

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКАЯ ГОСУДАРСТВЕННАЯ АКАДЕМИЯ»**

СРЕДНЕПРОФЕССИОНАЛЬНЫЙ КОЛЛЕДЖ



УТВЕРЖДАЮ
Зам. директора по УР
М.А. Малеева
« 08 » 02 2024г.

**РАБОЧАЯ ПРОГРАММА
УЧЕБНОЙ ДИСЦИПЛИНЫ**

ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

по специальности 09.02.07 Информационные системы и программирование

Черкесск 2024г.

Рабочая программа учебной дисциплины разработана на основе Федерального государственного образовательного стандарта по специальности среднего профессионального образования 09.02.07 Информационные системы и программирование, направление подготовки – 09.00.00 Информатика и вычислительная техника

Организация-разработчик: СПК ФГБОУ ВО «СевКавГА»


Разработчики:

Зинченко Т.В., преподаватель СПК ФГБОУ ВО «СевКавГА»

Чагарова З.Н., преподаватель СПК ФГБОУ ВО «СевКавГА»

Одобрена на заседании цикловой комиссии «Информационные дисциплины»

от «06» 02 2024г. протокол № 6

Руководитель образовательной программы  Л.А. Черных

Рекомендована методическим советом колледжа

от «08» 02 2024г. протокол № 3

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 образовательной программы:

Учебная дисциплина ОП.04.Основы алгоритмизации и программирования является обязательной частью общепрофессионального цикла основной образовательной программы в соответствии с ФГОС по специальности 09.02.07 Информационные системы и программирование.

Учебная дисциплина ОП.04.Основы алгоритмизации и программирования обеспечивает формирование профессиональных и общих компетенций по всем видам деятельности ФГОС по специальности 09.02.07 Информационные системы и программирование. Особое значение дисциплина имеет при формировании и развитии:

ОК 1. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам

ОК 2. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности

ОК 4 Эффективно взаимодействовать и работать в коллективе и команде

ОК 5 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста

ОК 9 Пользоваться профессиональной документацией на государственном и иностранном языках.

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.

1.2. Цель и планируемые результаты освоения дисциплины:

В рамках программы учебной дисциплины обучающимися осваиваются умения и знания

Код ПК, ОК	Умения	Знания
ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ПК 2.4, ПК 2.5	<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 	<ul style="list-style-type: none"> - понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции; - эволюцию языков программирования, их классификацию, понятие системы программирования; - основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; - подпрограммы, составление библиотек подпрограмм; - объектно-ориентированную модель программирования, основные принципы

	<p>соответствии со стандартом кодирования;</p> <ul style="list-style-type: none">- выполнять проверку, отладку кода программы.	<p>объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения.</p>
--	--	---

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем в часах
Объем образовательной программы	259
Самостоятельная работа	17
Консультации	2
Суммарная учебная нагрузка во взаимодействии с преподавателем	234
в том числе:	
лекции, уроки	125
практические занятия	109
лабораторные занятия	-
Промежуточная аттестация (экзамен)	6

2.2. Тематический план и содержание учебной дисциплины ОП.04.Основы алгоритмизации и программирования

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем в часах	Коды компетенций, формированию которых способствует элемент программы
Раздел 1. Введение в программирование			ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ПК 2.4, 2.5
Тема 1.1. Языки программирования	Содержание учебного материала	8	
	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.		
	Практические работы	8	
1. Знакомство со средой программирования. 2. Составление программ линейной структуры. 3. Составление программ разветвляющейся структуры. 4. Составление программ циклической структуры			
	Самостоятельная работа обучающихся:	-	
Тема 1.2. Типы данных	Содержание учебного материала	4	
	1. Типы данных. Простые типы данных. Производные типы данных. Структурированные типы данных.		
	Практические работы	-	
	Самостоятельная работа обучающихся	-	
Раздел 2. Основные конструкции языков программирования			ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ПК 2.4, 2.5
Тема 2.1. Операторы языка программирования	Содержание учебного материала	55	
	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.		

	<p>Вложенные циклы.</p> <p>4. Массивы. Двумерные массивы. Строки. Стандартные процедуры и функции для работы со строками.</p> <p>5. Структурированный тип данных – множество. Операции над множествами.</p> <p>6. Комбинированный тип данных – запись. Файлы последовательного доступа. Файлы прямого доступа</p>		
	<p>Практические работы</p> <ol style="list-style-type: none"> 1. Правила формирования и вычисления выражений. 2. Реализация ввода-вывода данных. 3. Реализация алгоритмов разветвляющихся вычислительных процессов с использованием условного оператора if. 4. Реализация алгоритмов разветвляющихся вычислительных процессов с использованием условного оператора множественного выбора switch. 5. Программирование циклических вычислительных процессов с использованием оператора цикла for. 6. Программирование циклических вычислительных процессов с использованием оператора цикла while. 7. Программирование циклических вычислительных процессов с использованием оператора цикла do...while. 8. Обработка одномерных массивов. 9. Обработка двумерных массивов. 10. Работа со строками. 11. Работа с данными типа множество. 12. Файлы последовательного доступа. 13. Типизированные файлы. Нетипизированные файлы. 	55	
	<p>Самостоятельная работа обучающихся: подготовка к практическим работам с использованием методических рекомендаций, изучение теоретического материала в соответствии с дидактическими единицами темы и подготовка ответов на вопросы, выданные преподавателем (работа с конспектами, учебной и специальной литературой по параграфам, главам учебных пособий, указанным преподавателем).</p>	2	
Раздел 3. Технологии программирования			

Тема 3.1. Процедуры и функции	Содержание учебного материала	6	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ПК 2.4, 2.5
	1. Общие сведения о подпрограммах. Определение и вызов подпрограмм. Область видимости и время жизни переменной. Механизм передачи параметров. Организация функций.		
	2. Рекурсия. Программирование рекурсивных алгоритмов.		
	Практические работы	4	
1. Организация процедур. Организация функций. 2. Применение рекурсивных функций.			
Самостоятельная работа обучающихся: подготовка к практическим работам с использованием методических рекомендаций, изучение теоретического материала в соответствии с дидактическими единицами темы и подготовка ответов на вопросы, выданные преподавателем (работа с конспектами, учебной и специальной литературой по параграфам, главам учебных пособий, указанным преподавателем).	2		
Тема 3.2. Структуризация в программировании	Содержание учебного материала	2	
	1. Основы структурного программирования. Методы структурного программирования.		
	Практические работы	-	
	Самостоятельная работа обучающихся: подготовка к практическим работам с использованием методических рекомендаций, изучение теоретического материала в соответствии с дидактическими единицами темы и подготовка ответов на вопросы, выданные преподавателем (работа с конспектами, учебной и специальной литературой по параграфам, главам учебных пособий, указанным преподавателем).	2	
Тема 3.3 Указатели.	Содержание учебного материала	6	
	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.		
	Практические работы	2	
	1. Указатели и массивы арифметических данных.		
Самостоятельная работа обучающихся:	2		

	подготовка к практическим работам с использованием методических рекомендаций, изучение теоретического материала в соответствии с дидактическими единицами темы и подготовка ответов на вопросы, выданные преподавателем (работа с конспектами, учебной и специальной литературой по параграфам, главам учебных пособий, указанным преподавателем).		
Тема 3.4. Модульное программирование	Содержание учебного материала	4	
	1. Модульное программирование. Понятие модуля. Структура модуля. Компиляция и компоновка программы.		
	2. Стандартные модули.		
	Практические работы	2	
1. Программирование модуля. Создание библиотеки подпрограмм.			
	Самостоятельная работа обучающихся: подготовка к практическим работам с использованием методических рекомендаций, изучение теоретического материала в соответствии с дидактическими единицами темы и подготовка ответов на вопросы, выданные преподавателем (работа с конспектами, учебной и специальной литературой по параграфам, главам учебных пособий, указанным преподавателем).	2	
Раздел 4. Технология объектно-ориентированного программирования (ООП)			
Тема 4.1 Основные принципы объектно-ориентированного программирования (ООП)	Содержание учебного материала	8	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ПК 2.4, 2.5
	1. История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс.		
	2. Основные принципы ООП: инкапсуляция, наследование, полиморфизм.		
	3. Классы объектов. Компоненты и их свойства.		
	4. Событийно-управляемая модель программирования. Компонентно-ориентированный подход.		
	Практические работы	-	
	Самостоятельная работа обучающихся: подготовка к практическим работам с использованием методических рекомендаций, изучение теоретического материала в соответствии с дидактическими единицами темы и подготовка ответов на вопросы, выданные преподавателем (работа с конспектами, учебной и специальной литературой по параграфам, главам учебных пособий, указанным преподавателем).	3	
Тема 4.2	Содержание учебного материала	10	

Интегрированная среда разработчика.	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.		
	Практические работы 1. Изучение интегрированной среды разработчика.	2	
Самостоятельная работа обучающихся: подготовка к практическим работам с использованием методических рекомендаций, изучение теоретического материала в соответствии с дидактическими единицами темы и подготовка ответов на вопросы, выданные преподавателем (работа с конспектами, учебной и специальной литературой по параграфам, главам учебных пособий, указанным преподавателем).	2		
Тема 4.3. Визуальное событийно-управляемое программирование	Содержание учебного материала		
	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.	6	
	3. События компонентов (элементов управления), их сущность и назначение. Создание процедур на основе событий.		
Практические работы 1. Создание проекта с использованием компонентов для работы с текстом. 2. Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени. 3. Создание проекта с использованием кнопочных компонентов. 4. Создание проекта с использованием компонентов стандартных диалогов и системы меню. 5. События компонентов (элементов управления), их сущность и назначение. Создание процедур обработки событий.	20		

	Самостоятельная работа обучающихся: подготовка к практическим работам с использованием методических рекомендаций, изучение теоретического материала в соответствии с дидактическими единицами темы и подготовка ответов на вопросы, выданные преподавателем (работа с конспектами, учебной и специальной литературой по параграфам, главам учебных пособий, указанным преподавателем).	2	
Тема 4.4 Разработка оконного приложения	Содержание учебного материала	6	
	1. Разработка функционального интерфейса приложения. Создание интерфейса приложения.		
	2. Разработка функциональной схемы работы приложения.		
	3. Разработка игрового приложения.		
Практические работы	10		
1. Разработка функциональной схемы работы приложения.			
2. Разработка оконного приложения с несколькими формами.			
3. Разработка игрового приложения.			
Самостоятельная работа обучающихся	-		
Тема 4.5 Этапы разработки приложений	Содержание учебного материала	6	
	1. Разработка приложения.		
	2. Проектирование объектно-ориентированного приложения.		
	3. Создание интерфейса пользователя.		
	4. Тестирование, отладка приложения.		
	Практические работы	4	
1. Разработка интерфейса приложения.			
2. Тестирование, отладка приложения.			
Самостоятельная работа обучающихся	-		
Тема 4.6 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		
Практические работы	2		
1. Классы ООП: виды, назначение, свойства, методы, события. Объявление создание класса.			

	Самостоятельная работа обучающихся	-	
Консультации		2	
Промежуточная аттестация (экзамен)		6	
Всего:		259	

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

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

Лаборатория программирования и баз данных, оснащенная необходимым для реализации программы учебной дисциплины оборудованием:

Комплект учебной мебели: доска меловая – 1 шт., стол ученический – 18 шт., стул ученический – 26 шт., стол – 1 шт., стул – 1 шт.

Комплект учебно-методической документации

Технические средства обучения: компьютер в сборе (системный блок *IntelCore 17-9700K, плата SICAVYNELCA-1151, корпус Corsair 270R, блок питания ATX-2.3 120мм, жесткий диск SATA-3.1 tb, монитор LG-21.5 22 МК 400Н-В 1920/1080, клавиатура + мышь) – 1 шт.; компьютер в сборе (корпус AEROCOOL V-2XVX-500 (10 шт.), корпус Aerocool Aero 500 USB 3.0 (2 шт.), системный блок IntelCore 137100 3.9, плата MSILCA 1151 H110 H110M, блок питания – 350W ATX 2.3, память DIMM DDR4 8192 MB, жесткий диск SATA-3.1 tb, монитор LG-21.5 22 МК 400Н-В 1920/1080, клавиатура + мышь) – 12 шт.; принтер HP LaserJet 1320; проектор EPSON E6-X400 1024x768; настенный экран DEXPWM-80 203*203 см 113.

3.2. Информационное обеспечение реализации программы

Список основной литературы	
1	Кудинов, Ю. И. Основы алгоритмизации и программирования : учебное пособие для СПО / Ю. И. Кудинов, А. Ю. Келина. — 2-е изд. — Липецк, Саратов : Липецкий государственный технический университет, Профобразование, 2020. — 71 с. — ISBN 978-5-88247-956-4, 978-5-4488-0757-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: https://www.iprbookshop.ru/92834.html . — Режим доступа: для авторизир. пользователей. - DOI: https://doi.org/10.23682/92834
2	Чурина, Т. Г. Основы алгоритмизации и программирования : учебное пособие для СПО / Т. Г. Чурина, Т. В. Нестеренко. — Саратов, Москва : Профобразование, Ай Пи Ар Медиа, 2020. — 214 с. — ISBN 978-5-4488-0802-9, 978-5-4497-0465-8. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: https://www.iprbookshop.ru/96017.html . — Режим доступа: для авторизир. пользователей. - DOI: https://doi.org/10.23682/96017

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

Результаты обучения	Критерии оценки	Методы оценки
<p><i>Перечень осваиваемых компетенций в рамках дисциплины:</i> ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ПК 2.4, ПК 2.5</p>	<p>Оценка «отлично» - обучающийся показывает полные и глубокие знания программного материала, логично и аргументировано отвечает на поставленный вопрос, а также дополнительные вопросы, показывает высокий уровень теоретических знаний. Практическую часть выполняет на 100%.</p>	<p>Текущий контроль в форме: - тестовых опросов; - карточек для опроса; - задач для самостоятельного выполнения.</p>
<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>Оценка «хорошо» - обучающийся показывает глубокие знания программного материала, грамотно его излагает, достаточно полно отвечает на поставленный вопрос и дополнительные вопросы, умело формулирует выводы. В тоже время при ответе допускает несущественные погрешности. Практическую часть выполняет на 90%-80%.</p>	<p style="text-align: center;">Промежуточная аттестация: экзамен.</p> <p>Оценка: - результативности работы обучающегося при выполнении тестовых и письменных опросов по карточкам, задач для самостоятельного выполнения; - результата подготовки к экзамену.</p>
<p><i>Перечень знаний, осваиваемых в рамках дисциплины:</i></p> <ul style="list-style-type: none"> - Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции. - Эволюцию языков программирования, их классификацию, понятие системы программирования. - Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти. 	<p>«удовлетворительно» - обучающийся показывает достаточные, но не глубокие знания программного материала; при ответе не допускает грубых ошибок или противоречий, однако в формулировании ответа отсутствует должная связь между анализом, аргументацией и выводами. Для получения правильного ответа требуется уточняющие вопросы. Практическую часть выполняет на 70%-60%.</p>	

<ul style="list-style-type: none"> - 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>Оценка «неудовлетворительно» - обучающийся показывает недостаточные знания программного материала, не способен аргументировано и последовательно его излагать, допускаются грубые ошибки в ответах, неправильно отвечает на поставленный вопрос или затрудняется с ответом. Практическую часть выполняет на менее 50%.</p>	
---	---	--

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКАЯ ГОСУДАРСТВЕННАЯ АКАДЕМИЯ»
СРЕДНЕПРОФЕССИОНАЛЬНЫЙ КОЛЛЕДЖ**

Фонд оценочных средств

для проведения текущего контроля и промежуточной аттестации образовательной
программы

по учебной дисциплине «Основы алгоритмизации и программирования»

по специальности **09.02.07 Информационные системы и программирование**

форма проведения оценочной процедуры
экзамен

г. Черкесск, 2024 год

I. ОБЩЕЕ ПОЛОЖЕНИЕ

Фонд оценочных средств (ФОС) предназначен для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины *Основы алгоритмизации и программирования*.

ФОС включает контрольные материалы для проведения текущего контроля и промежуточной аттестации в форме экзамена.

ФОС разработан в соответствии с Федеральным государственным образовательным стандартом по специальности СПО 09.02.07 *Информационные системы и программирование* и рабочей программой учебной дисциплины *Основы алгоритмизации и программирования*.

II. РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ, ПОДЛЕЖАЩЕЙ ПРОВЕРКИ

Предмет(ы) оценивания	Объект(ы) оценивания	Показатели оценки
<p>Уметь:</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.	<ul style="list-style-type: none">- алгоритмы линейной, разветвляющей и циклической структуры;- построение блок-схем с помощью прикладного ПО;- работа в системе программирования;- разработка программ на языке C#;- оформление кода;- отладка программ;	<ul style="list-style-type: none">- тестовые опросы;- карточки для опроса;- задачи для самостоятельного выполнения.
<p>Знать:</p> <ul style="list-style-type: none">- Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.- Эволюцию языков	<ul style="list-style-type: none">- алгоритмизация, свойства алгоритмов: дискретность, определенность, результативность, массовость, понятность;- виды и способы представления алгоритмов;	

<p>программирования, их классификацию, понятие системы программирования.</p> <ul style="list-style-type: none"> - 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. 	<ul style="list-style-type: none"> - языки программирования; - системы программирования; - базовые типы данных языка С#; - идентификаторы, константы, переменные, выражения, функции языка С#; - структура программы на С#; - операторы и операции; - массивы, структуры, указатели, файлы; - классы памяти; - подпрограммы, библиотеки подпрограмм; - понятия объектно-ориентированного программирования: объекты и классы; - принципы объектно-ориентированного программирования: наследование, инкапсуляция, полиморфизм, переопределение.
<p>ПК</p> <p>ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.</p> <p>ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.</p>	<ul style="list-style-type: none"> - тестирование ПО; - инспектирование компонент ПО на соответствие стандартам кодирования;
<p>ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам</p>	<ul style="list-style-type: none"> - обоснованность постановки цели, выбора и применения методов и способов решения профессиональных задач;

<p>ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности</p> <p>ОК 04. Эффективно взаимодействовать и работать в коллективе и команде</p> <p>ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста</p> <p>ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках.</p>	<p>- адекватная оценка и самооценка эффективности и качества выполнения профессиональных задач</p> <p>- использование различных источников, включая электронные ресурсы, Интернет-ресурсы, периодические издания по специальности для решения профессиональных задач</p> <p>- взаимодействовать с обучающимися, преподавателями в ходе обучения;</p> <p>-демонстрировать грамотность устной и письменной речи, ясность формулирования и изложения мыслей</p> <p>- эффективность использования информационно-коммуникационных технологий в профессиональной деятельности согласно формируемым умениям и получаемому практическому опыту;</p> <p>- эффективность использования в профессиональной деятельности необходимой технической документации, в том числе на английском языке.</p>	
---	---	--

ТЕСТОВЫЕ ВОПРОСЫ ПО ДИСЦИПЛИНЕ «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

09.02.07 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ПРОГРАММИРОВАНИЕ

Компетенции: ПК 2.4, ПК 2.5.

№ №	Правильный ответ	Содержание вопроса	Компетенция
1.		<p>Укажите правильный порядок следования приоритетов бинарных операций: (Отметьте один правильный вариант ответа.)</p> <p>1) арифметические, логические, отношения 2) отношения, логические, арифметические 3) арифметические, отношения, логические 4) правильных ответов нет</p>	ПК 2.4
2.		Что используют все языки программирования высокого	ПК 2.4

	уровня для предоставления программисту простого и легкого доступа к различным объектам? (Отметьте один правильный вариант ответа.) 1) идентификаторы 2) коммутаторы 3) спецификаторы 4) правильных ответов нет		
3.	В результате выполнения фрагмента программы double x = 0, y = 0, z = x/y; (Отметьте один правильный вариант ответа.) 1) ошибки не будет и значение переменной z будет равно null 2) возникнет ошибка на этапе компиляции программы 3) возникнет ошибка на этапе выполнения программы 4) ошибки не будет и значение переменной z будет равно Infinity 5) ошибки не будет и значение переменной z будет равно NaN 6) ошибки не будет и значение переменной z будет равно 0 7) правильных ответов нет	ПК 2.4	
4.	Какие синтаксические правила объявления переменных являются корректными: 1) <тип> [<атрибуты>] [<модификаторы>] <список объявителей>; 2) <список объявителей> : <тип>; 3) [<атрибуты>] [<модификаторы>] <тип> <список объявителей>; 4) [<атрибуты>] [<модификаторы>] <тип> : <список объявителей>; 5) правильных ответов нет	ПК 2.4	
5.	Отметьте истинные высказывания: (Ответ считается верным, если отмечены все правильные варианты ответов) 1) память, отводимая переменной значимого типа, принадлежит переменной и ни с какой другой переменной не разделяется 2) память, отводимая переменной ссылочного типа, принадлежит переменной и ни с какой другой переменной не разделяется 3) значения переменных значимого типа нельзя изменять 4) для ссылочных типов значение задается ссылкой на некоторый объект, расположенный в динамической памяти - "куче" 5) несколько ссылочных переменных могут ссылаться на один и тот же объект и разделять его значения 6) правильных ответов нет	ПК 2.4	
6.	Перечислите конструктивные типы данных _____	ПК 2.4	
7.	Перевод программы с языка программирования на язык машинных кодов называется ...	ПК 2.4	
8.	Процесс исправления ошибок в программе, при котором цель исправить все ошибки не ставится, называется...	ПК 2.4	
9.	Процесс, посредством которого проверяется правильность	ПК 2.4	

		программы называется ...		
10.		Под перегрузкой операции понимается существование нескольких реализаций одной и той же ...	ПК 2.4	
11.		Что сделает программа выполнив следующий код: Console.WriteLine(«Hello, World!»); 1) Напишет Hello, World! 2) Напишет на новой строчке Hello, World! + 3) Удалит все значения с Hello, World!	ПК 2.4	
12.		Чему равен d, если int a = 0; int b = a++; int c = 0; int d = a + b + c + 3; 1) 4 + 2) False 3) 3	ПК 2.4	
13.		Чему будет равен c, если int a = 0; int c = - -a; 1) Null 2) 1 3) -1 +	ПК 2.4	
14.		Чему будет равен c, если int a = 0; int c = a- -; 1) -1 2) 1 3) 0 +	ПК 2.4	
15.		Чему будет равен c, если int a = 10; int b = 4; bool c = (a == 10 && b == 4); 1) True + 2) Null 3) False	ПК 2.4	
16.		Чему будет равен c, если int a = 10; int b = 4; int c = a % b; 1) 2 + 2) 11 3) 1	ПК 2.4	
17.		Где правильно создана переменная: 1) \$x = 10; 2) char symbol = 'A'; + 3) x = 0;	ПК 2.4	
18.		Какие предъявляются требования к идеальному критерию тестирования? 1) достаточность 2) достижимость 3) полнота 4) проверяемость	ПК 2.4	
19.		Чему будет равно значение переменной A после выполнения фрагмента программы, если B = 5? A = 2; if (B <= 4) C = 1; else { C = 0; A = C; } 1) 24 2) 31 3) 6 4) 0	ПК 2.4	
20.		Какое число получится в результате выполнения фрагмента программы? { int[] a = {3, 8, 0, -6, 0, -1, -9, 3}; int i, k; k =	ПК 2.4	

		<pre>Math.Abs(a[7]); for (i = 0; i < 8; i++) if (Math.Abs(a[i]) > k) k = a[i]; Console.WriteLine(k); }</pre> <p>1) 3 2) 23 3) 4</p>		
21.		_____ – процесс выполнения программы с целью найти ошибки. Может выполняться как с компьютером, так и без него (общий термин).	ПК 2.4	
22.		_____ – попытка найти в программе ошибки путем доказательств на основе математических теорем о правильности программы безотносительно к внешней программной среде (вид тестирования).	ПК 2.4	
23.		_____ – попытка найти ошибки, выполняя программу в тестовой или моделируемой среде (вид тестирования).	ПК 2.4	
24.		_____ – попытка найти ошибки, выполняя программу в заданной программной среде (вид тестирования).	ПК 2.4	
25.		_____ – авторитетное подтверждение правильности программы (итоговое тестирование, для критичного ПО).	ПК 2.4	
26.		_____ – средство установления точной природы ошибок, процесс, противоположный тестированию, ведет к устранению ошибок.	ПК 2.4	
27.		_____ – контроль отдельного модуля в изолированной среде (например, с помощью ведущей программы), инспекция текста модуля на сессии программистов, которая иногда дополняется математическим доказательством правильности модуля.	ПК 2.4	
28.		_____ – контроль сопряжений между частями системы, как между компонентами в комплексе, так и между модулями отдельного компонента (например, у заглушки).	ПК 2.4	
29.		_____ – контроль и/или испытание системы по отношению к исходным целям. Является процессом контроля , если оно выполняется в моделируемой среде, и процессом испытания в реальной среде.	ПК 2.4	
30.		Существует два подхода к проектированию тестов: тестирование _____ ящиком и тестирование _____ ящиком.	ПК 2.4	
31.		<p>Типы данных в языке C# принято классифицировать как: (Ответ считается верным, если отмечены все правильные варианты ответов)</p> <p>1) фигурные 2) определенные программистом 3) простые 4) встроенные (базисные) 5) сложные (структурные)</p>	ПК 2.5	

		б) правильных ответов нет		
32.		Отличительной особенностью каких языков программирования является их ориентация не на систему команд той или иной ЭВМ, а на систему операторов, характерных для записи определенного класса алгоритмов?	ПК 2.5	
33.		Что понимают под языком программирования (ЯП)?	ПК 2.5	
34.		Как называется именованная спецификация одного или более столбцов (для каждого столбца указывается имя, а также его тип или домен)? (Отметьте один правильный вариант ответа.) 1) строчный тип данных 2) объектный тип данных 3) комбинированный тип данных 4) правильных ответов нет	ПК 2.5	
35.		Что понимается под наследованием типов? (Отметьте один правильный вариант ответа.) 1) однородная масса разрядов, имеющая какую либо структуру 2) возможность дисциплинированного создания новых типов на основе уже определенных 3) многовходовой программный модуль, точки входа которого соответствуют набору операций реализуемого типа 4) правильных ответов нет	ПК 2.5	
36.		Что представляет собой открытый массив? (Отметьте один правильный вариант ответа.) 1) фактический параметр подпрограммы, описывающий базовый тип элементов массива, но не определяющий его размерности и границы 2) формальный параметр подпрограммы, описывающий базовый тип элементов массива и определяющий его размерность и границы 3) формальный параметр подпрограммы, описывающий базовый тип элементов массива, но не определяющий его размерности и границы 4) правильных ответов нет	ПК 2.5	
37.		Каким способом параметр массив всегда передается в функцию? (Отметьте один правильный вариант ответа.) 1) по значению 2) по направлению 3) по адресу 4) правильных ответов нет	ПК 2.5	
38.		Что понимается под классом в объектно-ориентированном программировании (ООП)?	ПК 2.5	
39.		Какие типы в языке C# относятся к значимым: (Ответ считается верным, если отмечены все правильные варианты ответов) 1) логический 2) арифметический 3) структуры 4) перечисление 5) массивы 6) строки 7) классы	ПК 2.5	

		8) правильных ответов нет		
40.		<p>Определение класса в C# и других объектных языках содержит: (Ответ считается верным, если отмечены все правильные варианты ответов)</p> <ol style="list-style-type: none"> 1) вершины 2) методы 3) приключения 4) равнины 5) поля 6) события 7) правильных ответов нет 	ПК 2.5	
41.		<p>Какие типы в языке C# относятся к ссылочным: (Ответ считается верным, если отмечены все правильные варианты ответов)</p> <ol style="list-style-type: none"> 1) логический 2) арифметический 3) структуры 4) перечисление 5) массивы 6) строки 7) классы 8) правильных ответов нет 	ПК 2.5	
42.		<p>Какой из перечисленных типов языка C# не относится к значимым:</p> <p>_____</p> <p>_____</p>	ПК 2.5	
43.		<p>Выражения строятся из операндов - ..., ..., ..., - объединенных знаками операций и скобками</p>	ПК 2.5	
44.		<p>Метод класса называется ... , если существует несколько реализаций этого метода.</p>	ПК 2.5	
45.		<p>Массивом называют упорядоченную совокупность элементов ... типа.</p>	ПК 2.5	
46.		<p>Возникнет ли ошибка при выполнении данного фрагмента кода?</p> <pre>object model = 156; model = "157"; Console.WriteLine(model);</pre> <ol style="list-style-type: none"> 1)ДА 2)НЕТ 	ПК 2.5	
47.		<p>Отметьте правильные объявления переменной:</p> <ol style="list-style-type: none"> 1) int x; 2) int x(77); 3) int x =77; 4) int x = new int(77); 5) int x = new int(); 	ПК 2.5	
48.		<p>Отметьте правильные объявления:</p> <ol style="list-style-type: none"> 1) char a = new char("a"); 2) char b = "b"; 3) char[] c = new char[5]; 4) char[] d = {'a','b','c','d'}; 	ПК 2.5	

49.	Отметьте правильные объявления константы 1) const double size = 55; 2) const int size = 55; 3) const byte size = 55; 4) const double size = 5.5; 5) const size = 55;	ПК 2.5
50.	В каком фрагменте возникнет ошибка: 1) sbyte sb1=1, sb2= 2, sb3 = sb1+ sb2 2) ulong ul1 =1, ul2= 2, ul3 = ul1 + ul2 3) int x; if (5 >4) x= 4 4) int x=5, y; if(x>4) y=x	ПК 2.5
51.	Нотация-соглашения об _____ переменных, констант и других идентификаторов в программном коде.	ПК 2.5
52.	_____ — это имя, присваиваемое типу (классу, интерфейсу, структуре, записи, делегату или перечислению), элементу, переменной или пространству имен.	ПК 2.5
53.	В коде должны отсутствовать _____ участки кода.	ПК 2.5
54.	Комментарии с тройной косой чертой — это комментарии _____-документации.	ПК 2.5
55.	Комментарии бывают однострочные и _____	ПК 2.5

КАРТОЧКИ ДЛЯ ОПРОСА (ПК 2.4., ПК 2.5., ОК 1, 2, 4, 5, 9)

Карточки для опроса по теме «Языки программирования» **ВАРИАНТ № 1**

1. Алгоритм – это _____

2. Свойство алгоритма, согласно которому алгоритм должен приводить к решению задачи за конкретное число простых шагов, называется:

1. Дискретность;
2. Результативность;
3. Понятность;
4. Массовость.

3. Блок-схема – это

4. Как выглядит блок «модификация» (цикл) и для чего он используется?

5. Перевести математическое выражение на язык C#.

$$f = t^2 + \sin(|t + x|)^2 =$$

$$t = e^x + \ln(x^3)^2 =$$

ВАРИАНТ № 2

1. Программа – это

2. Что представляет собой графический способ изображения алгоритмов?

3. Свойство алгоритма, согласно которому алгоритм представляет собой процесс решения задачи как последовательное выполнение простых шагов, называется:

1. Дискретность;
2. Результативность;
3. Понятность;
4. Массовость.

4. Как выглядит блок «решение» (условие) и для чего он используется?

5. Перевести математическое выражение на язык C#.

$$y = x^5 + \cos^2 3x - \sqrt{e^{-x} + 7 \ln 2x} =$$

$$y = e^{a+p} - \cos|a^2 - x| =$$

ВАРИАНТ № 3

1. Отладка программы – это

2. Свойство алгоритма, согласно которому алгоритм решения задачи разрабатывается в общем виде, называется:

- 1) Дискретность;
- 2) Результативность;
- 3) Понятность;
- 4) Массовость.

3. Что представляет собой графический способ изображения алгоритмов. Почему получил широкое распространение?

4. Как выглядит блок «процесс» и для чего он используется?

5. Перевести математическое выражение на язык C#.

$$c = \sin^2 x / \ln|x^2 + a^3| =$$

$$d = \lg 4x * \sqrt{x^2 + 9x} / e^x =$$

ВАРИАНТ № 4

1. На этапе программирования осуществляется... _____

2. Свойством алгоритмов не является:

Дискретность;

1. Неопределенность;
2. Определенность;
3. Массовость;

4. Результативность.

3. Псевдокоды – это _____

4. Как выглядит блок «модификация» (цикл) и для чего он используется?

5. Перевести математическое выражение на язык C#.

$$w = \cos(x^2 + 2,1) + |a + b| =$$

$$c = e^{\sqrt{(x^3 + 5,8)}} / a =$$

**Карточки для опроса по теме «Типы данных»
Вариант № 1**

1. Идентификатор - это

2. «Типом данных»

называют _____

3. Переменные в языке C#. Что понимается под «объявлением переменных»? _____

4. Укажите идентификатор записанный неверно.

а) abc ; б) summa ; в) sr_ar ; г) sr-ar д) NAME ; е) nomer1_arf.

5. *Задача:* Продумать и объявить переменные для пересчета расстояния из миль в километры, если известно, что расстояние равно 10 миль, а одна миля равна 1,609 км.

Вариант № 2

1. Алфавит языка C # - это

2. Базовые типы данных в языке

C#. _____

3. Константы в языке C#. Символьные и строковые константы.

4. Укажите идентификатор записанный верно.

а) for ; б) sin ; в) 45krt; г) a125; д) _tn+m ; е) ABCarf.

5. *Задача:* Продумать и объявить переменные для вычисления произведения и суммы двух переменных, значения которых равны 10,5 и 3.

Вариант № 3

1. Служебные слова в языке C#. Для чего необходимы? _____

2. Понятие «типа данных». Базовые типы данных. _____

3. Выражения в языке C#.

4. *Задача:* Продумать и объявить переменные для вычисления периметра треугольника, если известны его стороны 10 см, 8,6 см и 5см.

5. Укажите идентификатор записанный верно.
а) сумма; б) summa= ; в) 57a_f ; г) _125s; д) a-f; е) 2ABC.

**Карточки для опроса на тему «Процедуры и функции»
Вариант № 1**

1. Что представляют собой заголовочные файлы? _____

2. Что выполняет директива препроцессора using system ? _____

3. Что выполняет функция writeline? _____

4. В каком заголовочном файле находится функция readline Для чего она необходима? _____
5. Для вывода какого значения используется спецификатор преобразований % n.mf ?

6. Для чего необходима функция textcolor? Ее синтаксис?

7. Какая функция используется для ввода информации? _____
8. Управляющая строка – это _____

9. Для ввода какого значения используется спецификатор преобразований % n.c ?

10. Что выполняет функция getchar()? _____

Задача: Составить программу, для вычисления средней скорости 3-х велосипедистов (тип данных float). Ввод значений скоростей каждого велосипедиста осуществляется с клавиатуры. Результат вывести на экран.

Вариант № 2

1. В файле с каким расширением записывается исходная программа?

2. Для чего необходимы директивы препроцессора? Как они обозначается? _____

3. Что представляет собой заголовочный файл <using system>? _____

4. Что выполняет функция read? _____

5. Чем отличаются друг от друга функции write и writeline? _____

6. Для вывода какого значения используется спецификатор преобразований %ns? _____

7. Для чего необходима функция textbackground? В каком заголовочном файле она расположена? _____

8. Что обозначает знак & ? _____

9. Для ввода какого значения используется спецификатор преобразований %i? _____

10. В приведенной строке исправьте ошибку:

```
scanf ( % f , &x);
```

Задача: Написать программу, осуществляющую вычисление среднего арифметического двух переменных, если известно, что это целые числа. Ввод должен осуществляться с клавиатуры в одной строке. Результат вывести на экран.

Карточки для опроса на тему «Операторы языка программирования» ВАРИАНТ № 1

1. Условие – это _____

2. Перечислите основные условные операторы:

3. Цикл –

это... _____

4. Когда используется цикл for ? Какова его структура?

5. Задача: Составить блок-схему и написать программу, которая вычисляет сумму четных членов ряда 1, 2, 3, 4, ..., n . Количество элементов n должно задаваться во время работы программы.

ВАРИАНТ № 2

1. В каких алгоритмах используются условные операторы ?

2. Две формы записи оператора условного выбора if:

3. Циклические алгоритмы – это:

4. Какой цикл называется циклом с постусловием? Какова его структура?

5. Задача: Составить блок-схему и написать программу, которая вычисляет произведение первых n членов ряда 1, 3, 5, 7... Количество элементов n должно задаваться во время работы программы.

Карточки для опроса на тему «Объектно-ориентированное программирование»

1. Объектно-ориентированное программирование - это

2. Объект – это

3. Параметры объектов, которые можно выбирать при создании программы, получили название:

4. Способность программы выбирать в процессе выполнения различные методы получило название

5. Класс - это

6. Раздел, предназначенный для объявлений, доступных для внешнего использования, называется:

7. Методы, которые после выполнения возвращают определенные значения, называются:

8. Функция - элемент, которая вызывается в момент создания объекта класса и должна инициализировать данные указанными в вызове значениями, называется...

9. Инкапсуляция – это

10. Механизм создания нового класса из старого – это

ЗАДАЧИ ДЛЯ САМОСТОЯТЕЛЬНОГО ВЫПОЛНЕНИЯ (ПК 2.4., ПК 2.5., ОК 1, 2, 4, 5, 9)

Структурное программирование

1. Написать программу, которая сравнивает два введенных с клавиатуры числа. Программа должна указать, какое число больше, или, если числа равны, вывести соответствующее сообщение.
2. Написать программу, которая проверяет, является ли введенное пользователем целое число четным.
3. Написать программу вычисления стоимости разговора по телефону с учетом 20% скидки, предоставляемой по субботам и воскресеньям.
4. Написать программу, которая вычисляет оптимальный вес для пользователя, сравнивает его с реальным весом и выдает рекомендацию о необходимости поправиться или похудеть. Оптимальный вес вычисляется по формуле: Рост (см) - 100-10% от роста.
5. Написать программу, которая вычисляет дату следующего дня. Ниже приведен рекомендуемый вид экрана во время работы программы.
Введите цифрами сегодняшнюю дату (число, месяц, год) -> 31 12 2002
Последний день месяца!
С наступающим Новым годом!
Завтра 1.1.2003
6. Написать программу, которая после введенного с клавиатуры числа (в диапазоне от 1 до 999), обозначающего денежную единицу, дописывает слово "рубль" в правильной форме. *Например, 15 рублей, 51 рубль и т. д.*
7. Написать программу, которая после введенного с клавиатуры числа (в диапазоне от 1 до 99), обозначающего денежную единицу, дописывает слово "копейка" в правильной форме. *Например, 46 копеек, 51 копейка и т. д.*
8. Составьте блок-схему и программу, которая запрашивает номер месяца года, а затем выводит на экран какому времени года данный месяц соответствует: зима, весна, лето или осень.

9. Составьте блок-схему и программу, которая выводит на экран названия стран в зависимости от первой введенной буквы запроса.
10. Составьте блок-схему и программу, которая выводит на экран имена ваших друзей в зависимости от первой буквы запроса.
11. Составьте блок-схему и программу, которая запрашивает у пользователя номер телефона аварийных служб, а затем выводит название этой службы: «Пожарная охрана», «Милиция», «Скорая помощь», «Газовая служба».
12. Составьте блок-схему и программу, которая запрашивает номер месяца года, а затем выводит на экран какому времени года данный месяц соответствует: зима, весна, лето или осень.
13. Составьте блок-схему и программу, которая выводит на экран названия стран в зависимости от первой введенной буквы запроса.
14. Составьте блок-схему и программу, которая выводит на экран имена ваших друзей в зависимости от первой буквы запроса.
15. Составьте блок-схему и программу, которая запрашивает у пользователя номер телефона аварийных служб, а затем выводит название этой службы: «Пожарная охрана», «Милиция», «Скорая помощь», «Газовая служба».
16. Написать алгоритм и программу, которая вычисляет среднее арифметическое вводимой с клавиатуры последовательности дробных чисел. Количество чисел должно задаваться во время работы программы.
17. Написать алгоритм и программу, которая выводит на экран таблицу стоимости, например, яблок в диапазоне от 100 г до 1 кг с шагом 100 г.
18. Написать алгоритм и программу, которая выводит таблицу значений функции $y = |x|$. Диапазон изменения аргумента от -3 до 6, шаг приращения аргумента 0,5.
19. Написать алгоритм и программу, которая выводит таблицу значений функции $y = |x - 2| + |x + 1|$. Диапазон изменения аргумента от -5 до 5, шаг приращения аргумента 0,5.
20. Напишите алгоритм и программу, которая выводит на экран таблицу умножения, например, на 7.
21. Составить алгоритм и написать программу, которая определяет максимальное число из введенных с клавиатуры последовательности положительных чисел (длина последовательности неограниченна).
22. Составить алгоритм и написать программу, которая определяет минимальное число во введенной с клавиатуры последовательности положительных чисел (длина последовательности неограниченна).
23. Написать программу, которая вычисляет среднее арифметическое отрицательных элементов массива, состоящего из 15 произвольных целых чисел.
24. Написать программу, которая вычисляет сумму положительных элементов массива, состоящего из 10 произвольных целых чисел.
25. Составить программу для вычисления среднего арифметического элементов массива $(A_1, A_2, \dots, A_{20})$.
26. Составить программу для вычисления суммы элементов массива (A_1, \dots, A_{20}) , стоящих на четных местах. Параметр цикла изменяется от 2 с шагом 2.
27. Составить программу для вычисления среднего арифметического отрицательных элементов массива $(C_1, C_2, \dots, C_{20})$, полагая, что в массиве есть отрицательные значения. Вычислить сумму отрицательных значений и их количество.
28. Написать программу, которая вычисляет среднее арифметическое отрицательных элементов массива, состоящего из 15 произвольных целых чисел.
29. Написать программу, которая вычисляет сумму положительных элементов массива, состоящего из 10 произвольных целых чисел.

30. Составить программу для вычисления среднего арифметического элементов массива $(A_1, A_2, \dots, A_{20})$.
31. Составить программу для вычисления суммы элементов массива (A_1, \dots, A_{20}) , стоящих на четных местах. Параметр цикла изменяется от 2 с шагом 2.
32. Составить программу для вычисления среднего арифметического отрицательных элементов массива $(C_1, C_2, \dots, C_{20})$, полагая, что в массиве есть отрицательные значения. Вычислить сумму отрицательных значений и их количество.
33. Написать программу, которая объединяет два упорядоченных по возрастанию массива в один, также упорядоченный массив.
34. На основе сведений о ежедневном пробеге автомобилей автопредприятия рассчитать среднесуточный и общий пробег каждого автомобиля за неделю.
35. Написать программу, которая объединяет два упорядоченных по возрастанию массива в один, также упорядоченный массив.
36. На основе сведений о ежедневном пробеге автомобилей автопредприятия рассчитать среднесуточный и общий пробег каждого автомобиля за неделю.
37. Написать программу, которая вводит по строкам с клавиатуры двумерный массив и вычисляет среднее арифметическое его элементов.
38. Написать программу, которая определяет номер строки квадратной матрицы, сумма элементов которой максимальна.
39. В двумерном массиве произвольных чисел $A(4,7)$ определить номера столбцов, в которых среднее арифметическое его элементов меньше, чем среднее арифметическое элементов массива.
40. В двумерном массиве произвольных чисел $A(4,7)$ вычислить среднее арифметическое положительных элементов и количество элементов, равных нулю.

Объектно-ориентированное программирование

1. Произвести разработку программы для проверки знаний по информатике с помощью технологии объектно-ориентированного программирования. Выполнить отладку и тестирование программы, а также разработку программой документации.
2. Произвести разработку обучающей программы с помощью технологии объектно-ориентированного программирования. Выполнить отладку и тестирование программы, а также разработку программой документации.
3. Произвести разработку приложения «Калькулятор» с помощью технологии объектно-ориентированного программирования. Выполнить отладку и тестирование программы, а также разработку программой документации.
4. Произвести разработку электронного учебника по химии с помощью технологии объектно-ориентированного программирования. Выполнить отладку и тестирование программы, а также разработку программой документации.
5. Произвести разработку справочного пособия по математике с помощью технологии объектно-ориентированного программирования. Выполнить отладку и тестирование программы, а также разработку программой документации.
6. Произвести разработку многооконного приложения с помощью технологии объектно-ориентированного программирования. Выполнить отладку и тестирование программы, а также разработку программой документации.
7. Произвести разработку обучающей программы по физике с помощью технологии объектно-ориентированного программирования. Выполнить отладку и тестирование программы, а также разработку программой документации.
8. Произвести разработку приложения «Ежедневник» с помощью технологии объектно-ориентированного программирования. Выполнить отладку и тестирование программы, а также разработку программой документации.

9. Произвести разработку приложения «Ежедневник» с помощью технологии объектно-ориентированного программирования. Выполнить отладку и тестирование программы, а также разработку программной документации.
10. Произвести разработку программы «Графический редактор» с помощью технологии объектно-ориентированного программирования. Выполнить отладку и тестирование программы, а также разработку программной документации.

III. ОПИСАНИЕ ОРГАНИЗАЦИИ ОЦЕНИВАНИЯ И ПРАВИЛ ОПРЕДЕЛЕНИЯ РЕЗУЛЬТАТОВ ОЦЕНИВАНИЯ

Уровень подготовки обучающихся по учебной дисциплине оценивается в баллах: «5» («отлично»), «4» («хорошо»), «3» («удовлетворительно»), «2» («неудовлетворительно»).

Оценка *«отлично»* - обучающийся показывает полные и глубокие знания программного материала, логично и аргументировано отвечает на поставленный вопрос, а также дополнительные вопросы, показывает высокий уровень теоретических знаний. Практическую часть выполняет на 100%.

Оценка *«хорошо»* - обучающийся показывает глубокие знания программного материала, грамотно его излагает, достаточно полно отвечает на поставленный вопрос и дополнительные вопросы, умело формулирует выводы. В тоже время при ответе допускает несущественные погрешности. Практическую часть выполняет на 90%-80%.

Оценка *«удовлетворительно»* - обучающийся показывает достаточные, но не глубокие знания программного материала; при ответе не допускает грубых ошибок или противоречий, однако в формулировании ответа отсутствует должная связь между анализом, аргументацией и выводами. Для получения правильного ответа требуется уточняющие вопросы. Практическую часть выполняет на 70%-60%.

Оценка *«неудовлетворительно»* - обучающийся показывает недостаточные знания программного материала, не способен аргументировано и последовательно его излагать, допускаются грубые ошибки в ответах, неправильно отвечает на поставленный вопрос или затрудняется с ответом. Практическую часть выполняет на менее 50%.

Экзамен по учебной дисциплине проводится в период экзаменационных сессий, установленных календарным учебным графиком. Экзамен принимается преподавателями, которые проводили занятия по данной учебной дисциплине.

Во время экзамена по учебной дисциплине допускается использование наглядных пособий, материалов справочного характера, нормативных документов, образцов техники и других информационно-справочных материалов, перечень которых заранее регламентируется.

IV. КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ЭКЗАМЕНА ПО УЧЕБНОЙ ДИСЦИПЛИНЕ «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

Предмет(ы) оценивания	Объект(ы) оценивания	Критерии оценки
<p>Уметь:</p> <ul style="list-style-type: none"> - Разрабатывать алгоритмы для конкретных задач. 	<ul style="list-style-type: none"> - алгоритмы линейной, разветвляющейся и циклической 	<p>Оценка «отлично» - обучающийся показывает полные и глубокие знания</p>

<ul style="list-style-type: none"> - 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>структуры;</p> <ul style="list-style-type: none"> - построение блок-схем с помощью прикладного ПО; - работа в системе программирования; - разработка программ на языке С#; - оформление кода; - отладка программ; 	<p>материала, логично и аргументировано отвечает на поставленный вопрос, а также дополнительные вопросы, показывает высокий уровень теоретических знаний. Практическую часть выполняет на 100%.</p> <p>Оценка «хорошо» - обучающийся показывает глубокие знания материала, грамотно его излагает, достаточно полно отвечает на поставленный вопрос и дополнительные вопросы, умело формулирует выводы. В тоже время при ответе допускает несущественные погрешности. Практическую часть выполняет на 90%-80%.</p>
<p>Знать:</p> <ul style="list-style-type: none"> - Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции. - Эволюцию языков программирования, их классификацию, понятие системы программирования. - Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти. - Подпрограммы, составление библиотек подпрограмм. - Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и 	<ul style="list-style-type: none"> - алгоритмизация, свойства алгоритмов: дискретность, определенность, результативность, массовость, понятность; - виды и способы представления алгоритмов; - языки программирования; - системы программирования; - базовые типы данных языка С#; - идентификаторы, константы, переменные, выражения, функции языка С#; - структура программы на С#; - операторы и операции; - массивы, структуры, указатели, файлы; - классы памяти; - подпрограммы, библиотеки подпрограмм; - понятия объектно-ориентированного программирования: объекты и классы; - принципы объектно-ориентированного программирования: наследование, инкапсуляция, 	<p>Оценка «удовлетворительно» - обучающийся показывает достаточные, но не глубокие знания материала; при ответе не допускает грубых ошибок или противоречий, однако в формулировании ответа отсутствует должная связь между анализом, аргументацией и выводами. Для получения правильного ответа требуется уточняющие вопросы. Практическую часть выполняет на 70%-60%.</p> <p>Оценка «неудовлетворительно» - обучающийся показывает недостаточные знания материала, не способен аргументировано и последовательно его излагать, допускается грубые ошибки в ответах, неправильно отвечает на поставленный вопрос или затрудняется с ответом.</p>

<p>объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения.</p>	<p>полиморфизм, переопределение.</p>	<p>Практическую часть выполняет на менее 50%.</p>
<p>ПК</p> <p>ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.</p> <p>ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.</p>	<p>- тестирование ПО;</p> <p>- инспектирование компонент ПО на соответствие стандартам кодирования;</p>	
<p>ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам</p> <p>ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности</p> <p>ОК 04. Эффективно взаимодействовать и работать в коллективе и команде</p> <p>ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей</p>	<p>– обоснованность постановки цели, выбора и применения методов и способов решения профессиональных задач;</p> <p>- адекватная оценка и самооценка эффективности и качества выполнения профессиональных задач</p> <p>- использование различных источников, включая электронные ресурсы, Интернет-ресурсы, периодические издания по специальности для решения профессиональных задач</p> <p>- взаимодействовать с обучающимися, преподавателями в ходе обучения;</p> <p>-демонстрировать грамотность устной и письменной речи, ясность формулирования и изложения мыслей</p> <p>- эффективность использования информационно-коммуникационных технологий в профессиональной деятельности согласно формируемым умениям и получаемому практическому</p>	

<p>социального и культурного контекста</p> <p>ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках.</p>	<p>опыту;</p> <p>- эффективность использования в профессиональной деятельности необходимой технической документации, в том числе на английском языке.</p>	
---	---	--

Условия выполнения задания

1. Место выполнения задания: Лаборатория программирования и баз данных.
2. Максимальное время выполнения задания: 20 мин.

ВОПРОСЫ К ЭКЗАМЕНУ ПО ДИСЦИПЛИНЕ «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

1. Понятие алгоритма. Виды алгоритмов.
2. Способы представления алгоритмов.
3. Понятие алгоритма. Свойства алгоритмов.
4. Языки программирования.
5. Процедурные языки программирования.
6. Проблемно-ориентированные языки программирования.
7. Машинно-зависимые языки программирования.
8. Машинно-независимые языки программирования.
9. Системы программирования.
10. Алфавит языка C#.
11. Идентификаторы в языке C#.
12. Комментарии и пробелы в языке C#.
13. Базовые типы данных языка C#.
14. Целые и вещественные константы языка C#
15. Символьные константы и константы перечислимого типа в языке C#.
16. Переменные в языке C#.
17. Выражения в языке C#.
18. Структура программы на языке C#.
19. Этапы разработки C# программ.
20. Директивы препроцессора.
21. Функции в языке C#.
22. Объявление и описание функций в языке C#.
23. Библиотеки функций
24. Оператор цикла for.
25. Оператор цикла while.
26. Оператор цикла do-while.
27. Вложенные циклы в языке C#.
28. Строковые массивы.
29. Инициализация и вывод строковых массивов в языке C#.
30. Функции работы со строками в языке C#.
31. Спецификаторы преобразований для ввода-вывода значений переменных в языке C#.
32. Арифметические операции и операции присваивания языка C#.
33. Условный оператор выбора в языке C#.
34. Условный оператор множественного выбора в языке C#.

35. Массивы данных в языке С#.
36. Двумерные массивы.
37. Структуры в языке С#.
38. Структуры. Ввод-вывод элементов структур.
39. Указатели в языке С#.
40. Операции над указателями.
41. Файлы в языке С#.
42. Функции чтения и записи информации в файлах.
43. Классы памяти.
44. Структурное программирование.
45. Принципы структурного программирования.
46. Модульное программирование.
47. Принципы модульного программирования.
48. Понятие объектно-ориентированного программирования.
49. Принципы объектно-ориентированного программирования.
50. Принципы разработки программ с помощью методики объектно-ориентированного программирования.

ЭКЗАМЕНАЦИОННЫЕ БИЛЕТЫ ПО ДИСЦИПЛИНЕ «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

1. Способы представления алгоритмов.
2. Алфавит языка С#.
3. Составить блок-схему и программу решения задачи, которая выводит таблицу значений функции $y = -2.4x^2 + 5x - 3$ в диапазоне от -2 до 2, с шагом 0,5

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 2

1. Идентификаторы в языке С#.
2. Массивы данных в языке С#.
3. Составить блок-схему и программу решения задачи, которая вводит с клавиатуры 4 дробных чисел и вычисляет их среднее арифметическое.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 3

1. Комментарии и пробелы в языке С#.
2. Структуры в языке С#.
3. Составить блок-схему и программу решения задачи, которая вычисляет факториал введенного с клавиатуры числа.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 4

1. Базовые типы данных языка С#.
2. Условный оператор выбора в языке С#.
3. Написать программу, которая вычисляет среднее арифметическое ненулевых элементов введенного с клавиатуры массива, состоящего из 10 целых чисел.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 5

1. Целые и вещественные константы языка С#
2. Двумерные массивы.

3. Составить блок-схему и программу решения задачи: вычисляющую сумму членов ряда $2, 4, 6, \dots, n$, Количество членов ряда n должно вводиться с клавиатуры.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 6

1. Символьные константы и константы перечислимого типа в языке C#.
2. Указатели в языке C#.
3. Составить блок-схему и программу решения задачи: вычисляющую сумму положительных элементов массива, если известно, что массив содержит 15 элементов.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 7

1. Переменные в языке C#.
2. Файлы в языке C#.
3. Составить блок-схему и программу решения задачи, которая вычисляет среднее арифметическое элементов массива $A\{15\}$, элементы которого целочисленные.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 8

1. Выражения в языке C#.
2. Условный оператор множественного выбора в языке C#.
3. Составить блок-схему и программу решения задачи: Вычислить величину дохода по вкладу. Процентная ставка (% годовых) и время хранения (дней) задаются во время работы программы. Доход = сумма * ставка / 365 / 100 * срок вклада.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 9

1. Понятие алгоритма. Виды алгоритмов.
2. Строковые массивы.
3. Составить блок-схему и программу решения задачи, которая вводит с клавиатуры последовательность целых чисел и после ввода каждого числа выводит на экран искомое число, возведенное в квадрат.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 10

1. Структура C# программы.
2. Понятие алгоритма. Свойства алгоритмов.
3. Составить блок-схему и программу решения задачи, которая выводит на экран таблицу умножения, например, на 5.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 11

1. Этапы разработки C# программ. Директивы препроцессора.
2. Системы программирования.
3. Вычислить количество положительных элементов массива $A\{7\}$, если известно, что элементы массива целочисленные.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 12

1. Функции в языке C#.
2. Процедурные языки программирования.
3. Составить блок-схему и программу решения задачи: которая вычисляет сумму первых n целых положительных чисел. Количество суммируемых чисел должно вводиться во время работы программы

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 13

1. Вложенные циклы в языке C#.
2. Способы представления алгоритмов.
3. Составить блок-схему и программу решения задачи, которая выводит таблицу квадратов a^2 первых десяти целых положительных чисел

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 14

1. Инициализация и вывод строковых массивов в языке C#.
2. Машинно-зависимые языки программирования.
3. Вычислить сумму отрицательных элементов матрицы $A(3*5)$.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 15

1. Функции работы со строками в языке C#.
2. Оператор цикла for.
3. Вычислить сумму положительных элементов каждой строки матрицы $A(5*4)$.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 16

1. Спецификаторы преобразований для ввода-вывода значений переменных в языке C#.
2. Функции чтения и записи информации в файлах.
3. Составить блок-схему и программу решения задачи: решить квадратное уравнение $5x^2 + 3x + 10 = 0$.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 17

1. Структуры в языке C#.
2. Проблемно-ориентированные языки программирования.
3. Составить блок-схему и программу решения задачи, которая выводит таблицу значений функции $y = |x - 2| + |x + 1|$. Диапазон изменения аргумента от -5 до 5, шаг приращения аргумента 0,5.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 18

1. Арифметические операции и операции присваивания языка C#.
2. Библиотеки функций
3. Определить с помощью указателей сумму целочисленных значений массива, состоящего из 50 элементов.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 19

1. Указатели в языке C#.
2. Оператор цикла while.
3. Составить блок-схему и программу решения задачи, которая сравнивает два введенных с клавиатуры числа. Программа должна указать, какое число больше, или, если числа равны, вывести соответствующее сообщение.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 20

1. Условный оператор выбора в языке C#.
2. Объявление и описание функций в языке C#.
3. Составить программу, которая k раз выводит на экран заданную фразу «Изучайте Си», число k должно вводиться с клавиатуры.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 21

1. Организация цикла в языке C#.
2. Машинно-независимые языки программирования.
3. Составить блок-схему и программу решения задачи: вычислить сумму элементов матрицы $A(5,5)$, расположенных над главной диагональю. Индекс номера строки I изменяется от 1 до 4, так как в 5-й строке нет элемента, лежащего над главной диагональю, а индекс номера столбца J от $I+1$ до 5.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 22

1. Строковые массивы в языке C#.
2. Оператор цикла do-while.
3. Вычислить количество положительных элементов матрицы $A(4*7)$.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 23

1. Операторы и операции в языке C#.
2. Структуры. Ввод-вывод элементов структур.
3. Определить с помощью указателей количество отрицательных элементов массива, состоящего из 15 элементов.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 24

1. Структуры в языке C#.
2. Идентификаторы в языке C#.
3. Написать программу, которая вычисляет среднее арифметическое отрицательных элементов введенного с клавиатуры массива, состоящего из 25 целых чисел.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 25

1. Работа с файлами в языке C#.
2. Структура программы на языке C#.
3. Описать структуру, содержащую информацию о сотрудниках предприятия и имеющую следующие поля: фамилия и инициалы, год рождения, оклад. Написать программу, которая выводит информацию о сотрудниках, чей оклад превышает 10 тысяч рублей.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 26

1. Директивы препроцессора.
2. Строковые массивы в языке C#.
3. Вычислить произведение положительных элементов массива $A\{7\}$, если известно, что элементы массива целочисленные.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 27

1. Типы данных языка C#.
2. Условный оператор множественного выбора в языке C#.
3. Написать программу, которая вычисляет количество ненулевых элементов введенного с клавиатуры массива, состоящего из 15 целых чисел.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 28

1. Переменные в языке C#.
2. Указатели в языке C#.
3. Составить блок-схему и программу решения задачи, которая вычисляет среднее арифметическое положительных элементов массива $A\{7\}$, элементы которого целочисленные.

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 29

1. Функции работы со строками в языке C#.
2. Оператор цикла while.
3. Вычислить сумму положительных элементов матрицы A(3*4).

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 30

1. Идентификаторы в языке C#.
2. Файлы в языке C#.
3. Составить блок-схему и программу решения задачи, которая выводит таблицу значений функции $y = |x - 2| + |x + 1|$. Диапазон изменения аргумента от -3 до 2, шаг приращения аргумента 0,5.