.



Лист согласования рабочей программы дисциплины

Рабочая программа дисциплины «Основы алгоритмизации» разработана в соответствии с федеральным государственным образовательным стандартом среднего профессионального образования по специальности 09.02.07 «Информационные системы и программирование», утвержденным приказом Минобрнауки России от 09.12.2016 № 1547

Рабочая программа дисциплины рассмотрена и одобрена на заседании Учебно-методического совета колледжа, протокол № 35 от 11 ноября 2021 г.

Регистрационный номер _____

СОДЕРЖАНИЕ

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

1. ОБЩАЯ ХАРАКТЕРИСТИКА 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:

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. 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 ПРИМЕРНОЙ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем в часах
Объем образовательной программы	228
в том числе:	
теоретическое обучение	74
практические занятия	76
Самостоятельная работа	76
Промежуточная аттестация	2

1.2. 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.	Введение в программирование	15	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		
	Самостоятельная работа обучающихся	3	
Тема 1.2. Типы данных	Содержание учебного материала	4	
	1. Типы данных. Простые типы данных. Производные типы данных. Структурированные типы данных.		
	В том числе практических занятий и лабораторных работ		
	Самостоятельная работа обучающихся	2	
Раздел 2.	Содержание учебного материала	60	ОК 1
Тема 2.1.	1. Операции и выражения. Правила формирования и вычисления выражений.	40	

Операторы языка программирования	Структура программы. Ввод и вывод данных. Оператор присваивания. Составной оператор.		ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	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		
	В том числе практических занятий и лабораторных работ		
	Самостоятельная работа обучающихся		
Раздел 3.	Содержание учебного материала	42	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	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		
	Самостоятельная работа обучающихся	6	
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		
	Самостоятельная работа обучающихся	2	
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		
	Самостоятельная работа обучающихся	6	
Раздел 4	<i>Основные конструкции языков программирования</i>	18	
Тема 4.1 Указатели.	Содержание учебного материала	12	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	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		
	Самостоятельная работа обучающихся	6	
Раздел 5	Содержание учебного материала	90	
Тема 5.1 Основные принципы объектно-ориентированного программирования (ООП)	1. История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс.	12	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	2. Основные принципы ООП: инкапсуляция, наследование, полиморфизм.		
	3. Классы объектов. Компоненты и их свойства.		
	4. Событийно-управляемая модель программирования. Компонентно-ориентированный подход.		
	В том числе практических занятий и лабораторных работ		
	Самостоятельная работа обучающихся	6	
Тема 5.2 Интегрированная среда разработчика.	Содержание учебного материала	12	ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	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		
	Самостоятельная работа обучающихся	6	
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		
	Самостоятельная работа обучающихся	5	
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	6	
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		
	Самостоятельная работа обучающихся	5	
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		
	Самостоятельная работа обучающихся	2	
Примерная 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. Изучение интегрированной среды разработчика. Создание проекта с использованием компонентов для работы с текстом.			

Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени. События компонентов (элементов управления), их сущность и назначение. Создание процедур на основе событий. Создание проекта с использованием кнопочных компонентов. Создание проекта с использованием компонентов стандартных диалогов и системы меню. Разработка функциональной схемы работы приложения. Разработка оконного приложения с несколькими формами. Разработка игрового приложения. Создание процедур обработки событий. Компиляция и запуск приложения. Разработка интерфейса приложения. Тестирование, отладка приложения. Классы ООП: виды, назначение, свойства, методы, события. Объявления класса. Создание наследованного класса. Программирование приложений. Перегрузка методов.		
Промежуточная аттестация	2	
Всего:	228	

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

3.1. Для реализации программы учебной дисциплины используется лаборатория **«Программирования баз данных»**, оснащенная необходимым для реализации программы учебной дисциплины оборудованием.

3.2. Информационное обеспечение реализации программы

3.2.1. Основная и дополнительная учебная литература и электронные образовательные ресурсы, необходимые для освоения дисциплины

Основная учебная литература

1. Семакин И.Г. Основы алгоритмизации и программирования: практикум : учебное пособие для студентов СПО / И. Г. Семакин, А. П. Шестаков– 4-е изд., стер.– Москва: Академия, 2020. – 141 с. – (Профессиональное образование). – ISBN 978-5-4468-9272-3.

2. Нагаева, И. А. Основы алгоритмизации и программирования : практикум : учебное пособие : [12+] / И. А. Нагаева, И. А. Кузнецов. – Москва ; Берлин : Директ-Медиа, 2021. – 168 с. : схем. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=598404> (дата обращения: – Библиогр.: с. 162-163. – ISBN 978-5-4499-1612-9. – DOI 10.23681/598404. – Текст : электронный.\

Дополнительная учебная литература

1. Костюкова, Н.И. Комбинаторные алгоритмы для программистов / Н.И. Костюкова. - 2-е изд./, исправ/. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 217 с. : ил.; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429067>

2. Окулов, С.М. Основы программирования : учебное пособие : [12+] / С.М. Окулов. – 10-е изд., электрон. – Москва : Лаборатория знаний, 2020. – 339 с. – (Развитие интеллекта школьников). – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=221981>– ISBN 978-5-00101-759-2. – Текст : электронный.

3. Трофимов, В. В. Основы алгоритмизации и программирования: учебник для среднего профессионального образования / В. В. Трофимов, Т. А. Павловская; под редакцией В. В. Трофимова. — Москва: Издательство Юрайт, 2020. — 137 с. — (Профессиональное образование). — ISBN 978-5-534-07321-8. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/454452>

Электронные образовательные ресурсы

Электронно-библиотечная система «Университетская Библиотека Онлайн» - <https://biblioclub.ru/>.

Научная электронная библиотека - www.elibrary.ru.

Научная библиотека открытого доступа - <https://cyberleninka.ru>

Дополнительные ресурсы информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. <https://www.ixbt.com/> — специализированный российский информационно-аналитический сайт с самыми актуальными новостями из сферы IT

2. <https://3dnews.ru/> - Интернет издание - публикация новостей и аналитики в компьютерных технологиях, результатов тестирования компьютерной техники (видеокарт, мультимедиа, принтеров,

сканеров и др.).

3. <http://www.cnews.ru/> - издание о высоких технологиях. Информация о высоких технологиях.
4. <https://compress.ru/> - Компьютер ПРЕСС – Обзор новостей компьютерной аналитики.
5. <https://www.microsoft.com/ru-ru/learning/training.aspx> /Учебные курсы по ИТ Microsoft
6. <http://www.intuit.ru/> Интернет-университет информационных технологий («ИНТУИТ»)
7. <http://www.elw.ru/> Журнал «e-Learning World – Мир электронного обучения»
8. <https://www.it-world.ru> Новости и аналитика рынка информационных технологий
9. <https://www.osp.ru/> Все новости мира компьютеров и связи.
10. <http://biblioclub.ru/> - электронно-библиотечная система «Университетская библиотека ONLINE».
11. <http://lib.usue.ru> – Информационно библиотечный комплекс
12. <http://www.eLIBRARY.RU> - научная электронная библиотека
13. <http://www.knigafund.ru> -Электронная библиотека студента «КнигаФонд»
14. <http://i-exam.ru/> - Единый портал интернет-тестирования в сфере образования.

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

Результаты обучения	Критерии оценки	Формы и методы оценки
<i>Перечень умений, осваиваемых в рамках дисциплины:</i> • 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.	«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко. «Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с	Примеры форм и методов контроля и оценки • Компьютерное тестирование на знание терминологии по теме; • Тестирование.... • Контрольная работа • Самостоятельная работа. • Защита реферата.... • Семинар • Защита курсовой работы (проекта) • Выполнение проекта; • Наблюдение за выполнением практического задания. (деятельностью студента)
<i>Перечень знаний, осваиваемых в рамках дисциплины:</i> • 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.	ошибками. «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. «Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.	• Оценка выполнения практического задания(работы) • Подготовка и выступление с докладом, сообщением, презентацией... • Решение ситуационной задачи....
---	--	--

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ОП.04 Основы алгоритмизации и программирования

По специальности

**09.02.07 «Информационные
системы и программирование»**

Квалификация

**«Специалист по информационным
системам»**

Форма обучения

Очная

1.1. Оценочные средства по итогам освоения дисциплины

1.1.1. Цель оценочных средств

Целью оценочных средств является установление соответствия уровня подготовленности обучающегося на данном этапе обучения требованиям рабочей программы по дисциплине «Основы алгоритмизации и программирования».

Оценочные средства предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины «Основы алгоритмизации и программирования». Перечень видов оценочных средств соответствует рабочей программе дисциплины.

Комплект оценочных средств включает контрольные материалы для проведения всех видов контроля в форме устного и письменного опроса, практических занятий, и промежуточной аттестации в форме вопросов и заданий к зачету.

Структура и содержание заданий – задания разработаны в соответствии с рабочей программой дисциплины «Основы алгоритмизации и программирования».

1.1.2. Объекты оценивания – результаты освоения дисциплины

Объектом оценивания являются формируемые компетенции ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4.

Результатами освоения дисциплины являются:

- 3-1 общие принципы построения алгоритмов, основные алгоритмические конструкции;
 - 3-2 понятие системы программирования;
 - 3-3 основные элементы процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти;
 - 3-4 подпрограммы, составление библиотек программ;
- объектно-ориентированную модель программирования, понятие классов и объектов, их свойств и методов
- У-1 использовать языки программирования, строить логически правильные и эффективные программы.

Таблица 1 - Перечень компетенций, формируемых в процессе освоения дисциплины с указанием этапов их формирования

№ п/п	Этапы формирования компетенций в процессе освоения дисциплины. (контролируемые модули, разделы, темы дисциплины (результаты по разделам))	Перечень компетенций. (код контролируемой компетенции (или её части) / и ее формулировка)	Планируемые результаты освоения дисциплины	Формы контроля, наименование оценочного средства
1	Раздел 1. Основные средства ОАиП	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 4. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами	3-1 общие принципы построения алгоритмов, основные алгоритмические конструкции; 3-2 понятие системы программирования; 3-3 основные элементы процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти	Входной контроль (контрольная работа)

		ОК 5. 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 языках		
2	Раздел 2. Средства для оптимизации программирования	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 4. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами ОК 09. Использовать информационные технологии в профессиональной деятельности. ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языках ПК 1.2. Осуществлять выбор технологии, инструментальных средств и средств вычислительной техники при организации процесса разработки и исследования объектов профессиональной деятельности ПК 2.3. Обеспечивать сбор данных для анализа использования и функционирования программно-технических средств компьютерных сетей	3-4 подпрограммы, составление библиотек программ; объектно-ориентированную модель программирования, понятие классов и объектов, их свойств и методов	Текущий (контроль (тест))
3	Раздел 3. Работа с динамической памятью.	ПК 1.2. Осуществлять выбор технологии, инструментальных средств и средств вычислительной техники при организации	У-Использовать языки программирования, строить логически правильные и эффективные программы.	Рубежные контроль (задания)

		процесса разработки и исследования объектов профессиональной деятельности ПК 2.3. Обеспечивать сбор данных для анализа использования и функционирования программно-технических средств компьютерных сетей ПК 2.4. Взаимодействовать со специалистами и смежного профиля при разработке методов, средств и технологий применения объектов профессиональной деятельности		ПА (экзамен)
--	--	--	--	--------------

1.1.3. Формы контроля и оценки результатов освоения

Контроль и оценка результатов освоения – это выявление, измерение и оценивание знаний и умений формирующихся компетенций в рамках освоения дисциплины. В соответствии с учебным планом и рабочей программой дисциплины «Основы алгоритмизации и программирования» предусматривается входной, текущий, рубежный и итоговый контроль результатов освоения (промежуточная аттестация в форме зачета с оценкой).

1.2. Примерные (типовые) контрольные задания или иные материалы, необходимые для оценки знаний, умений, владений (или опыта деятельности), в процессе освоения дисциплины (модуля, практики), характеризующих этапы формирования компетенций в процессе освоения дисциплины

1.2.1. Примерные (типовые) контрольные задания или иные материалы для проведения входного контроля

Контрольная работа (на уровне знаний)

1. Что такое идентификатор? Каким условиям должен удовлетворять идентификатор, чтобы не было ошибки в программе?
Какова структура ячейки памяти?
2. Что такое конъюнкция? Создайте таблицу истинности для конъюнкции.
Сформулируйте 2 правила получения дополнительного кода числа.
3. Что такое дизъюнкция? Создайте таблицу истинности для дизъюнкции.
Сложить и проверить: 011010010+10101011
4. Что такое импликация? Создайте таблицу истинности для импликации.
Перевести из шестнадцатеричной системы счисления в десятичную: 1F8
5. Создайте таблицу истинности для отрицания.
Вычесть и проверить: 11011000 – 01100110
6. Какие высказывания являются эквивалентными? Создайте таблицу истинности для эквивалентности.
Пусть n – число линий в шине адреса. Каков объем адресного пространства?
7. Перечислите свойства алгоритма.
Перевести из десятичной системы счисления в шестнадцатеричную: 730
8. Назовите способы представления алгоритма.
Сложить и проверить: 10D+18
9. Как графически представить элемент алгоритма «проверка условия»?
Перевести из шестнадцатеричной системы счисления в десятичную: 7BC
10. Сколько байт содержится в слове, двойном слове, параграфе?
Сложить и проверить: 11110001+01010101

11. Назовите критерии качества программного изделия.
Вычесть и проверить: 11000110-00111100
12. Назовите атрибуты файлов на дисках.
Сложить и проверить: CCA+ADB.
13. Какие языки программирования высокого уровня вы знаете?
Вычесть и проверить: F83 – 3B.
14. Как называется язык программирования низкого уровня?
Перевести из десятичной системы счисления в шестнадцатеричную: 640.
15. Что такое трансляция программы?
Перевести из двоичной системы счисления в десятичную: 01010010
16. Что такое система программирования?
Перевести из шестнадцатеричной системы счисления в десятичную: 8AE
17. Как графически изображается ввод/вывод данных?
Сложить и проверить: C1A+A0B.
18. Назовите свойства ООП.
Вычесть и проверить: 7C - 6F.
19. Как графически изображается обработка данных?
Сложить и проверить: 01010010+01110011
20. Какую операцию в блок-схеме обозначает ромб?
Перевести из шестнадцатеричной системы счисления в десятичную: 3E6
21. Какую операцию в блок-схеме обозначает прямоугольник?
Перевести из двоичной системы счисления в десятичную: 01110011
22. Какую операцию в блок-схеме обозначает параллелограмм?
Перевести из десятичной системы счисления в шестнадцатеричную 477
23. Перечислите свойства алгоритма.
Какова структура ячейки памяти?
24. Что такое система программирования?
Сформулируйте 2 правила получения дополнительного кода числа.
25. Назовите критерии качества программного изделия.
Назовите свойства ООП.
26. Что такое конъюнкция? Создайте таблицу истинности для конъюнкции.
Сложить и проверить: C1A+A0B.
27. Что такое компиляция программы?
Что такое дизъюнкция? Создайте таблицу истинности для дизъюнкции
28. Как называется язык программирования низкого уровня?
Сложить и проверить: 011010010+10101011
29. Как графически представить элемент алгоритма «проверка условия»?
Пусть n – число линий в шине адреса. Каков объем адресного пространства?
30. Что такое идентификатор? Каким условиям должен удовлетворять идентификатор, чтобы не было ошибки в программе?
Какие высказывания являются эквивалентными? Создайте таблицу истинности для эквивалентности.

1.2.2. Примерные (типовые) контрольные задания или иные материалы для проведения текущего контроля

Тест (ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4) (на уровне знаний)

1. Что такое идентификатор? Каким условиям должен удовлетворять идентификатор, чтобы не было ошибки в программе? Что такое библиотечный модуль? Какова его структура?
2. Каков диапазон значений и объем памяти типа integer? word? byte? Должно ли совпадать имя библиотечного модуля с именем файла, который его содержит, или это необязательно? Как обратиться к библиотечному модулю в теле головной программы?
3. Каков объем памяти типов string и char? В чем различие между типами string и char? Какова максимально возможная длина строки? Какое соответствие должно соблюдаться между формальными и фактическими параметрами в подпрограммах?

4. В чем сходство и отличие между типами byte, word, integer? Расскажите об операциях, которые можно выполнять над строковыми величинами. Какая функция (процедура) является аналогом операции сцепления (+) при работе со строками?
5. Где в программе объявляются переменные? Как объявляется логический тип? Какие он может принимать значения? Что такое автоматический контроль ввода-вывода при работе с файлами? Как его отключить?
6. Покажите общую структуру программы на языке PASCAL. Может ли фактических параметров процедуры быть больше, чем формальных? А меньше?
7. Как работают операторы ввода READ, READLN? Чем они отличаются друг от друга? Что такое подпрограмма? Какие параметры подпрограммы называют формальными? фактическими?
8. Как работают операторы вывода WRITE, WRITELN? Чем они отличаются друг от друга? Что такое массив? Что такое размерность массива? Объявите в блоке VAR одномерный массив с именем MAS, состоящий из десяти элементов целого типа.
9. Когда возникает необходимость в условном операторе? Как называется этот оператор и каков формат его записи? Каков формат объявления данных (покажите на примере)? Какие типы данных вы знаете?
10. Расскажите об операторе выбора case. В чем его сходство с условным оператором? Какого типа должен быть параметр счетного цикла FOR? Покажите на примере.
11. Чем полный условный оператор отличается от неполного условного оператора? Какова структура счетного цикла FOR?
12. Какова структура циклов с пред- и постусловием (WHILE ... DO и REPEAT ... UNTIL)? Как выполняются эти циклы? Каков объем памяти типа real? Как еще можно объявить действительные числа?
13. Сколько раз исполнится фрагмент программы? For i := 1 to 10 Do k:=k+2; С каким шагом здесь изменяется переменная цикла? Укажите ее. Чем отличается инициализация статического и динамического массивов?
14. Может ли фактических параметров процедуры быть больше, чем формальных? А меньше? Как работает команда eof ()? Какой тип она возвращает?
15. Что такое библиотечный модуль? Какова его структура? Должно ли совпадать имя библиотечного модуля с именем файла, который его содержит, или это необязательно? Как обратиться к библиотечному модулю в теле головной программы?
16. Когда возникает необходимость в операторе выбора? Как называется этот оператор и каков формат его записи? Что такое массив? Что такое размерность массива? Объявите в блоке VAR двумерный массив с именем MAS, состоящий из 3-х строк и 4-х столбцов целого типа.
17. Что такое подпрограмма? Какие параметры подпрограммы называют формальными? фактическими? Каков диапазон значений и объем памяти типа integer? word? byte?
18. Как объявить файловую переменную? Зачем при программировании файлов необходимо отключать автоматический контроль ввода/вывода? Что такое блоки var, const?
19. Что обозначает {\$I-}? Как создать собственный тип данных?
20. Для чего используется оператор type? Где применяется команда Assign?
21. Чем команда Append отличается от Rewrite? Какая информация записывается в блоки Interface и Implementation в библиотечном модуле?
22. Как инициализировать файл на запись? Можно ли в программе подключить несколько библиотек?
23. Как инициализировать файл на чтение? Какое расширение у файла, содержащего исходный текст программы? Исходный текст библиотечного модуля?
24. Какой оператор проверяет существование файла? Что такое компиляция программы?
25. Что делает команда Reset? Какое расширение у файла, содержащего откомпилированный текст программы? Откомпилированный текст библиотечного модуля?
26. Какая команда закрывает файл? Почему файл обязательно необходимо закрыть? Какова структура программы на языке PASCAL?
27. Для чего необходима файловая переменная при работе с файлами? Как работают вложенные циклы? Приведите пример.

- 28 Покажите программно, как записать строку в файл? Что такое идентификатор? Каким условиям должен удовлетворять идентификатор, чтобы не было ошибки в программе?
- 29 Как прочитать строку из файла? Как прочитать все строки, находящиеся в файле? Должны ли совпадать тип и количество формальных и фактических параметров в подпрограмме?
- 30 Для чего нужна команда Ioresult? Приведите пример ее использования.
Где объявляются подпрограммы в Turbo Pascal? Существуют ли подпрограммы без параметров?

Критерии оценивания теста

% правильных ответов	Оценка по традиционной системе
85-100	отлично
70-84	хорошо
50-69	удовлетворительно
0-49	неудовлетворительно

1.2.3. Примерные (типовые) контрольные задания или иные материалы для проведения рубежного контроля (ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4) (на уровне умения)

№	Содержание задания	Порядок выполнения работы:
Тема: Массивы и оператор цикла		
1	Какой тип могут иметь индексы массивов?	Спроектировать и реализовать на Паскале программу со следующими программными функциями: 1.Программа запрашивает и вводит элементы массива с клавиатуры терминала, отыскивает минимальный элемент во введенном массиве, выводит его значение и значения его индексов; 2.Программа запрашивает и вводит элементы массива с клавиатуры терминала, отыскивает максимальный элемент во введенном массиве; 3.Программа запрашивает и вводит элементы массива с клавиатуры терминала, отыскивает максимальный элемент главной диагонали введенного массива, выводит его значение и значения его индексов; 4.Программа запрашивает и вводит элементы массива с клавиатуры терминала, отыскивает максимальный элемент побочной диагонали введенного массива, выводит его значение и значения его индексов.
2	Какого типа могут быть элементы массивов?	
3	Массивы какой размерности допустимы в Паскале?	
4	Как можно в программе ссылаться на элемент массива?	
5	Управляющие переменные какого типа допустимы в операторе for?	
6	Какого типа выражения допустимы в качестве начального и конечного значений?	
7	Как необходимо записать несколько операторов, чтобы они выполнялись в цикле for?	
Тема: Процедуры Паскаля		
1	Как можно передавать данные из вызывающей программы в процедуру?	1.Напишите процедуру вычисления квадратов двух дробных чисел и вывода на экран их суммы;
2	Как можно передать результаты выполнения процедуры в вызывающую её программу?	

3 3	Какие два типа формальных параметров определены для процедуры?	2.Напишите процедуру вычисления суммы квадратов двух дробных чисел и их разности; Программа должна работать в режиме диалога. В варианте 1 опишите процедуру с двумя параметрами-значениями. Результат печатайте в самой процедуре. В варианте 2 опишите процедуру с двумя параметрами-значениями и двумя параметрами-переменными. Результат печатайте в головной программе.
--------	--	---

1.2.4. Примерные (типовые) контрольные задания или иные материалы для проведения промежуточной аттестации

Примерные (типовые) вопросы к экзамену по дисциплине «Основы алгоритмизации и программирования»

1.на уровне знаний

Вопрос	Код компетенции (согласно РПД)
1. Алгоритм и его свойства. Графическое изображение алгоритмов линейной, разветвляющейся и циклической структур. Основные требования ГОСТа 19.003-80..	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
1. Система программирования.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
2. Характеристика языка Турбо Паскаль. Алфавит языка Паскаль.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
3. Структура программы на Паскале. Простые типы данных. Краткая характеристика.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
4. Операции и выражения. Оператор присваивания. Составной оператор.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
5. Операторы ввода-вывода WRITE, READ. Инициализация переменных.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
6. Строки. Операции над строками. Стандартные процедуры и функции для строк.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
7. Условный оператор <i>if</i> . Оператор выбора <i>case</i> .	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
8. Операторы цикла (счетный, с предусловием, с постусловием).	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
9. Множества. Операции над множествами.	ОК 1, ОК 2, ОК 4,

	ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
10. Метки и операторы перехода.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
11. Записи. Оператор присоединения With ... do.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
12. Структурированные типы данных. Краткая характеристика.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4.
13. Подпрограммы-процедуры. Обращение к процедуре. Параметры-значения. Параметры-переменные.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
14. Подпрограммы-функции. Обращение к функции. Формальные и фактические параметры.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
15. Одномерные массивы. Методы сортировки массивов.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
16. Одномерные массивы. Алгоритм поиска минимального и максимального элементов.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
17. Многомерные массивы. Методы работы с двумерными массивами.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
18. Процедуры работы с текстовыми файлами.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
19. Текстовые файлы. Алгоритм записи данных в файл. Ошибки открытия файла	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
20. Текстовые файлы. Алгоритм чтения данных из файла. Ошибки открытия файла.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
21. Создание библиотечного модуля.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4
22. Интерфейсная, исполняемая и иницилирующая части модуля. Компиляция модулей.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 1.2, ПК 2.3, ПК 2.4

Содержание задания	Код компетенции (согласно РПД)
Вычислить значение выражения $y = x^n / 4$; где x, n – задаются пользователем с клавиатуры. Результаты вывести на экран.	ПК 1.2, ПК 2.3, ПК 2.4
Определить сколько положительных, четных значений попадает в промежуток от -12 до +12. Результат вывести на экран.	ПК 1.2, ПК 2.3, ПК 2.4
Организовать диалог пользователя и компьютера по заполнению паспортных данных.	ПК 1.2, ПК 2.3, ПК 2.4
Составить программу вычислений следующих функций: Результаты вывести на экран.	ПК 1.2, ПК 2.3, ПК 2.4
Определить, какое из трех значений, вводимых пользователем с клавиатуры, является кратным 5. Вывести на экран соответствующее сообщение.	ПК 1.2, ПК 2.3, ПК 2.4
Определить количество отрицательных значений из 15 чисел, вводимых пользователем с клавиатуры. Результат вывести на экран.	ПК 1.2, ПК 2.3, ПК 2.4
Определить разность всех четных чисел, вводимых пользователем с клавиатуры. Ввод данных производить до тех пор, пока не встретится число 0. Результаты вывести на экран.	ПК 1.2, ПК 2.3, ПК 2.4
Вычислить где n изменяется от 10 до 0 с шагом 2. Значение x вводится пользователем с клавиатуры. Результаты вывести на экран.	ПК 1.2, ПК 2.3, ПК 2.4
Вычислить значение выражения $y = x^n / 3$; где x, n – задаются пользователем с клавиатуры. Результаты вывести на экран.	ПК 1.2, ПК 2.3, ПК 2.4
Определить сколько положительных значений, кратных 3 попадает в промежуток от -15 до +15. Результат вывести на экран.	ПК 1.2, ПК 2.3, ПК 2.4
Найти среднеарифметическое значение чисел от 10 до 25.Результат вывести на экран.	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить одномерный целочисленный массив, размером N , числами. Заменить все числа кратные 8 на 0. Вывести массив на экран до и после модификации.	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить двумерный целочисленный массив, размером N , числами. Зеркально отобразить его содержимое по горизонтали. Вывести массив на экран до и после модификации.	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить двумерный массив случайными числами. Зеркально отобразить его содержимое по вертикали. Вывести массив на экран до и после модификации.	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить двумерный массив случайными числами. Повернуть его содержимое на 90 градусов. Вывести массив на экран до и после модификации.	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить двумерный массив случайными числами. Повернуть его содержимое на 180 градусов. Вывести массив на экран до и после модификации	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить одномерный массив случайными числами в диапазоне от 0 до 50. Определить среднее арифметическое всех чисел входящих в массив и записать это значение в последний элемент массива. Вывести массив на экран до и после модификации	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить одномерный массив случайными числами. Посчитать количество элементов между максимальным и минимальным элементами массива. Вывести массив на экран до и после модификации.	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить одномерный массив случайными числами. Поменять местами максимальный и минимальный элементы массива. Вывести массив на экран до и после модификации.	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить одномерный массив случайными числами. Используя побитовые операции прибавить ко всем четным числам 1. Вывести массив на экран до и после модификации.	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить одномерный массив размерностью > 30 элементов случайными целыми числами из диапазона от 10 до 20. Создать второй массив и перенести в него элементы первого массива, но так, чтобы они встречались в нем только	ПК 1.2, ПК 2.3, ПК 2.4

один раз. Вывести оба массива на экран.	
Заполнить одномерный массив размерностью > 30 элементов случайными целыми числами из диапазона от 10 до 20. Посчитать какое количество элементов встречается в массиве больше 2 раз. Вывести массив и результат на экран.	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить одномерный массив вещественными числами по формуле , где x номер элемента массива. Вывести массив на экран. Заполнить одномерный массив вещественными числами по формуле , где A, B, C день и месяц сдачи экзамена и номер билета соответственно, а x номер элемента массива. Вывести массив на экран.	ПК 1.2, ПК 2.3, ПК 2.4
Дан двумерный целочисленный массив A размером 4×4 . Заполнить массив числами. Определить максимальный элемент массива. Вывести на экран полученный массив и значение максимального элемента.	ПК 1.2, ПК 2.3, ПК 2.4
Дан двумерный целочисленный массив A размером 4×4 . Заполнить массив числами. Определить минимальный элемент массива. Вывести на экран полученный массив и значение минимального элемента.	ПК 1.2, ПК 2.3, ПК 2.4
Написать программу, которая запрашивает десятичное число и вычисляет сумму цифр четных позиций.	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить одномерный массив случайными числами. Используя побитовые операции прибавить ко всем нечетным числам 1. Вывести массив на экран до и после модификации.	ПК 1.2, ПК 2.3, ПК 2.4
Написать программу, которая вычисляет сумму цифр натурального числа, введенного с клавиатуры.	ПК 1.2, ПК 2.3, ПК 2.4
Дана целочисленная прямоугольная матрица. Вычислить количество строк, содержащих хотя бы один нулевой элемент.	ПК 1.2, ПК 2.3, ПК 2.4
Написать программу, которая запрашивает две строки, а затем соединяет их и выводит новую строку на экран.	ПК 1.2, ПК 2.3, ПК 2.4
Написать программу, которая запрашивает десятичное число и вычисляет сумму цифр нечетных позиций.	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить два одномерных массива случайными числами. Вывести на экран оба массива и числа присутствующие в обоих массивах одновременно.	ПК 1.2, ПК 2.3, ПК 2.4
Дана целочисленная прямоугольная матрица. Вычислить количество положительных элементов по каждому столбцу.	ПК 1.2, ПК 2.3, ПК 2.4
Написать программу, которая считывает текст из файла и выводит на экран количество восклицательных предложений.	ПК 1.2, ПК 2.3, ПК 2.4
Составить функцию определения простого числа. Используя функцию определить количество простых чисел в массиве.	ПК 1.2, ПК 2.3, ПК 2.4
Даны две даты в формате ДД.ММ.ГГ. Написать функцию определяющую наибольшее из них. Формат даты задать в виде структуры.	ПК 1.2, ПК 2.3, ПК 2.4
Написать функцию, определяющую количество счастливых билетов в рулоне проездных билетов. Считать, что рулон начинается с билета № 423445 и содержит 200 билетов. Результат вывести на экран.	ПК 1.2, ПК 2.3, ПК 2.4
Составить банк данных о маршрутах поездов, указав: номер поезда, время отправления, время в пути, промежуточные пункты маршрута. Данные записать в файл.	ПК 1.2, ПК 2.3, ПК 2.4
Составить банк данных о маршрутах поездов, указав: номер поезда, время отправления, время в пути, промежуточные пункты маршрута. Данные записать в файл.	ПК 1.2, ПК 2.3, ПК 2.4
Записать в файл последовательного доступа N целых чисел, полученных с помощью генератора случайных чисел. Подсчитать количество пар противоположных чисел среди компонентов этого файла.	ПК 1.2, ПК 2.3, ПК 2.4
Заполнить файл последовательного доступа f целыми числами, полученными с помощью генератора случайных чисел. Получить в файле g те компоненты файла f , которые являются четными.	ПК 1.2, ПК 2.3, ПК 2.4
Построить описание класса содержащего информацию о почтовом адресе организации. Предусмотреть возможность раздельного изменения составных частей адреса, создания и уничтожения объектов этого класса.	ПК 1.2, ПК 2.3, ПК 2.4
Составить описание класса прямоугольников со сторонами, параллельными осям координат. Предусмотреть возможность перемещения прямоугольников на плоскости, изменения размеров.	ПК 1.2, ПК 2.3, ПК 2.4
Составить описание класса для объектов-векторов, задаваемых координатами	ПК 1.2, ПК 2.3, ПК 2.4

концов в трехмерном пространстве. Обеспечить операции сложения и вычитания векторов с получением нового вектора (суммы или разности), вычисления скалярного произведения двух векторов, длины вектора, косинуса угла между векторами.	
Составить описание класса для работы с цепными списками строк (строки произвольной длины) с операциями включения в список, удаления из списка элемента с заданным значением данного.	ПК 1.2, ПК 2.3, ПК 2.4
Построить систему классов для описания плоских геометрических фигур: круга, квадрата, прямоугольника. Предусмотреть методы для создания объектов, перемещения на плоскости.	ПК 1.2, ПК 2.3, ПК 2.4

Методические рекомендации и указания

1. Методические указания для обучающихся по освоению дисциплины

Дисциплина «Основы алгоритмизации и программирования» считается освоенной обучающимся, если он имеет положительные результаты входного, текущего контроля и промежуточной аттестации.

Для достижения вышеуказанного обучающийся должен соблюдать следующие правила, позволяющие освоить дисциплину на высоком уровне:

1. Начало освоения курса должно быть связано с изучением всех компонентов программы дисциплины «Основы алгоритмизации и программирования» с целью понимания ее содержания и указаний, которые будут доведены до сведения обучающегося на первой лекции и первом занятии семинарского типа.

Перед началом курса целесообразно ознакомиться со структурой дисциплины на основании программы, а так же с последовательностью изучения тем и их объемом. С целью оптимальной самоорганизации необходимо сопоставить эту информацию с графиком занятий и выявить наиболее затратные по времени и объему темы, чтобы заранее определить для себя периоды объемных заданий.

2. Каждая тема содержит лекционный материал, список литературы для самостоятельного изучения, вопросы и задания для подготовки к занятиям семинарского типа. Необходимо заранее обеспечить себя этими материалами и литературой или доступом к ним.

3. После лекции необходимо изучить лекционный материал по соответствующей теме, обратить особое внимание на актуальные и проблемные вопросы рассматриваемой темы.

4. Занятие семинарского типа, как правило, начинается с опроса по лекционному материалу темы и материалам указанных к теме литературных источников. В связи с этим подготовка к практическому занятию заключается в повторении лекционного материала и изучении вопросов предстоящего занятия.

При возникновении затруднений с пониманием материала занятия обучающийся должен обратиться с вопросом к преподавателю для получения соответствующих разъяснений в отведенное для этого преподавателем время на занятии либо по электронной почте. В интересах обучающегося своевременно довести до сведения преподавателя информацию о своих затруднениях в освоении предмета и получить необходимые разъяснения.

5. Подготовка к экзамену является заключительным этапом изучения дисциплины. Экзамен проводится в устной форме. Каждый билет содержит по два вопроса: один – теоретический, второй – практическое задание (или тесты).

Содержание вопросов находится в доступном режиме с начала изучения дисциплины. В связи с этим целесообразно изучать вопросы не в период экзаменационной сессии непосредственно в дни перед зачетом, а по каждой теме вместе с подготовкой к соответствующему текущему занятию. Кроме того необходимо помнить, что часть вопросов (не более 10%) непосредственно перед экзаменом может быть дополнена или изменена. В связи с этим целесообразно изучать не только вопросы, выносимые на экзамен, но и иные вопросы, рассматриваемые на лекциях и занятиях семинарского типа.

2. Методические указания по подготовке к сдаче зачета с оценкой

Зачет с оценкой является итоговой формой контроля знаний обучающегося по дисциплине

«Основы алгоритмизации и программирования», способом оценки результатов его учебной деятельности. Основной целью зачета с оценкой является проверка степени усвоения полученных обучающимся знаний и их системы.

Для успешной сдачи зачета необходимо продемонстрировать разумное сочетание знания и понимания учебного материала. На зачете проверяется не столько механическое запоминание обучающимся изложенной информации, сколько его способность её анализировать, объяснять, аргументировать и отстаивать свою позицию.

К зачету целесообразно готовиться с самого начала учебного цикла, поскольку только систематическая подготовка может обеспечить формирование у обучающегося качественных системных знаний.

При подготовке следует пользоваться комплексом различных источников - не только конспектами лекций, материалами по подготовке к семинарским занятиям, но также и учебной, научной, справочной литературой.

Преподаватель вправе задать на зачете обучающемуся наводящие, уточняющие и дополнительные вопросы в рамках билета.

Основными критериями, которыми преподаватель руководствуется при оценке знаний, являются следующие:

- соответствие ответа обучающегося теме вопросов;
- умение строить ответ полно, но лаконично с акцентом на наиболее важных моментах;
- степень осведомлённости о научных и нормативных источниках;
- умение связывать теорию с практикой;
- приведение конкретных примеров.