

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Граков Федор Николаевич

Должность: Исполняющий обязанности директора Института агроинженерии

Дата подписания: 12.02.2025 10:02:51

Уникальный программный ключ:

b565aee4d22ae3b3a9be55d7e0bce1146dd07a5

**МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ»**

УТВЕРЖДАЮ

И.о. директора Института агроинженерии



Корнешук Н.Г.

«23» мая 2024 г.

РАБОЧАЯ ПРОГРАММА

УЧЕБНОЙ ДИСЦИПЛИНЫ

ОП.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

для специальности

09.02.07 «Информационные системы и программирование»

**квалификация - специалист по информационным ресурсам
среднего профессионального образования**

(программа подготовка специалистов среднего звена)

Троицк, 2024 г.

Рабочая программа учебной дисциплины разработана в соответствии с требованиями ФГОС СПО 09.02.07 Информационные системы и программирование, утвержденного приказом Министерства образования и науки РФ от 09.12.2016 г. № 1547 и учебным планом. Реализация воспитательного потенциала учебной дисциплины в процессе организации учебной деятельности обучающихся предусматривает использование воспитательных возможностей содержания дисциплины для формирования у обучающихся планируемых личностных результатов в соответствии с целью и задачами Рабочей программы воспитания.

При реализации программы учебной дисциплины используются различные образовательные технологии, в том числе дистанционные образовательные технологии и электронное обучение.

Настоящая рабочая программа дисциплины составлена в рамках основной профессиональной образовательной программы (ОПОП) и учитывает особенности обучения при инклюзивном образовании лиц с ограниченными возможностями здоровья (ОВЗ) и инвалидов.

Составитель:

- кандидат педагогических наук, доцент кафедры «Математические и естественнонаучные дисциплины» Витт А.М.

Рабочая программа дисциплины обсуждена на заседании кафедры «Математические и естественнонаучные дисциплины»

«14» мая 2024 г. (протокол № 9).

Зав. кафедрой «Математические и естественнонаучные дисциплины»,
доктор технических наук, профессор

-Обс-

Е.М. Басарьгина

Рабочая программа дисциплины одобрена методической комиссией Института агроинженерии

«21» мая 2024 г. (протокол № 5).

Председатель методической комиссии
Института агроинженерии ФГБОУ ВО
Южно-Уральский ГАУ, доктор педагогических наук, доцент

У

Н.Г. Корнешук

Директор Научной библиотеки



И.В. Шатрова

И.В. Шатрова

СОДЕРЖАНИЕ

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ.....	4
1.1 Область применения рабочей программы и место дисциплины в структуре основной образовательной программы	4
1.2. Цель и планируемые результаты освоения дисциплины.....	4
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ.....	6
2.1. Объем учебной дисциплины и виды учебной работы	6
2.2. Тематический план и содержание учебной дисциплины	7
3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ.....	12
3.1 Материально-техническое обеспечение.....	12
3.2. Информационное обеспечение обучения.....	13
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ.....	15
Приложение.....	17

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

ОП.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

1.1 Область применения рабочей программы и место дисциплины в структуре основной образовательной программы

Рабочая программа учебной дисциплины ОП.04 Основы алгоритмизации и программирования является обязательной частью основной образовательной программы по специальности среднего профессионального образования 09.02.07 Информационные системы и программирование в соответствии с ФГОС СПО 09.02.07 Информационные системы и программирование.

Рабочая программа учебной дисциплины ОП.04 Основы алгоритмизации и программирования разработана на основе Примерной основной образовательной программы СПО по специальности 09.02.07 «Информационные системы и программирование».

Согласно Учебному плану Колледжа по специальности 09.02.07 Информационные системы и программирование учебная дисциплина реализуется в рамках общепрофессионального цикла.

1.2. Цель и планируемые результаты освоения дисциплины

В рамках освоения программы учебной дисциплины обучающийся должен

знать:

- понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции;
- эволюцию языков программирования, их классификацию, понятие системы программирования;
- основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти;
- подпрограммы, составление библиотек подпрограмм;
- объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения;

уметь:

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на

конкретном языке программирования;

– оформлять код программы в соответствии со стандартом кодирования;

– выполнять проверку, отладку кода программы;

формировать компетенции:

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

ОК 2 . Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК4. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

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

ОК 9. Использовать информационные технологии в профессиональной деятельности.

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

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.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Максимальная учебная нагрузка	98
Обязательная аудиторная учебная нагрузка (работа обучающихся во взаимодействии с преподавателем)	126
В том числе:	
теоретические занятия	42
лабораторные и практические занятия	84
<i>в том числе:</i> <i>лабораторные и практические занятия в форме практической подготовки*</i>	-
курсовая работа/индивидуальный проект	-
Промежуточная аттестация** Экзамен (3,4 семестр)	22
Внеаудиторная самостоятельная работа обучающегося	9
Промежуточная аттестация (итоговая по дисциплине) в форме Экзамена в 4 семестре <i>*часовая нагрузка на лабораторные и/или практические занятия в форме практической подготовки выделяются из часов лабораторных и/или практических занятий в соответствии с учебным планом</i> <i>**на дифференцированный зачет и/или зачет выделяется не более 2 часов из часов обязательной аудиторной учебной нагрузки</i>	

2.2. Тематический план и содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающегося, курсовая работа (проект)	Объем в часах	Коды компетенций, формированию которых способствует элемент программы
Раздел 1.	<i>Введение в программирование</i>	9	
Тема 1.1. Языки программирования	Содержание учебного материала	2	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 ПК 5.3, ПК 5.4, ПК 8.3, ПК 9.2, ПК 9.3.
	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	3	
	1. Знакомство со средой программирования.		
	2. Компилирование кода		
Тема 1.2. Типы данных	Содержание учебного материала	2	
	1. Типы данных. Простые типы данных. Производные типы данных. Структурированные типы данных.		
	Практические работы	4	
1. Составление программ линейной структуры.			
	2. Алгоритмы для решения математических задач		
Раздел 2.	Содержание учебного материала	7	
Тема 2.1. Операторы языка программирования	1. Операции и выражения. Правила формирования и вычисления выражений. Структура программы. Ввод и вывод данных. Оператор присваивания. Составной оператор.	3	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 ПК 5.3, ПК 5.4, ПК 8.3, ПК 9.2, ПК 9.3.
	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		
	Практические работы 1. Составление программ разветвляющейся структуры. 2. Составление программ циклической структуры 3. Обработка одномерных массивов. 4. Обработка двумерных массивов. 5. Работа со строками. 6. Работа с данными типа множество. 7. Файлы последовательного доступа. 8. Типизированные файлы. 9. Нетипизированные файлы.	4	
Раздел 3.	Содержание учебного материала	12	
Тема 3.1. Процедуры и функции	1. Общие сведения о подпрограммах. Определение и вызов подпрограмм. Область видимости и время жизни переменной. Механизм передачи параметров. Организация функций.	2	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 ПК 5.3, ПК 5.4, ПК 8.3, ПК 9.2, ПК 9.3.
	2. Рекурсия. Программирование рекурсивных алгоритмов.		
	Практические работы 1. Организация процедур. 2. Организация функций. 3. Применение рекурсивных функций. 4. Решение задачи «Ханойские башни»	4	
Тема 3.2. Структуризация в программировании	Содержание учебного материала 1. Основы структурного программирования. Методы структурного программирования.	2	
Тема 3.3. Модульное программирование	Содержание учебного материала 1. Модульное программирование. Понятие модуля. Структура модуля. Компиляция и компоновка программы.	2	
	2. Стандартные модули.		
	Практические работы 1. Программирование модуля. 2. Создание библиотеки подпрограмм.	2	
Раздел 4	Основные конструкции языков программирования	5	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10
Тема 4.1 Указатели.	Содержание учебного материала	3	

	1. Указатели. Описание указателей. Основные понятия и применение динамически распределяемой памяти. Создание и удаление динамических переменных.		ПК 5.3, ПК 5.4, ПК 8.3, ПК 9.2, ПК 9.3.
	2. Структуры данных на основе указателей.		
	3. Задача о стеке.		
	Практические работы 1. Использование указателей для организации связанных списков.	2	
Раздел 5	Содержание учебного материала	39	
Тема 5.1 Основные принципы объектно-ориентированного программирования (ООП)	1. История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс.	2	
	2. Основные принципы ООП: инкапсуляция, наследование, полиморфизм.		
	3. Классы объектов. Компоненты и их свойства.		
	4. Событийно-управляемая модель программирования. Компонентно-ориентированный подход.		
	Практические работы 1. Классы ООП: виды, назначение, свойства, методы, события.	2	
Тема 5.2 Интегрированная среда разработчика.	Содержание учебного материала	3	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10 ПК 5.3, ПК 5.4, ПК 8.3, ПК 9.2, ПК 9.3.
	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.		
	Практические работы 1. Изучение интегрированной среды разработчика.	2	
Тема 5.3. Визуальное событийно-управляемое программирование	Содержание учебного материала	3	
	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.		
	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.	2	
Тема 5.4. Разработка оконного приложения	Содержание учебного материала	4	
	1. Разработка функционального интерфейса приложения. Создание интерфейса приложения.		
	2. Разработка функциональной схемы работы приложения.		
	3. Разработка игрового приложения.		
	Практические работы 1. Разработка функциональной схемы работы приложения. 2. Разработка оконного приложения с несколькими формами. 3. Создание процедур обработки событий. Компиляция и запуск приложения	4	
Тема 5.5. Этапы разработки приложений	Содержание учебного материала	4	
	1. Разработка приложения.		
	2. Проектирование объектно-ориентированного приложения.		
	3. Создание интерфейса пользователя.		
	4. Тестирование, отладка приложения.		
	Практические работы 1. Разработка игрового приложения. 2. Тестирование, отладка приложения. 3. Разработка интерфейса приложения.	3	
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		
	Практические работы 1. Объявления класса. 2. Создание наследованного класса. 3. Программирование приложений. 4. Перегрузка методов.	3	

Самостоятельная учебная работа	7	
1. Разработка программной документации		
2. Написание тест-кейсов для модуля программы		
3. Заполнение отчета об ошибках в баг-трекер		
Промежуточная аттестация Экзамен	22	
Всего:	98	

3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1 Материально-техническое обеспечение

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

Специальные помещения представляют собой учебные аудитории для проведения занятий всех видов, предусмотренных образовательной программой, в том числе групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы, мастерские и лаборатории, оснащенные оборудованием, техническими средствами обучения и материалами, учитывающими требования международных стандартов:

Посадочные места по числу студентов, рабочее место преподавателя, выход в Интернет, внутривузовская компьютерная сеть, доступ в электронную информационно-образовательную среду.

Ауд. 423

Мультимедиапроектор;

РМП - Компьютер DUAL Gb2010/GA-H61M/500Gb/2Gb;

15 РМУ - Компьютеры DUAL Gb2010/GA-H61M/500Gb/2Gb

Ауд. 426

Мультимедиапроектор – 1 штука, Стационарный компьютер –1 штука.

Ауд. 420

Мультимедиапроектор

РМП - Компьютер DUAL G2010/GA-H61M-500Gb/2Gb;

15 РМУ - Компьютеры DUAL G2010/GA-H61M?500Gb/2Gb

Ауд. 303 НОУТБУК HP 615 (VC289EA) RM76/2G/320/DVDR W/HD3200/DOS/15.6; ПЕРСОНАЛЬНЫЙ КОМПЬЮТЕР В КОМПЛЕКТЕ: системный блок Pentium E 5400 2.7GHZ, жесткий диск 250 Gb, монитор 19" LCD, клавиатура, мышь – 30 шт.; ПРИНТЕР CANON LBP-1120 лазерный; Экран с электроприводом; ПРИНТЕР CANON LBP-1120 лазерный; ИК ПУЛЬТ ДУ ДЛЯ ЭКРАНА С ЭЛЕКТРОПРИВОДОМ; КОЛОНКИ 5+1 SVEN ИНО.

3.2. Информационное обеспечение обучения (перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы):

Информационное обеспечение:

Операционная система Microsoft Windows XP;
Пакет программ Microsoft Office;
AutoCAD;
Компас 3D

Основные источники:

1 Тюкачев, Н. А. С#. Алгоритмы и структуры данных. + Электронное приложение : учебное пособие для спо / Н. А. Тюкачев, В. Г. Хлебостроев. — 2-е изд., стер. — Санкт-Петербург : Лань, 2023. — 232 с. — ISBN 978-5-507-45437-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/269837> .

2 Тюкачев, Н. А. С#. Основы программирования. + Электронное приложение : учебное пособие для спо / Н. А. Тюкачев, В. Г. Хлебостроев. — 3-е изд., стер. — Санкт-Петербург : Лань, 2023. — 272 с. — ISBN 978-5-507-45438-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/269840> .

3 Янцев, В. В. JavaScript и PHP. Content management system. + Электронное приложение : учебное пособие для спо / В. В. Янцев. — Санкт-Петербург : Лань, 2022. — 192 с. — ISBN 978-5-507-44845-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/266651> .

4 Янцев, В. В. JavaScript. Визуальные редакторы : учебное пособие для спо / В. В. Янцев. — 2-е изд., стер. — Санкт-Петербург : Лань, 2023. — 168 с. — ISBN 978-5-507-46080-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/297032> .

5 Андрианова, А. А. Алгоритмизация и программирование. Практикум : учебное пособие для спо / А. А. Андрианова, Л. Н. Исмагилов, Т. М. Мухтарова. — 2-е изд., стер. — Санкт-Петербург : Лань, 2022. — 240 с. — ISBN 978-5-8114-8948-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/186390> .

Дополнительные источники:

1 Батасова, В. С. Программирование. Сборник задач : учебное пособие для спо / . — 2-е изд., стер. — Санкт-Петербург : Лань, 2022. — 168 с. — ISBN 978-5-8114-9317-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/189452> .

2 Романенко, Т. А. Программные коллекции данных. Проектирование и реализация : учебное пособие для спо / . — Санкт-Петербург : Лань, 2021. — 152 с. — ISBN 978-5-8114-8207-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/183217> .

3. Беляев, С. А. Разработка игр на языке JavaScript : учебное пособие для спо / С. А. Беляев. — 3-е изд., стер. — Санкт-Петербург : Лань, 2023. — 152 с. — ISBN 978-5-507-46099-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/297650> .

4 Апанасевич, С. А. Структуры и алгоритмы обработки данных. Линейные структуры : учебное пособие для спо / С. А. Апанасевич. — Санкт-Петербург : Лань, 2021. — 136 с. — ISBN 978-5-8114-6918-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/153672>

5 Павлов, Л. А. Структуры и алгоритмы обработки данных : учебник для спо / Л. А. Павлов, Н. В. Первова. — 2-е изд., стер. — Санкт-Петербург : Лань, 2022. — 256 с. — ISBN 978-5-507-44105-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/207563> .

6 Ганичева, А. В. Теория принятия решений : учебное пособие для спо / А. В. Ганичева. — Санкт-Петербург : Лань, 2021. — 100 с. — ISBN 978-5-8114-7516-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/176876> .

7 Игнатъев, А. В. Тестирование программного обеспечения : учебное пособие для спо / А. В. Игнатъев. — 3-е изд., стер. — Санкт-Петербург : Лань, 2023. — 56 с. — ISBN 978-5-507-45426-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/269876> .

Электронные источники:

Интернет ресурсы:

1. Единое окно доступа к образовательным ресурсам - Электронные текстовые дан. - Режим доступа: <http://window.edu.ru/>

2. Единая коллекция цифровых образовательных ресурсов, включающая цифровые образовательные ресурсы, методические материалы, тематические коллекции, инструменты (программные средства), предназначенные для поддержки учебной деятельности и организации учебного процесса - Электронные текстовые дан. - Режим доступа: <http://school-collection.edu.ru/>

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Контроль и оценка результатов освоения дисциплины осуществляется преподавателем в процессе проведения практических занятий, тестирования, а также выполнения обучающимися индивидуальных заданий, рефератов, исследований.

Оценочные средства для текущего контроля и промежуточной аттестации представлены в Фонде оценочных средств.

Результаты обучения	Критерии оценки	Формы и методы оценки
<p><i>Перечень умений, осваиваемых в рамках дисциплины:</i></p> <ul style="list-style-type: none"> • Разрабатывать алгоритмы для конкретных задач. • Использовать программы для графического отображения алгоритмов. • Определять сложность работы алгоритмов. • Работать в среде программирования. • Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования. • Оформлять код программы в соответствии со стандартом кодирования. • Выполнять проверку, отладку кода программы. 	<p>«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.</p> <p>«Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.</p>	<p>Примеры форм и методов контроля и оценки</p> <ul style="list-style-type: none"> • Компьютерное тестирование на знание терминологии по теме; • Тестирование.... • Контрольная работа • Самостоятельная работа. • Защита реферата, доклада, презентации. • Наблюдение за выполнением практического задания. (деятельностью студента) • Оценка выполнения практического задания(работы) • Подготовка и выступление с докладом, сообщением, презентацией... • Решение ситуационной задачи <p>Экспертная оценка на промежуточной аттестации</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. • Подпрограммы, составление библиотек подпрограмм • Объектно-ориентированную модель программирования, основные принципы объектно- 	<p>«Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.</p> <p>«Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы,</p>	

ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения.	выполненные учебные задания содержат грубые ошибки.	
---	---	--

Фонд оценочных средств по учебной дисциплине

ОП.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

основной профессиональной образовательной программы
09.02.07 «Информационные системы и программирование»
квалификация - специалист по информационным ресурсам

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

1.1 Область применения

Фонд оценочных средств (ФОС) предназначен для контроля и оценки образовательных достижений обучающихся по учебной дисциплине ОП.04 Основы алгоритмизации и программирования

ФОС включает материалы для текущего контроля образовательных результатов и промежуточной аттестации обучающихся.

ФОС разработан в соответствии с требованиями ФГОС СПО 09.02.07 Информационные системы и программирование к образовательным результатам специалиста среднего звена «специалист по информационным ресурсам», Примерной основной образовательной программой и рабочей программой учебной дисциплины ОП.04 Основы алгоритмизации и программирования.

1.2 Требования к результатам обучения

Оценка качества подготовки обучающихся по специальности СПО 09.02.07 Информационные системы и программирование осуществляется в двух основных направлениях:

- контроль и оценка образовательных достижений обучающихся по учебным дисциплинам, МДК;
- оценка уровня сформированности компетенций обучающихся.

Контролируемые компетенции:

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

ОК 2 . Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК4. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

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

ОК 9. Использовать информационные технологии в профессиональной деятельности.

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

ПК 5.3. Разрабатывать подсистемы безопасности информационной системы в соответствии с техническим заданием

ПК 5.4. Производить разработку модулей информационной системы в соответствии с техническим заданием.

ПК 8.3. Осуществлять разработку дизайна веб-приложения с учетом современных тенденций в области веб-разработки.

ПК 9.2. Разрабатывать веб-приложение в соответствии с техническим заданием.

ПК 9.3. Разрабатывать интерфейс пользователя веб-приложений в соответствии с техническим заданием.

В результате изучения дисциплины обучающийся должен:

знать:

- понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции;
- эволюцию языков программирования, их классификацию, понятие системы программирования;
- основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти;
- подпрограммы, составление библиотек подпрограмм;
- объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения;

уметь:

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;
- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- оформлять код программы в соответствии со стандартом кодирования;
- выполнять проверку, отладку кода программы;

Форма промежуточной аттестации освоения учебной дисциплины –
Экзамен.

1.3 Система контроля и оценки результатов освоения обучающимися программы учебной дисциплины

Контроль и оценка результатов освоения дисциплины, курса, модуля включает оценочные средства для текущего контроля и промежуточной аттестации.

Формы оценочных средств, рекомендуемых к применению при проведении текущего контроля и промежуточной аттестации (по выбору)

№ п/п	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1.	Задания для самостоятельной работы	Средство проверки умений применять полученные знания по заранее определенной методике для решения задач или заданий по модулю или дисциплине в целом.	Комплект заданий
2.	Разноуровневые задачи и задания	Различают задачи и задания: 1. Ознакомительного, позволяющие оценивать и диагностировать знание фактического материала (базовые понятия, алгоритмы, факты) и умение правильно использовать специальные термины и понятия, узнавание объектов изучения в рамках определенного раздела дисциплины; 2. Репродуктивного уровня, позволяющие оценивать и диагностировать умения синтезировать, анализировать, обобщать фактический и теоретический материал с формулированием конкретных выводов, установлением причинно-следственных связей; 3. Продуктивного уровня, позволяющие оценивать и диагностировать умения, интегрировать знания различных областей, аргументировать	Комплект разноуровневых задач и заданий

		собственную точку зрения, выполнять проблемные задания.	
3.	Реферат	Продукт самостоятельной работы студента, представляющий собой краткое изложение в письменном виде полученных результатов теоретического анализа определенной научной (учебно-исследовательской) темы, где автор раскрывает суть исследуемой проблемы, приводит различные точки зрения, а также собственные взгляды на нее.	Темы рефератов
4.	Сообщение Доклад	Продукт самостоятельной работы студента, представляющий собой публичное выступление по представлению полученных результатов решения определенной учебно-практической, учебно-исследовательской или научной темы.	Темы докладов, сообщений
5.	Творческое задание	Частично регламентированное задание, имеющее нестандартное решение и позволяющее диагностировать умения, интегрировать знания различных областей, аргументировать собственную точку зрения. Может выполняться индивидуально или группой обучающихся.	Темы групповых и/или индивидуальных творческих заданий.
6.	Тест	Система стандартизированных заданий, позволяющая автоматизировать процедуру измерения уровня знаний и умений обучающегося.	Комплект тестовых заданий.
7.	Эссе	Средство, позволяющее оценить умение обучающегося письменно излагать суть поставленной проблемы, самостоятельно проводить анализ этой проблемы с использованием концепций и аналитического инструментария соответствующей дисциплины, делать выводы, обобщающие	Тематика эссе.

		авторскую позицию по поставленной проблеме.	
8.	Ролевая игра	Средство оценки способности обучающихся к выполнению реальных производственных задач, но в смоделированных условиях, приближенных к реальным	Сценарий, план игры
9.	Деловая игра, круглый стол	Средство оценки индивидуальных достижений обучающихся, позволяющее диагностировать уровень теоретических знаний и овладение практическими навыками деятельности в нестандартных ситуациях	Сценарий, план игры
10.	Кейс-задачи	Ситуация, представляемая в форме профессионально смоделированной задачи, в процессе решения которой у обучающего оценивается навык анализа профессиональных ситуаций, критического оценивания различных точек зрения, умение работать с информацией, способность моделировать решение профессиональной задачи	Комплект кейс-задач

Соотношение типов заданий и критериев оценки

Тип (вид) задания	Критерии оценки
Практическая работа	Критерии и нормы оценки практических работ.
Тесты	Шкала оценки образовательных достижений.
Устные ответы	Критерии и нормы оценки устных ответов.
Ситуационная задача	Критерии и нормы оценки ситуационной задачи.
Ролевая игра, деловая игра, круглый стол	Критерии и нормы оценки к деловой игре (ролевой игре, дискуссии, круглому столу, конференции и т.п.)
Проверка конспектов, рефератов, творческих работ, презентаций	Соответствие содержания работы заявленной теме; правилам оформления работы.

Критерии и нормы оценки практических работ

«5»	сформированность терминологического аппарата; владение системой знаний на уровне осознанного применения при выполнении учебных/ учебно-профессиональных действий; оригинальность решения, в том числе при решении нестандартных задач; гибкость, системность, глубину мышления; применение методов, адекватных поставленной цели и задачам; выполнение работы в логической последовательности; грамотное использование символики и графических средств; проявление высокого уровня самостоятельности; от 90 до 100% правильность выполнения практической работы
«4»	сформированность терминологического аппарата; владение программным материалом для выполнения учебных/ учебно-профессиональных действий, применение освоенных алгоритмов в типовой (знакомой) ситуации; применение методов, адекватных поставленной цели и задачам; выполнение работы в логической последовательности; грамотное использование символики и графических средств; выполнение практической работы самостоятельное; правильность выполнения – от 70 до 89%.
«3»	недостаточную сформированность терминологического аппарата; недостаточное владение программным материалом для выполнения учебных/ учебно-профессиональных действий; применение освоенных алгоритмов в типовой (знакомой) ситуации с незначительными

	нарушениями; применение нерациональных методов для выполнения практической работы; отступление от логической последовательности при выполнении работы; неточность использования символики и графических средств; проявление недостаточного уровня самостоятельности (выполнение работы с помощью преподавателя); правильность выполнения – от 51 % до 69%
«2»	недостаточную сформированность либо несформированность терминологического аппарата; недостаточное владение программным материалом для выполнения учебных/ учебно-профессиональных действий; применение освоенных алгоритмов в типовой (знакомой) ситуации со значительными нарушениями; применение нерациональных методов для выполнения практической работы; нарушение логической последовательности при выполнении работы; неточность использования символики и графических средств; проявление недостаточного уровня самостоятельности (выполнение работы с помощью преподавателя); правильность выполнения – менее 50 %
«1»	за полное незнание и непонимание учебного материала или отказ отвечать

Шкала оценки образовательных достижений (тестов)

Процент результативности (правильных ответов)	Оценка уровня подготовки	
	балл (отметка)	вербальный аналог
90 ÷ 100	5	отлично
89 ÷ 70	4	хорошо
69 ÷ 51	3	удовлетворительно
менее 50	2	неудовлетворительно

Критерии и нормы оценки устных ответов

«5»	за глубокое и полное овладение содержанием учебного материала, в котором обучающиеся легко ориентируются, за умение связывать теорию с практикой, высказывать и обосновывать свои суждения. Отличная отметка предполагает грамотное, логическое изложение ответа
«4»	если обучающийся полно освоил материал, владеет понятийным аппаратом, ориентируется в изученном материале, грамотно излагает ответ, но содержание, форма ответа имеют отдельные недостатки

«3»	если обучающийся обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности в определении понятий, не умеет доказательно обосновывать свои суждения
«2»	если обучающийся имеет разрозненные, бессистемные знания, не умеет выделять главное и второстепенное, допускает ошибки в определении понятий, искажающие их смысл, беспорядочно и неуверенно излагает материал
«1»	за полное незнание и непонимание учебного материала или отказ отвечать

Критерии и нормы оценки ситуационной задачи

«5»	Ответ на вопрос задачи дан правильный. Объяснение хода ее решения подробное, последовательное, грамотное, с теоретическими обоснованиями, с использованием демонстрационного материала (при необходимости), с правильным и свободным владением профессиональной терминологией; ответы на дополнительные вопросы верные, четкие
«4»	Ответ на вопрос задачи дан правильный. Объяснение хода ее решения подробное, но недостаточно логичное, с единичными ошибками в деталях, некоторыми затруднениями в теоретическом обосновании, а также при пояснении демонстрационного материала (при необходимости); ответы на дополнительные вопросы верные, но недостаточно четкие
«3»	Ответ на вопрос задачи дан правильный. Объяснение хода ее решения недостаточно полное, непоследовательное, с ошибками, слабым теоретическим обоснованием, со значительными затруднениями и ошибками в пояснении использованного (при необходимости) демонстрационного материала; ответы на дополнительные вопросы недостаточно четкие, с ошибками в деталях
«2»	Ответ на вопрос задачи дан неправильный. Объяснение хода ее решения представлено неполное, непоследовательное, с грубыми ошибками, без теоретического обоснования, без умения пояснять демонстрационный материал (при необходимости); ответы на дополнительные вопросы неправильные или отсутствуют

Критерии и нормы оценки подготовки к деловой игре (ролевой игре, дискуссии, круглому столу, конференции и т.п.)

«5»	обучающийся представил подготовленный материал, отвечающий заявленным требованиям; качество подготовленных материалов соответствует всем перечисленным критериям: – соответствие подготовленного материала заявленной теме;
-----	--

	<ul style="list-style-type: none"> – качество подготовленных материалов; – групповой характер работы; – упорядоченный и направляемый обмен мнениями с соответствующей организацией места и времени работы, но на основе самоорганизации участников; <p>направленность на достижение учебных целей</p>
«4»	<p>обучающийся представил подготовленный материал, отвечающий заявленным требованиям; качество подготовленных материалов не соответствует одному из критериев:</p> <ul style="list-style-type: none"> – соответствие подготовленного материала заявленной теме; – качество подготовленных материалов; – групповой характер работы; – упорядоченный и направляемый обмен мнениями с соответствующей организацией места и времени работы, но на основе самоорганизации участников; <p>направленность на достижение учебных целей</p>
«3»	<p>обучающийся представил подготовленный материал, отвечающий заявленным требованиям; качество подготовленных материалов не соответствует двум-трем из критериев:</p> <ul style="list-style-type: none"> – соответствие подготовленного материала заявленной теме; – качество подготовленных материалов; – групповой характер работы; – упорядоченный и направляемый обмен мнениями с соответствующей организацией места и времени работы, но на основе самоорганизации участников; <p>направленность на достижение учебных целей</p>
«2»	<p>обучающийся не подготовил материал</p>

Критерии и нормы оценки промежуточной аттестации

Оценки **«отлично»** заслуживает обучающийся, обнаруживший всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять практическое задание, усвоивший общие и профессиональные компетенции, соответствующие ФГОС, усвоивший взаимосвязь основных понятий тем и их значение для приобретаемой специальности, проявивший творческие способности. Обучающийся освещает различные вопросы программного материала, делает содержательные выводы, демонстрирует знание специальной литературы в рамках учебного методического комплекса и дополнительных источников информации, в том числе Интернет-ресурсов.

На оценку **«хорошо»** оценивается ответ, если обучающийся при ответе продемонстрировал системные знания и умения по поставленным вопросам. Содержание вопроса изложил связно, грамотным языком, раскрыл последовательно суть изученного материала, демонстрируя прочность

полученных знаний и умений, но при ответе были допущены незначительные ошибки, нарушалась последовательность изложения или отсутствовали некоторые несущественные элементы содержания тем.

Оценки «удовлетворительно» заслуживает обучающийся, обнаруживший знания основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по специальности/профессии, справляющийся с выполнением заданий, предусмотренных программой, знакомый с основной литературой, рекомендованной программой, но, у обучающегося обнаружены неточности в развернутом раскрытии понятий, терминов, определений, план ответа выстроен непоследовательно, в ответе допущены погрешности, исправленные под руководством преподавателя.

Оценка «неудовлетворительно» выставляется обучающемуся, если в ответе обнаружены пробелы в знаниях основного учебно-программного материала, недостаточно раскрыты понятия, термины, допущены принципиальные ошибки в выполнении практических заданий. Ответ содержит ряд серьезных неточностей. Выводы поверхностны. **Результаты оценки уровня освоения дисциплины (модуля) и компетенций обучающимися при текущем контроле успеваемости**

Код и наименование компетенции	Формы текущего контроля успеваемости*	Показатели	Уровень освоения	Результаты оценки
ОК 1. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам. ОК 2 . Осуществлять поиск, анализ и интерпретацию информации, необходимой для	Контрольная работа, тест, доклад	обучающийся овладел необходимыми компетенциями, приобрёл знания, умения; выполнил 100% заданий, подлежащего текущему контролю успеваемости самостоятельно и в требуемом объеме; обучающийся проявил умение обобщать, систематизировать и научно классифицировать материал, анализировать показатели с подробными пояснениями и аргументированными выводами	освоил	отлично

<p>выполнения задач профессиональной деятельности. ОК4. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами. ОК 5. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста. ОК 9. Использовать информационные технологии в профессиональной деятельности. ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке. ПК 5.3. Разрабатывать подсистемы безопасности информационной системы в соответствии с техническим заданием ПК 5.4. Производить разработку модулей информационной системы в соответствии с техническим</p>	<p>обучающийся приобрел знания, умения; овладел компетенциями (сформировал полностью или частично (не менее 70% компетенций)), закрепленные рабочей программой учебного предмета, курса, дисциплины (модуля), практической подготовкой, обучающийся выполнил 75% экзаменационных испытаний, или при выполнении допущены незначительные ошибки; обучающийся показал владение навыками систематизации материала; проявил умение обобщать, систематизировать и научно классифицировать материал; задания выполнил по стандартной методике без ошибок; сделал выводы по анализу показателей, но даны недостаточно полные пояснения</p>	освоил	хорошо
	<p>обучающийся овладел необходимыми компетенциями (сформировал частично (не менее 50% компетенций), приобрел знания, умения; не менее 50% задания, подлежащего текущему контролю успеваемости, выполнил по стандартной методике без существенных ошибок; сделал выводы по анализу показателей, но даны недостаточно полные пояснения</p>	частично освоил	удовлетворительно
	<p>обучающийся не приобрел знания, умения и не овладел компетенциями в объеме или выполнил менее чем на 50% с грубыми ошибками</p>	не освоил	неудовлетворительно

<p>заданием. ПК 8.3. Осуществлять разработку дизайна веб-приложения с учетом современных тенденций в области веб-разработки. ПК 9.2. Разрабатывать веб-приложение в соответствии с техническим заданием. ПК 9.3. Разрабатывать интерфейс пользователя веб-приложений в соответствии с техническим заданием.</p>				
---	--	--	--	--

Результаты оценки уровня освоения дисциплины (модуля) и компетенций обучающихся при промежуточной аттестации

Код и наименование компетенции	Формы промежуточной аттестации (зачет, дифференцированный зачет, экзамен, защита курсовой работы)	Показатели	Уровень освоения	Результаты оценки
<p>ОК 1. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.</p> <p>ОК 2 . Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.</p> <p>ОК4. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.</p>	<p>Экзамен</p>	<p>обучающийся овладел необходимыми компетенциями, приобрёл знания, умения; выполнил 100% заданий, подлежащего текущему контролю успеваемости самостоятельно и в требуемом объеме; обучающийся проявил умение обобщать, систематизировать и научно классифицировать материал, анализировать показатели с подробными пояснениями и аргументированными выводами</p>	<p>освоил</p>	<p>отлично</p>

<p>ОК 5. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.</p> <p>ОК 9. Использовать информационные технологии в профессиональной деятельности.</p> <p>ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке.</p> <p>ПК 5.3. Разрабатывать подсистемы безопасности информационной системы в соответствии с техническим заданием</p> <p>ПК 5.4. Производить разработку модулей информационной системы в соответствии с техническим</p>		<p>обучающийся приобрел знания, умения; овладел компетенциями (сформировал полностью или частично (не менее 70% компетенций)), закрепленные рабочей программой учебного предмета, курса, дисциплины (модуля), практической подготовкой , обучающийся выполнил 75% экзаменационных испытаний, или при выполнении допущены незначительные ошибки; обучающийся показал владение навыками систематизации материала; проявил умение обобщать, систематизировать и научно классифицировать материал; задания выполнил по стандартной методике без ошибок; сделал выводы по анализу показателей, но даны недостаточно полные пояснения</p>	<p>освоил</p>	<p>хорошо</p>
---	--	---	---------------	---------------

<p>заданием. ПК 8.3. Осуществлять разработку дизайна веб-приложения с учетом современных тенденций в области веб-разработки. ПК 9.2. Разрабатывать веб-приложение в соответствии с техническим заданием. ПК 9.3. Разрабатывать интерфейс пользователя веб-приложений в соответствии с техническим заданием.</p>		<p>обучающийся овладел необходимыми компетенциями (сформировал частично (не менее 50% компетенций), приобрел знания, умения; не менее 50% задания, подлежащего текущему контролю успеваемости, выполнил по стандартной методике без существенных ошибок; сделал выводы по анализу показателей, но даны недостаточно полные пояснения</p>	<p>частично освоил</p>	<p>удовлетворительно</p>
		<p>обучающийся не приобрел знания, умения и не овладел компетенциями в объеме или выполнил менее чем на 50% с грубыми ошибками</p>	<p>не освоил</p>	<p>неудовлетворительно</p>

2. КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА

2.1 Оценочные средства для текущего контроля образовательных результатов обучающихся

ОП.04 Основы алгоритмизации и программирования

Тестовое задание

Номер вопроса	Вопрос	Верный ответ
1	Понятие "алгоритм" произошло...	
	от имени ученого-математика	1
	от латинского слова, в переводе означающего "исполнитель"	
	от греческого слова, в переводе означающего "по шагам"	
	от латинского слова, в переводе означающего "программирование"	
2	Какое свойство алгоритма означает, что процесс решения задачи должен представлять последовательность простых шагов?	
	результативность	
	определенность	
	дискретность	1
	массовость	
3	Какое свойство алгоритма означает, что алгоритм должен приводить к решению за конечное число шагов?	
	результативность	1
	определенность	
	дискретность	
	массовость	
4	Какое свойство алгоритма означает, что каждое правило алгоритма должно быть четким и однозначным?	
	результативность	
	определенность	1
	дискретность	
	массовость	
5	Какое свойство алгоритма означает, что он должен разрабатываться в общем виде, для решения целого класса однотипных задач?	
	результативность	
	определенность	
	дискретность	
	массовость	1
6	Какого способа записи алгоритма не существует?	
	словесного	

	графического		
	естественного	1	
	программного		
7	Алгоритм, в котором та или иная команда выполняется в зависимости от истинности условия...		
	линейный		
	разветвляющийся	1	
	циклический		
8	Графическое представление алгоритма		
	таблица		
	блок-схема	1	
	чертеж		
	иерархия		
9	Какой тип блоков не используется в линейном алгоритме?		
	блок условия	1	
	блок действия		
	блок ввода		
	блок вывода		
10	Выберите условие принадлежности точки x отрезку $[a;b]$		
	$(x \leq b) \text{ OR } (x \geq a)$		
	$(x \leq b) \text{ AND } (x \geq a)$	1	
	$(x \leq a) \text{ OR } (x \geq b)$		
	$(x \leq a) \text{ AND } (x \geq b)$		
11	Выберите условие x лежит вне отрезка $[a;b]$		
	$(x < b) \text{ OR } (x > a)$		
	$(x < b) \text{ AND } (x > a)$		
	$(x < a) \text{ OR } (x > b)$	1	
	$(x \leq a) \text{ AND } (x \geq b)$		
12	Какой логической операции в алгебре множеств соответствует операция объединения множеств?		
	дизъюнкции	1	
	конъюнкции		
	импликации		
	инверсии		
13	Какой логической операции в алгебре множеств соответствует операция пересечения множеств?		
	дизъюнкции		
	конъюнкции	1	
	импликации		
	инверсии		

14	Заголовок цикла определяет...		
	количество повторений цикла	1	
	набор команд для повторения		
	условие печати результатов		
	параметры цикла		
15	Составное высказывание ложно тогда и только тогда, когда оба элементарных высказывания ложны. О какой логической операции идет речь?		
	конъюнкция		
	дизъюнкция	1	
	импликация		
	эквиваленция		
16	Составное высказывание истинно тогда и только тогда, когда оба элементарных высказывания истинны. О какой логической операции идет речь?		
	конъюнкция	1	
	дизъюнкция		
	импликация		
	эквиваленция		
17	Что такое тело цикла?		
	параметр цикла		
	условие выхода из цикла		
	первая строка циклической структуры		
	набор команд для повторения	1	
18	Какое свойство алгоритма не выполняется, когда входными данными являются константы?		
	результативность		
	определенность		
	дискретность		
	массовость	1	
19	Какого вида ошибок при отладке программы не существует?		
	грамматические	1	
	синтаксические		
	логические		
20	Алгоритмическая конструкция выбор используется для...		
	определения условия выхода из цикла		
	перечня констант и переменных		
	организации множественного ветвления	1	
	определения тела цикла		
21	В каком виде циклической структуры тело цикла выполнится хотя бы один раз?		

	цикл с параметром		
	цикл с предусловием		
	цикл с постусловием	1	
22	Для какого из указанных значений числа X истинно выражение NOT(X<2) AND (X<3)?		
	Ответ: 1		
	Ответ: 2	1	
	Ответ: 3		
	Ответ: 4		
23	Для какого из указанных значений числа X истинно выражение NOT(X>2) OR (X>6)?		
	Ответ: 2	1	
	Ответ: 3		
	Ответ: 4		
	Ответ: 5		

Контрольная работа

Вариант № 1

Задание 1.

Цепочка из трех бусин, помеченных латинскими буквами, формируется по следующим правилам. В конце цепочки стоит одна из бусин А, В, С. На первом месте одна из бусин В, D, С, которой нет на третьем месте. В середине — одна из бусин А, С, Е, В, не стоящая на первом месте. Какая из перечисленных цепочек создана по этим правилам?

1)СВВ 2)ЕАС 3)BCD 4)BCB

Задание 2.

Определить значения переменных а и в после исполнения данного алгоритма, построив трассировочную таблицу.

a=11

a=112-a*9

v=51-33+a*2

a=a-2

a=a-v/a+39

где / - операция целочисленного деления.

Задание 3.

Сколько раз выполнится тело цикла, чему равны значения переменных а и в после выполнения фрагмента алгоритма?

Задание 4.

У исполнителя Утроитель две команды, которым присвоены номера:

1. Вычти 1

2. Умножь на3

Первая из них уменьшает число на экране на 1, вторая - утраивает его. Запишите порядок команд в алгоритме получения из числа 5 числа 26, содержащем не более 5 команд, указывая лишь номера команд.

Задание 5.

Алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала записывается буква, следующая в алфавите после первой буквы исходной цепочки, затем две последние буквы исходной цепочки в обратном порядке и, наконец, первая буква исходной

цепочки. Получившаяся цепочка является результатом работы алгоритма. Например, если исходная цепочка символов была СОН, то результатом алгоритма будет цепочка ТНОС.

Дана цепочка символов КОШ. Какая цепочка символов получится, если к данной цепочке применить алгоритм трижды (т.е. алгоритм применяется к данной цепочке, затем к результату его работы, а затем ко второму результату работы алгоритма)?

Русский алфавит:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

Вариант 2.

Задание 1.

В формировании цепочки из четырех бусин используются некоторые правила. В конце цепочки стоит одна из бусин Р, N, Т, О. На первом — одна из бусин Р, R, Т, О; которой нет на третьем месте. На третьем месте — одна из бусин О, Р, Т, не стоящая в цепочке последней. Какая из перечисленных цепочек могла быть создана с учетом этих правил?

1)PORT 2)TTTO 3)TTOO 4)OORO

Задание 2.

Определить значения переменных а и в после исполнения данного алгоритма, построив трассировочную таблицу.

$a=24$

$b=18$

$a=a+b/3+16$

$b=a+2*b$

$a=b/a+b/2$

где / - операция целочисленного деления.

Задание 3.

Сколько раз выполнится тело цикла, чему равны значения переменных а и в после выполнения фрагмента алгоритма?

Задание 4.

У исполнителя Утроитель две команды, которым присвоены номера:

1. Вычти 2

2. Умножь на 3

Первая из НИХ уменьшает число на экране на 2, вторая - утраивает его. Запишите порядок команд в программе получения из 11 числа 13, содержащей не более 5 команд, указывая лишь номера команд.

Задание 5.

Алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала записываются первые две буквы исходной цепочки в обратном порядке, затем буква, следующая в алфавите за последней буквой исходной цепочки, и, наконец, исходная цепочка символов, записанная в обратном порядке. Получившаяся цепочка является результатом работы алгоритма. Например, если исходная цепочка символов была СОН, то результатом алгоритма будет цепочка ОСОНОС.

Дана цепочка символов НИЛ. Какая цепочка символов получится, если к данной цепочке применить алгоритм дважды (т.е. алгоритм применяется к данной цепочке, затем к результату его работы)?

Русский алфавит:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

Вариант 3.

Задание 1.

Для составления цепочек используются бусины, помеченные буквами А, В, С, D, E. На первом месте в цепочке стоит одна из бусин А, С, E. На втором — любая гласная, если первая буква согласная, и любая согласная, если первая гласная. На третьем месте — одна из бусин С, D, E, не стоящая в цепочке на первом месте. Какая из перечисленных цепочек создана по этим правилам?

1)CBE 2)ADD 3)ECE 4)EAD

Задание 2.

Определить значения переменных а и в после исполнения данного алгоритма, построив трассировочную таблицу.

$$a=15$$

$$b=2*a+24$$

$$a= b/2+a$$

$$b=2*a+2*b+9$$

$$a=b/a+2*a$$

где / - операция целочисленного деления.

Задание 3.

Сколько раз выполнится тело цикла, чему равны значения переменных I и A после выполнения фрагмента алгоритма?

Задание 4.

У исполнителя Калькулятор две команды, которым присвоены номера:

1. Прибавь 2

2. Умножь на 3

Выполняя первую из них, Калькулятор прибавляет к числу на экране 2, а выполняя вторую, утраивает его. Запишите порядок команд в программе получения из 0 числа 28, содержащей не более 6 команд, указывая лишь номера команд.

Задание 5.

Алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала записывается буква, предшествующая в алфавите первой букве исходной цепочки, затем исходная цепочка символов, записанная в обратном порядке. Получившаяся цепочка является результатом работы алгоритма. Например, если исходная цепочка символов была СОН, то результатом алгоритма будет цепочка РНОС.

Дана цепочка символов РОК. Какая цепочка символов получится, если к данной цепочке применить алгоритм трижды (т.е. алгоритм применяется к данной цепочке, затем к результату его работы, а затем ко второму результату работы)?

Русский алфавит:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

Вариант 4.

Задание 1.

Цепочка из трех бусин формируется по следующим правилам. На первом месте в цепочке стоит одна из бусин А, Б, В. На втором — одна из бусин Б, В, Г. На третьем месте — одна из бусин А, В, Г, не стоящая в цепочке на первом или втором месте. Какая из следующих цепочек создана по этим правилам?

1)АГБ 2)ВАГ 3)БГГ 4)ББГ

Задание 2.

Определить значения переменных а и в после исполнения данного алгоритма, построив трассировочную таблицу.

$$a=48$$

$$b=a/4-5$$

$$a= a-b$$

$$b=2*a+37$$

$$a=b/a+2*a$$

где / - операция целочисленного деления.

Задание 3.

Сколько раз выполнится тело цикла, чему равны значения переменных I, A и R после выполнения фрагмента алгоритма?

Задание 4.

Исполнитель Калькулятор имеет только две команды, которым присвоены номера:

1. Вычти 3
2. Умножь на 2

Выполняя команду номер 1, Калькулятор вычитает из числа на экране 3, а выполняя команду номер 2, умножает число на экране на 2. Напишите программу, содержащую не более 5 команд, которая из числа 5 получает число 25. Укажите лишь номера команд.

Задание 5.

Алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала записывается буква, предшествующая в алфавите последней букве исходной цепочки, затем первые две буквы исходной цепочки, и наконец, буква, следующая в алфавите за первой буквой исходной цепочки символов. Получившаяся цепочка является результатом работы алгоритма. Например, если исходная цепочка символов была СОН, то результатом алгоритма будет цепочка МСОТ.

Дана цепочка символов АРТ. Какая цепочка символов получится, если к данной цепочке применить алгоритм дважды (т.е. алгоритм применяется к данной цепочке, затем к результату его работы)?

Русский алфавит:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

Перечень тем докладов по теме: История и классификация языков программирования:

- 1 История и классификация языков программирования.
- 2 Системы программирования.
- 3 Процедурные и неперодурные языки программирования.
- 4 Объектно-ориентированное программирование.
- 5 Язык программирования Паскаль. История создания. Использование в современности.
- 6 Биография Никлауса Вирта.

Тестовое задание:

Язык программирования Pascal . Операторы ввода, вывода, присваивания

Номер вопроса	Вопрос	Верный ответ
1	Языки программирования - это...	
	искусственные языки	1
	естественные языки	
	смешанные языки	
2	Как называется средство для преобразования текстов из языков программирования высокого уровня в машинный код?	
	маршрутизатор	
	транслятор	1
	автокод	
	контроллер	
3	Какой из языков программирования является машинно-зависимым?	
	бейсик	
	паскаль	

	асемблер	1
	си	
4	Какой язык программирования относится к языкам низкого уровня?	
	асемблер	1
	бейсик	
	фортран	
	кобол	
5	Какой язык программирования относится к языкам высокого уровня?	
	асемблер	
	автокод	
	си	1
6	Что такое алфавит языка программирования?	
	фиксированный для данного языка набор символов	1
	буквы, используемые в данном языке	
	правила записи объектов языка	
7	Что такое идентификатор?	
	фиксированный для данного языка набор символов	
	последовательность символов, применяющаяся для имен констант, переменных, функций	1
	средство трансляции в машинный код	
	средство, определяющее тип данных	
8	К какому типу данных относится число 23.56?	
	string	
	integer	
	longint	
	real	1
9	Выберите пример правильного имени переменной:	
	FFG11	1
	6PO	
	АПРЛ	
	FF+HH	
10	Выберите пример неправильного имени переменной:	
	DFR56	
	ИВЛ23	1
	_VFA	
	DH55DF	

11	Выберите идентификатор целого типа данных	
	word	1
	double	
	extended	
	char	
12	Выберите идентификатор логического типа данных	
	word	
	boolean	1
	extended	
	char	
13	Выберите пример описания интервального типа данных	
	type season = (winter, spring, summer, autumn);	
	var color: (red, yellow, green);	
	var fad: = 100..200;	1
	var cad: integer;	
14	Чему равен результат операции (47 mod 15)?	
	7.	
	2.	1
	4.	
	10.	
15	Величина, не изменяющая свое значение в процессе выполнения программы называется...	
	константа	1
	переменная	
	функция	
	выражение	
16	Чему равен результат операции (67 div 10)?	
	7.	
	6.	1
	4.	
	10.	
17	Чему равно значение выражения Frac(45.56)?	
	46.	
	45.	
	0.45	
	0.56	1
18	Чему равно значение выражения Inc(5,5)?	
	5.	
	10.	1

	0.	
	55.	
19	Чему равно значение выражения Dec(15,10)?	
	25.	
	10.	
	5.	1
	15.	
20	Чему равно значение выражения Inc(20)?	
	19.	
	10.	
	2.	
	21.	1
21	Чему равно значение выражения Dec(15)?	
	15.	
	14.	1
	16.	
	25.	
22	Функция SQR предназначена для:	
	возведения аргумента в третью степень	
	вычисления квадратного корня из аргумента	
	возведения аргумента во вторую степень	1
	вычисления корня третьей степени из аргумента	
23	С понятием переменная не связано следующее понятие:	
	тип	
	идентификатор	
	объект	1
	значение	
24	Внутреннюю форму представления данных в ЭВМ определяет:	
	идентификатор	
	транслятор	
	тип данных	1
	операция	
25	Раздел описания переменных задается служебным словом:	
	const	
	uses	
	type	
	var	1

26	Тело программы начинается после служебного слова:	
	begin	1
	var	
	uses	
	type	
27	Создателем языка программирования Pascal является:	
	Паскаль	
	Вирт	1
	Лавлейс	
	Куртс	

Пакет преподавателя:

№ вопроса	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
-----------	---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Контрольная работа:

Программирование разветвляющихся алгоритмов

ВАРИАНТ 1

Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} \frac{4x^2 + 1}{x - 5}, & \text{если } x < 5, \\ 3x^2 - 2, & \text{если } x \geq 5. \end{cases}$$

ВАРИАНТ 2

Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} \frac{5x^2 + 2}{x + 4}, & \text{если } x > -4, \\ 3x^2 + 7, & \text{если } x \leq -4 \end{cases}$$

ВАРИАНТ 3

Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} \frac{7x^2 - 1}{2x + 6}, & \text{если } x < -3, \\ 4x^2 - 5, & \text{если } x \geq -3 \end{cases}$$

ВАРИАНТ 4

Составьте программу вычисления значения функции, используя полное ветвление:

$$y = \begin{cases} \frac{9x^2 + 5}{3x + 12}, & \text{если } x < -4, \\ 4x^2 - 7, & \text{если } x \geq -4 \end{cases}$$

Программирование циклических алгоритмов

Вариант 1

1. Составить программу вычисления суммы и произведения четных чисел из промежутка от 1 до 10.

Вариант 2

2. Составьте программу вывода на экран всех нечетных трехзначных чисел.

Вариант 3

3. Дано целое число N(больше 0). Найти сумму $1+1/2+1/3+\dots+1/N$ (вещественное число)

Пакет преподавателя

Вариант 1

```
program jjj;
var a: array [1..10] of integer; i,summa,proizv:integer;
begin
for i:=1 to 10 do begin
  Writeln('напишите элемент массива номер ',i); read(a[i]);
  end;
For i:=1 to 10 do begin
  if a[i] mod 2=0 then summa:=summa+a[i] else
  if (a[i]<=10) and (a[i]>=1) then  proizv:=proizv*a[i];
  end;
Writeln(summa,' ',proizv); readln;
end.
```

Вариант 2

```
var i,s: integer;
begin
i := 100;
s := 0;
repeat
s := s + i;
i := i + 5;
until i <= 1000;
writeln(s);
end.
```

Вариант 3

```
var H: real;
  N, i: integer;
begin
  H := 0;
  readln(N);
  for i := 1 to N do
    H := H + 1/i;
  writeln(H:6:4)
end.
```

Программирование циклических алгоритмов

- 1) Оператор цикла с предусловием, формат команды.
- 2) Заголовок и тело цикла с предусловием.
- 3) Оператор цикла с параметром, формат команды.
- 4) Заголовок и тело цикла с параметром.
- 5) Шаг изменения счетчика цикла с параметром.
- 6) Использование служебных слов to и downto в заголовке цикла с параметром.
- 7) Оператор цикла с постусловием, формат команды, условие и тело цикла.
- 8) Организация повтора программы с использованием цикла с постусловием.

9) Использование составного оператора в теле цикла.

Язык программирования

Номер вопроса	Вопрос	Верный ответ
1	Выберите правильную форму описания строки s на языке Паскаль.	
	var s:length(s);	
	var s:set of char;	
	var s:string;	1
	type s of string;	
2	Выберите правильную форму описания множества, состоящего из целых чисел.	
	var M:set of integer;	
	var K:set of char;	
	var D:set of string;	
	var T:set of byte;	1
3	Функция, определяющая длину строки s.	
	length(s);	1
	concat(s);	
	pos(s);	
	insert(s);	
4	Назначение функции CHR(N).	
	Определяет по символу порядковый номер	
	Определяет по порядковому номеру символ	1
	Объявляет символьный тип данных	
	Выдает номер первого вхождения символа N в строке s	
5	Назначение функции ORD(C).	
	Определяет по символу порядковый номер	1
	Определяет по порядковому номеру символ	
	Объявляет символьный тип данных	
	Выдает номер первого вхождения символа N в строке s	
6	Назначение функции POS(s1,s).	
	Вставляет подстроку s1 в строку s	
	Удаляет из строки s подстроку s1	
	Ищет первое вхождение подстроки s1 в строке s	1
	Сцепляет строки s1,s	
7	Назначение функции Concat(s1,s).	

	Вставляет подстроку s1 в строку s	
	Удаляет из строки s подстроку s1	
	Ищет первое вхождение подстроки s1 в строке s	
	Сцепляет строки s1,s	1
8	Назначение функции Str(x,s).	
	Вставляет символ x в строку s	
	Удаляет из строки s символ x	
	Ищет первое вхождение подстроки x в строке s	
	Преобразует числовое значение x в строку s	1
9	Назначение функции Val(s,x,error).	
	Преобразует строковое значение s в числовое x	1
	Вставляет символ x в строку s	
	Удаляет из строки s символ x	
	Преобразует числовое значение x в строку s	
10	Определите результат выполнения операции [1,2,5,9]*[1,3,5,7]	
	[1,2,3,5,7,9]	
	[1,2,3]	
	[1,5]	1
	[2,9]	
11	Определите результат выполнения операции [1,2,5,9]+[1,3,5,7]	
	[1,2,3,5,7,9]	1
	[1,2,3]	
	[1,5]	
	[2,9]	
12	Определите результат выполнения операции [1,2,5,9]-[1,3,5,7]	
	[1,2,3,5,7,9]	
	[1,2,3]	
	[1,5]	
	[2,9]	1
13	Определите результат выполнения операции 'a' in ['a','b','c','d']	
	false	
	true	1
	['a']	
	['a','b','c','d']	
14	Операция in, применяемая в Паскале ко множественному типу, используется для...	
	включения нового элемента во множество	
	исключения элемента из множества	

	проверки принадлежности элемента множеству	1
	объединения двух множеств	
15	Определите результат выполнения операции 7 in [1..5]	
	false	1
	true	
	[1..5,7]	
	[1..7]	
16	Что выведется на экран в результате выполнения следующего фрагмента программы? S:='оса'; s1:=copy(s,1,2); k:='новатор'; delete(k,5,3); k1:=concat(s1,k); write(k1);	
	основа	1
	осор	
	новатороса	
	аватар	
17	Что выведется на экран в результате выполнения следующего фрагмента программы? S:='ЧЭМК'; L:=length(s); write(L);	
	6.	1
	5.	
	4.	
	3.	
18	Что выведется на экран в результате выполнения следующего фрагмента программы? S:='HELLO'; k:=pos('L',s); write(k);	
	1.	
	2.	
	3.	1
	4.	
19	Для получения из строки s:='формальность' строки s:='форма' необходимо использовать команду...	
	insert(s,1,5);	
	copy(s,5,1);	
	delete(s,6,7);	1
	delete(s,1,5);	

20	Для получения из строки s:='огород' строки s:='город' необходимо использовать команду...	
	delete(s,1,5);	
	delete(s,1,1);	1
	delete(s,2,6);	
	delete(s,1,2);	

Пакет преподавателя:

№ вопроса	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Верный ответ	3	4	1	2	1	3	4	4	1	3	1	4	2	3	1	1	1	3	3	2

Язык программирования Pascal вопросы:

- 1) Описание одномерного массива.
- 2) Использование операторов цикла при решении задач обработки массивов.
- 3) Выделение отдельного элемента одномерного массива.
- 4) Поиск элементов в массиве, кратных заданному числу (условие кратности). Привести фрагмент программы.
- 5) Описание двумерного массива.
- 6) Использование операторов цикла при решении задач обработки массивов.
- 7) Выделение отдельного элемента двумерного массива.
- 8) Свойства диагональных элементов квадратной матрицы.

Язык программирования Pascal вопросы:

- 1) Понятие подпрограммы, назначение подпрограмм.
- 2) Виды подпрограмм в языке программирования Pascal.
- 3) Состав подпрограммы.
- 4) Вызов подпрограммы.
- 5) Функции, описание функции, заголовок функции, тело функции.
- 6) Процедуры, описание процедуры, заголовок процедуры, тело процедуры.
- 7) Параметры-значение и параметры-переменные процедуры.
- 8) Формальные и фактические параметры, работа механизма передачи параметров.

Язык программирования Pascal вопросы:

- 1) Структурированный тип запись, определение, назначение.
- 2) Описание типа запись, используемые служебные слова.
- 3) Компоненты записи и их типы.
- 4) Массив записей, описание, назначение.
- 5) Составные имена.
- 6) Оператор присоединения и его область действия.

Вопросы, которые необходимо раскрыть при создании презентации:

- 1) Описание файловых переменных.
- 2) Установление связи между программой и файлом на диске.
- 3) Открытие файла для чтения, записи, добавления данных.
- 4) Запись данных в файл, чтение данных из файла.
- 5) Завершение работы с файлом.

Язык программирования Pascal:

Практические задания

Вариант 1

1. Что означает команда языка Pascal `Ellipse(50,100,150,200)`
2. Что означает команда языка Pascal `SetBrushColor(clRed)`
3. Написать команду рисования прямоугольника, заливка зеленым цветом

Вариант 2

1. Что означает команда языка Pascal `SetPixel(100,200,clRed)`
2. Что означает команда языка Pascal `SetPenColor(clGreen)`
3. Написать команду рисования эллипса, заливка синим цветом

Вариант 3

1. Что означает команда языка Pascal `Rectangle(50,100,250,300)`
2. Что означает команда языка Pascal `Uses GraphABC`
3. Написать команду рисования круга, заливка красным цветом

Вариант 4

1. Что означает команда языка Pascal `FloodFill(100,150,clBlue)`
2. Что означает команда языка Pascal `SetPenWidth(5)`
3. Написать команду рисования линии красным цветом

Язык программирования C

Темы докладов:

- 1 Язык программирования C. История. Возможности.
- 2 Биография Д.Ритчи.

Язык программирования C

Тестирование

Инструкция для студентов

1. Последовательность и условия выполнения задания *все задания выполняются последовательно, самостоятельно.*
2. Вы можете воспользоваться *нет*
3. Максимальное время выполнения задания: 20 мин.
4. Перечень раздаточных и дополнительных материалов *нет*

Номер вопроса	Вопрос	Верный ответ
1	Выберите правильно записанное условие проверки числа x на четность на языке СИ:	
	<code>if (x%2=0)</code>	
	<code>if (x%2==2)</code>	
	<code>if (x mod 2 =0)</code>	
	<code>if (x%2==0)</code>	1

2	<pre>int mark; printf("Vvedite ocenku"); scanf("%d",&mark); if (mark == 4) printf("Vi udarnik.\n"); else if (mark > 4) printf("Vi otlichnik!\n"); else printf ("Ychitsya nado.\n");</pre> <p>С клавиатуры вводится 3, какой результат будет на экране?</p>	
	Vi otlichnik!	
	Vi udarnik.	
	Ychitsya nado.	1
3	Значение вещественной переменной нужно вывести следующим образом: выравнивание по левому краю, три знака после запятой. Выберите соответствующую команду.	
	printf("%f5.3",x);	
	printf("%d3",x);	
	printf("%f-8.3",x);	1
	printf("%-3f",x);	
4	Укажите правильно заданную команду для вывода вещественной переменной a.	
	printf("%f",a);	1
	scanf("%d",&a);	
	scanf("%f",&a);	
	printf("%d",a);	
5	&&	
	логическое "И"	1
	логическое "ИЛИ"	
	логическое "НЕ"	
6		
	логическое "И"	
	логическое "ИЛИ"	1
	логическое "НЕ"	
7	!	
	логическое "И"	
	логическое "ИЛИ"	
	логическое "НЕ"	1
8	"не равно" в СИ обозначается	
	==	
	<>	

	>>	
	!=	1
9	Чему равен результат pow(3,3)%5=	
	4.	
	5.	
	2.	1
	3.	
10	Описание вещественного типа с СИ	
	float	1
	int	
	real	
	char	
11	Какой спецификатор нужно использовать для вывода вещественного числа?	
	%d	
	%f	1
	%c	
	%i	
12	Какой спецификатор нужно использовать для вывода целого числа?	
	%d	1
	%f	
	%c	
	%s	
13	Какой спецификатор нужно использовать для вывода одного символа?	
	%d	
	%f	
	%c	1
	%i	
14	Требуется вывести вещественное число в СИ, так чтобы после запятой было 3 знака. Какой формат нужно использовать?	
	%3d	
	%3f	
	%4.3f	1
	%4.3d	
15	В каком формате появится число на экране при использовании команды: printf("%4.2f",6);	
	6.	
	6.00	1
	6.00000	
	6.000000	

16	Какой оператор в СИ находит остаток от деления?	
	/	
	mod	
	%	1
	&	
17	Оператор получения адреса	
	%	
	&&	
	&	1
	\n	
18	\n в СИ обозначает:	
	горизонтальная табуляция	
	перевод курсора на новую строку	1
	спецификатор целочисленного формата	
	спецификатор вещественного формата	
19	Оператор соотношения "равно" записывается:	
	=	
	:=	
	!=	
	==	1
20	Как записывается логическое "И" в СИ?	
	and	
	&	
	&&	1
	i	
21	Как записывается логическое "ИЛИ" в СИ?	
	ili	
	OR	
		1
22	Как записывается логическое "НЕ" в СИ?	
	!	1
	not	
	&	
23	Язык СИ разработал...	
	Том Куртц	

	Никлаус Вирт	
	Деннис Ритчи	1
	Билл Гейтс	
24	Чему равен результат выполнения операции $15\%3=...$	
	0.	1
	1.	
	5.	
	3.	
25	Какую библиотеку нужно подключить для работы операторов <code>scanf()</code> и <code>printf()</code> в СИ?	
	<code><math.h></code>	
	<code><stdio.h></code>	1
	<code><conio.h></code>	
	<code><stdlib.h></code>	
26	Какую библиотеку нужно подключить для работы оператора <code>fabs()</code> в СИ?	
	<code><math.h></code>	1
	<code><stdio.h></code>	
	<code><conio.h></code>	
	<code><stdlib.h></code>	
27	Как находится модуль вещественного числа в СИ?	
	<code>mod</code>	
	<code>fabs</code>	1
	<code>abs</code>	
28	Как записать на СИ "корень квадратный из x"?	
	$x ^ 0.5$	
	<code>power(x,2)</code>	
	<code>pow(x,1/2)</code>	1
	<code>pow(x,y,2)</code>	
29	Оператор форматированного вывода:	
	<code>scanf()</code>	
	<code>print()</code>	
	<code>printf()</code>	1
	<code>write()</code>	
30	Оператор ввода:	
	<code>scanf()</code>	1
	<code>read()</code>	
	<code>printf()</code>	
	<code>input()</code>	

31	Укажите правильно заданную команду для ввода вещественной переменной a.	
	printf("%f",a);	
	scanf("%d",&a);	
	scanf("%f",&a);	1
	scanf("%-f",a);	

Пакет преподавателя:

№ вопроса	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Верный ответ	4	3	3	1	1	2	3	4	3	1	2	1	3	3	2	3	3	2	4	3	4	2	3	1	2	1	2	3	3	1	3	

Язык программирования C

Проверочная работа

Вариант 1

Написать программу на языке программирования C. В массиве из 20 целочисленных элементов, заданных случайным образом из отрезка [-15;15] найти максимальный элемент и его номер.

Вариант 2

Написать программу на языке программирования C. В массиве из N(вводится с клавиатуры) целочисленных элементов, заданных случайным образом из отрезка [-5;5] подсчитать количество нечетных положительных элементов, вывести найденные элементы на печать.

Вариант 3

Написать программу на языке программирования C, которая определяет в целочисленной матрице, состоящей из элементов из отрезка [-5;5], номер строки, которая содержит наибольшее количество элементов, равных нулю.

Язык программирования C

Тестирование

Номер вопроса	Вопрос	Верный ответ
1	If (выражение) оператор_1 ;else оператор_2;	
	структура оператора условия	1
	структура цикла с параметром	
	структура оператора множественного выбора	
	структура условного выражения	
2	int mark; printf("Vvedite ocenku"); scanf("%d",&mark); if (mark == 4) printf("Vi udarnik.\n"); else if (mark > 4) printf("Vi otlichnik!\n"); else printf ("Ychitsya nado.\n");	
	С клавиатуры вводится 3, какой результат будет на экране?	
	Vi otlichnik!	

	Vi udarnik.	
	Ychitsya nado.	1
3	<pre>int mark; printf("Vvedite ocenku"); scanf("%d",&mark); if (mark == 4) printf("Vi udarnik.\n"); else if (mark > 4) printf("Vi otlichnik!\n"); else printf ("Ychitsya nado.\n");</pre> <p>С клавиатуры вводится 5, какой результат будет на экране?</p>	
	Vi otlichnik!	1
	Vi udarnik.	
	Ychitsya nado.	
4	<pre>int a,b,i; printf("Vvedite a\n"); scanf("%d",&a); printf("Vvedite b\n"); scanf("%d",&b); if(a == 0) printf("Net resheniya.\n"); else printf("Ect reshenie=%2.3f\n",b/a);</pre> <p>С клавиатуры ввели 5 и 10. Какой будет результат?</p>	
	Ect reshenie=0.5	
	Ect reshenie=2	1
	Net resheniya	
5	Оператор «&&» обозначает	
	логическое "И"	1
	логическое "ИЛИ"	
	логическое "НЕ"	
6	Оператор « » обозначает	
	логическое "И"	
	логическое "ИЛИ"	1
	логическое "НЕ"	
7	Оператор «!» обозначает	
	логическое "И"	
	логическое "ИЛИ"	
	логическое "НЕ"	1
8	"не равно" в СИ обозначается	
	==	

	<>	
	><	
	!=	1
9	pow(3,3)%5=	
	4.	
	5.	
	2.	1
	3.	
10	Описание вещественного типа с СИ	
	float	1
	int	
	real	
	char	
11	Какой спецификатор нужно использовать для вывода вещественного числа?	
	%d	
	%f	1
	%c	
	%i	
12	Какой спецификатор нужно использовать для вывода целого числа?	
	%d	1
	%f	
	%c	
	%s	
13	Какой спецификатор нужно использовать для вывода одного символа?	
	%d	
	%f	
	%c	1
	%i	
14	Требуется вывести вещественное число в СИ, так чтобы после запятой было 3 знака. Какой формат нужно использовать?	
	%3d	
	%3f	
	%4.3f	1
	%4.3d	
15	В каком формате появится число на экране при использовании команды: printf("%4.2f",6);	
	6.	
	6.00	1
	6.00000	

	6.000000	
16	Какой оператор в СИ находит остаток от деления?	
	/	
	mod	
	%	1
	«&&»	
17	Описание строки s из 10 символов в СИ нужно выполнить:	
	string s(10);	
	char s[10];	
	char s[11];	1
	string s[10];	
18	\n в СИ обозначает:	
	горизонтальная табуляция	
	перевод курсора на новую строку	1
	спецификатор целочисленного формата	
	спецификатор вещественного формата	
19	Оператор соотношения "равно" записывается:	
	=	
	:=	
	!=	
	==	1
20	Функция strlen() ...	
	вычисляет длину строки	1
	выполняет сцепление строк	
	проверяет вхождение одной строки в другую	
	выполняет преобразование числа в строку	
21	Как записывается логическое "ИЛИ" в СИ?	
	ili	
	OR	
		1
22	Как записывается логическое "НЕ" в СИ?	
	!	1
	not	
	&&	
23	Язык СИ разработал...	

	Том Куртц	
	Никлаус Вирт	
	Деннис Ритчи	1
	Билл Гейтс	
24	15%3=...	
	0.	1
	1.	
	5.	
	3.	
25	Какую библиотеку нужно подключить для работы операторов scanf() и printf() в СИ?	
	<math.h>	
	<stdio.h>	1
	<conio.h>	
	<stdlib.h>	
26	Какую библиотеку нужно подключить для работы оператора fabs() в СИ?	
	<math.h>	1
	<stdio.h>	
	<conio.h>	
	<stdlib.h>	
27	Как находится модуль вещественного числа в СИ?	
	mod	
	fabs	1
	abc	
28	Как записать на СИ "корень квадратный из x"?	
	x ^ 0.5	
	power(x,2)	
	pow(x,0.5)	1
	pow(x,y,2)	
29	Оператор форматированного вывода:	
	scanf()	
	print()	
	printf()	1
	write()	
30	Оператор ввода:	
	scanf()	1
	read()	
	printf()	

	input()	
31	Укажите правильно заданную команду для ввода вещественной переменной a.	
	printf("%f",a);	
	scanf("%d",&a);	
	scanf("%f",&a);	1
	scanf("%-f",a);	
32	Укажите правильно заданную команду для вывода вещественной переменной a.	
	printf("%f",a);	1
	scanf("%d",&a);	
	scanf("%f",&a);	
	printf("%d",a);	
33	Значение вещественной переменной нужно вывести следующим образом: выравнивание по левому краю, три знака после запятой. Выберите соответствующую команду.	
	printf("%f5.3",x);	
	printf("%d3",x);	
	printf("%f-8.3",x);	1
	printf("%-3f",x);	
34	Служебное слово switch используется в конструкции...	
	условия	
	множественного выбора	1
	цикла с параметром	
	форматированного вывода	
35	Выберите правильно записанное условие проверки числа x на четность на языке СИ:	
	if (x%2=0)	
	if (x%2==2)	
	if (x mod 2 =0)	
	if (x%2==0)	1
36	Переход на новую строку в языке Си осуществляется:	
	t	
	%d	
	n	1
	&&	
37	Укажите количество ошибок при использовании условного оператора на языке Си: <pre> if (a==5) { printf("Good"); b=a; </pre>	

	<pre> } </pre>	
	1.	
	2.	
	3.	
	нет ошибок	1
38	<p>Укажите неправильный вариант для обозначении имен переменных:</p> <p>A: VAZ B: _sww C: 5a D: re4 E: kl*s</p>	
	<i>C и E</i>	1
	<i>Только E</i>	
	<i>Все неправильные</i>	
	<i>Все правильные</i>	
39	<p>Каким словом обозначается «главная» функция в программе на языке Си:</p>	
	<i>mail</i>	
	<i>main</i>	1
	<i>#include</i>	
	<i>return</i>	
40	<p>Укажите какое значение будет выведено на экран:</p> <pre> h=4.3767; j=0; printf("%.1f",h); </pre>	
	4.0	
	0.3767	
	0.0	
	4.4	1
41	<p>Укажите значение элемента массива K[2] после завершения цикла:</p> <pre> int K[78]; for(i=0;i<=34;i+=2) K[i]=i-10; </pre>	
	-8	1
	2	
	34	
	-10	
42	<p>Укажите количество ошибок в строке, подключающей модуль для работы операторов ввода и вывода: <code>#incude <studio.n></code></p>	

	1.	
	2.	
	3.	1
	нет ошибок.	
43	Целый тип в языке Си обозначается:	
	<i>integer</i>	
	<i>float</i>	
	<i>int</i>	1
	Такого типа не существует	
44	Укажите количество ошибок при использовании условного оператора на языке Си: <pre> if (a<-3); { printf(“%d\n”,a); a++; } </pre>	
	1.	1
	2.	
	3.	
	нет ошибок.	
45	Укажите количество ошибок при использовании оператора цикла на языке Си: <pre> for(t=-4,t<4;t=t+0.5 { printf(“%f\n”,t); } </pre>	
	1.	
	2.	1
	3.	
	4.	
46	Укажите неправильный вариант для обозначении имен переменных: A: a45 B: y_22 C: pус D: f% E: _1	
	B	
	C	
	D	
	C и D	1
47	Подключение какой библиотеки необходимо для использования математических функций:	
	<i>math.h</i>	1
	<i>stdio.h</i>	

	<code>#include</code>	
	<code>conio.h</code>	
48	Укажите какое значение будет выведено на экран: <pre>h=-6.278; j=6; printf(“%4.2f”,j);</pre>	
	6.00	1
	-6.28	
	-6.00	
	6.28	
49	Укажите значение элемента массива K[13] после завершения цикла: <pre>int K[13]; for(i=0;i<13;i++) K[i]=i;</pre>	
	13.	
	0.	
	Массив задан неправильно	
	Элемент K[13] не определится	1
50	Укажите количество ошибок в строке: <pre>print(“%d %d”,y,&y);</pre>	
	1.	
	2.	1
	3.	
	Ошибок нет.	
51	Для подключения модулей используется:	
	<code>#include</code>	1
	<code>stdio.h</code>	
	<code>main</code>	
	<code>getch()</code>	
52	Тип переменной указывается:	
	<i>через : после переменной</i>	
	<i>после переменной</i>	
	<i>указывать не обязательно</i>	
	<i>до ее написания</i>	1
53	Укажите количество ошибок при использовании условного оператора на языке Си: <pre>if (c<>3) { G=sin(c); printf(“%f\n”,G); c--; };</pre>	

	1.	
	2.	1
	3.	
	Ошибок нет.	
54	Укажите количество ошибок при использовании оператора цикла на языке Си: <pre>for (g=0,g<13,g++); { printf(“%f\n”,g); }</pre>	
	1.	
	2.	
	3.	1
	4.	
55	Подключение какой библиотеки необходимо для использования функции <code>getch()</code> ?	
	<code>stdio.h</code>	
	<code>stdlib.h</code>	
	<code>getch.h</code>	
	<code>conio.h</code>	1
56	Укажите значение элемента массива <code>K[0]</code> после завершения цикла: <pre>int K[4]; for(i=3;i>0;i=i-1) K[i]=i*i; K[0]=K[2];</pre>	
	4.	1
	2.	
	0.	
	1.	
57	Укажите количество ошибок при использовании условного оператора: <pre>if e=12 printf(“%d”,e);</pre>	
	1.	
	2.	1
	4.	
	Нет ошибок.	
58	Для использования функции <code>strlen()</code> необходимо подключить библиотеку:	
	<code>stdio.h</code>	
	<code>conio.h</code>	
	<code>string.h</code>	1
	<code>stdlib.h</code>	

59	Укажите диапазон случайных чисел для random(12)-6:	
	от 0 до 6	
	от -6 до 0	
	от -6 до 5	1
	от -6 до -1	
60	Укажите количество ошибок при использовании условного оператора на языке Си: <pre> while (d!=13) { G=cos(sqrt(d)); printf(“%f\n”,G); d--; } </pre>	
	1.	
	2.	
	3.	
	Нет условного оператора.	1
61	<pre> while (d!=13) { G=cos(sqrt(d)); printf(“%f\n”,G); d--; } </pre> <p>Какой оператор используется в коде программы?</p>	
	оператор условия	
	оператор цикла с параметром	
	оператор цикла с предусловием	1
	оператор цикла с постусловием	
62	Укажите значение элемента массива K[4] после завершения цикла: <pre> int K[100]; for(i=3;i<100;i++) { if(i==4) K[i]=10; else K[i]=2*i; } </pre>	
	100.	
	4.	
	8.	
	10.	1
63	Каким символом завершается программа на языке Си:	
	«}»	1
	«getch»	
	«.»	

	«return»	
64	Можно ли в операторе условия if не ставить скобки {}	
	<i>Всегда надо ставить</i>	
	<i>Можно, если нет else</i>	
	<i>Нельзя</i>	
	<i>Можно, если блок состоит только из одного оператора</i>	1
65	Целочисленный массив обозначается:	
	<i>float A[4]</i>	
	<i>int A[13]</i>	
	<i>int A[1][3]</i>	
	<i>2 или 3 вариант</i>	1
66	Укажите количество ошибок при использовании условного оператора на языке Си: <pre> if (x<10) { printf(“%f\n”,x); x++; } else printf(“end”); </pre>	
	1.	
	2.	
	3.	
	Нет ошибок.	1
67	Какой модуль необходимо подключить для функции random?	
	<i>stdio.h</i>	
	<i>stdlib.h</i>	1
	<i>conio.h</i>	
	<i>randomize.h</i>	
68	Какой из операторов не является оператором цикла?	
	<i>for</i>	
	<i>while</i>	
	<i>if</i>	1
	<i>2 и 3 варианты</i>	
69	Двумерный массив обозначается:	
	<i>W[3][5]</i>	1
	<i>W[3;5]</i>	
	<i>W[3,5]</i>	
	<i>W[3],[5]</i>	
70	В каком порядке следует подключать модули?	

	<i>В порядке следования необходимых функций</i>	
	<i>В строго определённом</i>	
	<i>Не имеет значения</i>	1
	<i>Их можно вообще не подключать</i>	
71	a=5 b=7 После выполнения операции a+=b значение переменной a равно:	
	12.	1
	5.	
	35.	
	операция не имеет смысла	
72	int a,b; float c; printf("Vvedite a\n"); scanf("%d",&a); printf("Vvedite b\n"); scanf("%d",&b); c=a/b; printf("c=%3.2f\n",c); С клавиатуры вводятся 12 и 5. Какой результат выведется на экран?	
	c=0.05	
	c=2.00	1
	c=2.40	
	c=3.20	
73	В каком выражении использован оператор инкремента?	
	4*a--	
	a*=b	
	x*y++	1
	ceil(x)	
74	Оператор, уменьшающий значение переменной на единицу - это ...	
	fabs	
	инкремента	
	floor	
	дкремента	1

Пакет преподавателя:

№ вопроса	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
Верный ответ	1	3	1	2	1	2	3	4	3	1	2	1	3	3	3	2	3	3	2	4	1	4	2	3	1	2	1	2	3

№ вопроса	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58
Верный ответ	3	1	3	1	3	2	4	3	4	1	2	4	1	3	3	1	2	4	1	1	4	2	1	4	2	3	4	1	2

№ вопроса	5	6	6	6	6	6	6	6	6	6	6	7	7	7	7	7
Верный ответ	3	3	4	3	4	1	4	4	4	2	1	3	1	2	3	4

Алгоритм. Исполнители алгоритма

Инструкция для студентов

1. Последовательность и условия выполнения задания: *изучение и выполнение заданий, составление отчета, студенты, выполнившие и сдавшие практическое задание раньше остальных, могут получить дополнительную оценку, выполнив задания из пункта «Дополнительно на оценку»*
2. *Вы можете воспользоваться конспектами, учебником*
3. Максимальное время выполнения задания: *2 часа.*
4. Перечень раздаточных и дополнительных материалов *методические указания к лабораторным работам*

Практическое задание

Цель работы: Научиться составлять циклические алгоритмы, создавать программы, используя полученные знания.

Содержание работы.

Основные понятия.

1 Графический способ представления циклических алгоритмов (метод блок-схемы).

Проверка условий	
Цикл с параметром	

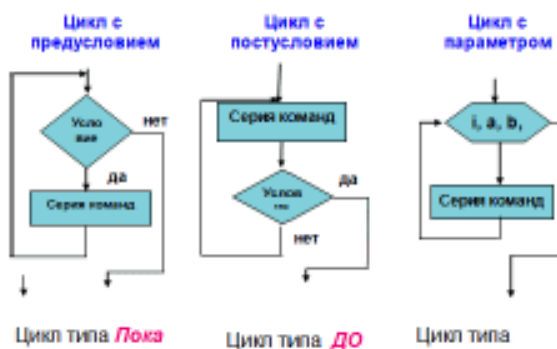
2 Алгоритм, в котором вычисления повторяются по одной и той же совокупности формул, называется циклическим

3 Существуют следующие конструкции для организации циклов:

- цикл с предусловием;
- цикл с постусловием;
- цикл со счетчиком.

4 Графическое представление циклических алгоритмов

Виды циклических алгоритмов



5 Циклы с предусловием используются тогда, когда выполнение цикла связано с некоторым логическим условием. Оператор цикла с предусловием имеет две части: условие выполнения цикла и тело цикла. При выполнении оператора цикла определенная группа операторов выполняется до тех пор, пока

определенное в операторе условие истинно. Если условие сразу ложно, то оператор не выполнится ни разу.

Составить и записать алгоритм решения задачи в графическом и словесно-формульном виде

Исходные данные:

Дано целое число N . Вычислить значение выражения $1 + \frac{2}{2!} + \frac{3}{3!} + \dots + \frac{N}{N!}$, результат вывести как действительное число.

Решение:

$$1 \ N! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot N$$

Значение выражения будем рассчитывать поэтапно.

$$\text{На первом шаге к значению суммы } S \text{ добавим } 1 = \frac{1}{1!} = \frac{1}{1}$$

$$\text{На втором } \frac{2}{2!} = \frac{2}{1 \cdot 2}$$

$$\text{На третьем } \frac{3}{3!} = \frac{3}{1 \cdot 2 \cdot 3}$$

...

$$\text{На } N\text{-ном } \frac{N}{N!} = \frac{N}{1 \cdot 2 \cdot \dots \cdot N}$$

В зависимости от заданного пользователем значения N количество шагов в алгоритме может быть разным, поэтому для определения шага добавим промежуточную переменную I , которая будет меняться от 1 до N

Так как к значению суммы на каждом шаге мы добавляем какое-то число, необходимо определить начальное значение, не влияющее на результат. Ясно, что это $S = 0$.

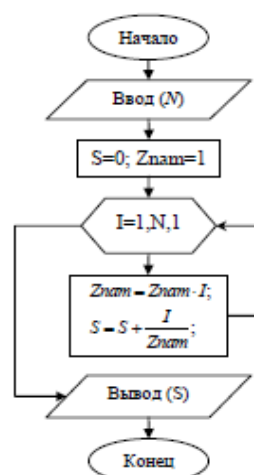
На каждом шаге считать факториал трудно, поэтому обратим внимание, что на любом этапе вычислений значение факториала отличается от предыдущего на число, равное значению шага, значит введем переменную для факториала: $Z_{\text{fak}} = 1$.

Получили общие формулы:

$$Z_{\text{fak}} = Z_{\text{fak}} \cdot I;$$

$$S = S + \frac{I}{Z_{\text{fak}}};$$

Задание



1. Начало;
2. Ввод (N);
3. $S=0$;
4. $Z_{\text{fak}}=1$;
5. Для $I = 1$ до N выполнить
нц
 $Z_{\text{fak}} = Z_{\text{fak}} \cdot I$;
 $S = S + \frac{I}{Z_{\text{fak}}}$;
кц
6. Вывод (S);
7. Конец.

Дополнительно на оценку

Задания к практической работе.

1 Даны действительные числа x, y . Вывести в порядке возрастания все целые числа, расположенные между x и y , а также количество этих чисел.

2 Даны действительные числа x, y . Вывести в порядке убывания все целые числа, расположенные между x и y , а также количество этих чисел.

3 Дано действительное число – цена 1 кг конфет. Вывести стоимость 1, 2 ... 15 кг конфет.

Алгоритм. Базовые алгоритмические конструкции

Практическое задание

Массив - это структура данных, что представляет собой совокупность фиксированного размера и конфигурации упорядоченных однородных независимых переменных.

Массив относится к так называемым структурированным данным, то есть таким, что имеют фиксированную внутреннюю структуру (организацию).

Массив характеризуется:

1. Количеством размерностей (количеством координат, необходимых для определения местонахождения нужного элемента массива).
2. Общим идентификатором (именем) для всех элементов массива.
3. Индексом или совокупностью индексов, которые определяют каждый отдельный элемент массива.

Одномерный массив (вектор) - имеет одну размерность

При обращении к отдельному элементу массива необходимо указать его индекс (местонахождение в массиве):

$A[7]$ $i:=7$; $A[i]$

Здесь i - индекс элемента массива, может быть только целого или натурального типа

Двумерные массивы (матричные) - имеют две размерности, $m*n$. Доступ к отдельному элементу массива осуществляется путем определения двух его координат: номера строки $i=1..m$ и столбца $j=1..n$

Операции предоставления выполняются аналогично:

$a[3,9]=8$; $B:=A[1,1]$

В трехмерном массиве для доступа к элементу необходимо указать три индекса $A[i,j,k]$. Можно создавать массивы с большей размерностью, но работа с массивами, размерность которых превышает 3, существенным образом усложняет алгоритм, поэтому, по возможности, необходимо избегать организации подобных структур данных.

Сортировка массивов

Методы сортировки можно разбить в соответствии с определяющими их принципами на три основные группы:

1. Сортировка с помощью вставки (by Insertion) или с помощью включения
2. Сортировка с помощью выбора (by Selection) или с помощью выделения
3. Сортировка с помощью обмена (by Exchange) или пузырьковая.

Каждая группа имеет прямой метод (самый простой) и улучшенный(усложненный) методы сортировки

I. Сортировка с помощью вставки

Принцип сортировки: массив распределяется на отсортированную и неотсортированную части. На первом шаге за отсортированную часть (последовательность) принимается первый элемент массива. Каждый следующий элемент из неотсортированной части вставляем в заранее отсортированную последовательность так, чтобы эта последовательность оставалась отсортированной.

При этом надо:

1. Найти место, куда нужно вставить этот элемент
2. Сдвинуть элементы, которые стоят справа в отсортированной части на одну позицию вправо.

3. На освобожденное место поставить элемент, который анализируется (вставляется).

Два способа выполнения этих действий:

1) каждый следующий элемент сравнивается с элементами в отсортированной части, находится место вставки, все следующие элементы сдвигаются на одну позицию вправо и после этого вставляется элемент;

2) элемент, который вставляется, последовательно, слева направо, сравнивается с любым из элементов в отсортированной части. Если нужно, элемент в отсортированной части сразу сдвигается на одну позицию вправо. Как только найдено нужное место вставки, элемент, который анализируется, вставляется на нужную позицию.

II. Сортировка с помощью прямого выбора

Принцип сортировки: массив также делится на отсортированную и неотсортированную части. На первом шаге весь массив - неотсортированный. В неотсортированной части находится минимальный (или максимальный) элемент и меняется местами с первым элементом неотсортированной части. Граница отсортированной части сдвигается вправо на 1. Процедура выполняется циклически, n-1 раз (последний элемент передвигать не надо)

III. Сортировка с помощью прямого обмена (пузырьковая)

Принцип сортировки: слева направо поочередно направляется сравнение двух соседних элементов. Если они не упорядочены между собой, то меняются местами. В базовом алгоритме прохождения массива и очередное приведение в порядок повторяются n-1 раз.

ПРИМЕР ПРОГРАММЫ

1. Нахождение среднего арифметического ста чисел

```
Program N1;  
var M: Array [1..100] of Integer; A: Real; I: Byte;  
begin Randomize;  
For I := 1 to 100 do M[I] := Random(500);  
For I := 1 to 100 do  
A := A + M[I];  
A := A / 100;  
Write('Среднее арифметическое всех чисел массива: ', A);  
end.
```

2. Найдите максимальное число среди введенных чисел и выведите его на экран.

```
Program N2;  
var M: Array [1..10] of Integer; Max: Integer; I: Byte;  
begin Writeln('Введите 10 чисел: ');  
For I := 1 to 10 do  
begin Write('N',i,': ');  
Readln(M[i]);  
end;  
Max := M[1];  
For I := 1 to 10 do  
if Max < M[i] then Max := M[i];  
Write('Максимальное число: ', Max);  
Readln;  
end
```

Задание №1

Найдите среднее арифметическое первых двухсот чисел.

Дополнительные задания на оценку

1. Задание № 2

Найдите минимальное число среди введенных 10 чисел.

2. Задание № 3

Найдите максимальное число среди введенных 15 чисел.

Программирование графики

Практическое задание

Для формирования графических изображений в Турбо Паскале предназначен стандартный модуль GRAPH. В нем содержится 79 графических процедур, функций, десятки стандартных констант и типов данных. В состав модуля GRAPH входит ряд программ драйверов для наиболее распространенных видеоадаптеров: CGA, MCGA, EGA, VGA, HERCULES, AT&T, 3270 PC. Эти драйверы хранятся в файлах с расширением BGI.

Скелет графической программы:

```
program имя;
USES GRAPH; {подключение модуля}
var GR,GM:integer;
begin
GR:=detect;
initgraph(gr,gm, ' '); {вызов процедуры GRAPH}
графические операторы
readln;
closegraph;
end.
```

INITGRAPH(драйвер, режим, 'путь') - подключает графический режим. Драйвер - это переменная целого типа, определяющая тип графического дисплея. Режим - это переменная типа integer, определяющая режим работы графического адаптера(GM). Путь - указывает путь программам с TPU.

DETECT - режим автоматического определения типа.

CLOSEGRAPH - прекращает работу графического адаптера и восстанавливает текстовый режим работы экрана.



Цвет

0 - черный

1 - синий

2 - зеленый

3 - голубой

4 - красный

5 - лиловый

6 - коричневый

7 - светло-серый

8 - темно-серый

9 - светло-синий

10 - светло-зеленый

11 - светло-голубой

12 - светло-красный

13 - светло-лиловый

14 - желтый

15 - белый

ГРАФИЧЕСКИЕ ОПЕРАТОРЫ:

3. SETCOLOR(цвет) - устанавливает текущий цвет для выводимых линий и символов;
4. SETBKCOLOR(цвет) - устанавливает цвет фона;
5. MOVETO(x,y) - переводит курсор в позицию x,y, точка при этом не высвечивается;
6. LINETO(x,y) - рисует линию от текущей позиции курсора до x,y;
7. LINEREL(Dx,Dy) - рисует линию от текущей позиции на относительное расстояние Dx,Dy;

8. PUTPIXEL(x,y,цвет) - выводит точку;
9. LINE(x1,y1,x2,y2) - выводит линию;
10. CIRCLE(x,y,r) - выводит окружность;
11. ARC(x,y,начальный угол,конечный угол,r) - выводит дугу (углы измеряются в градусах против часовой стрелки);
12. ELLIPSE(x,y,нач.угол,кон.угол,Rx,Ry) - эллипс;
13. RECTANGLE(x1,y1,x2,y2) - прямоугольник, где x1,y1 - координаты левого верхнего угла, x2,y2 - правого нижнего угла;
14. BAR(x1,y1,x2,y2) - закрашенный прямоугольник;
15. BAR3D(x1,y1,x2,y2,глубина,false или true) - параллелепипед; где x1,y1 - координаты левого верхнего угла; x2,y2 - координаты правого нижнего угла, true - верхняя грань есть, false - верхняя грань отсутствует.
16. FILLELLIPSE(x,y,Rx,Ry) - закрашенный эллипс;
17. SECTOR(x,y,нач.угол,кон.угол,Rx,Ry) - сектор закрашенный;
18. SETFILLESTYLE(штриховка,цвет) - каким цветом или какими штрихами будет закрашена замкнутая область;

Штриховка:

0 - фоном	7 - +++++
1 -сплошная	8 - x x x
2 - _ _ _ _	9- квадратами
3 - ///	10 - редкими точками
4 - толстыми///	11 - частыми точками
5 - \\\	12 - пользовательские
6 - толстыми \\\	

19. FLOODFILL(x,y,цвет,цвет границы) - штрихует произвольную замкнутую область;

Пример графических программ:

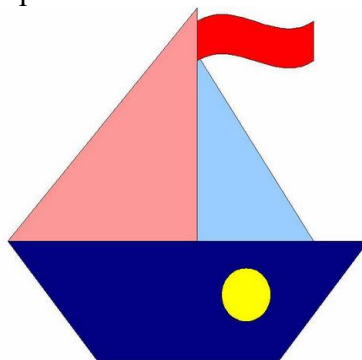
```

Программа №1.
program g1;
uses graph;
var gr, gm: integer;
begin
gr:=detect;
initgraph(gr, gm, ' ');
setcolor(3);
setbkcolor(14);
setfillstyle(8,4);
fillellipse(100,100,30,50);
setfillstyle(2,2);
circle(200,200,50);
floodfill(200,200,3);
readln;
closegraph;
end.

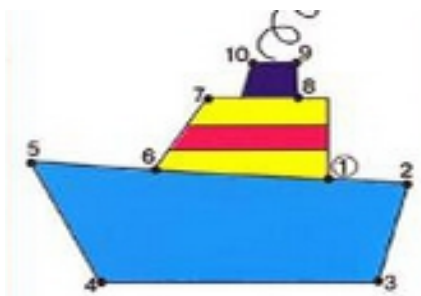
```

Задание

Нарисовать методами программирования:



Дополнительно на оценку



2.2 Оценочные средства для промежуточной аттестации обучающихся

Перечень вопросов для подготовки к экзамену

1. Алгоритм. Базовые алгоритмические конструкции.
2. Исполнители алгоритма. СКИ. Пошаговое выполнение алгоритма.
3. История и классификация языков программирования.
4. Pascal. Алфавит. Идентификаторы, переменные, константы. Общая структура программы. Комментарии.

5. Pascal. Стандартные типы данных. Арифметические и логические выражения. Стандартные функции.
6. Pascal. Простые операторы. Операторы присваивания, составной и простой операторы, операторы ввода-вывода.
7. Pascal. Условный оператор.
8. Pascal. Оператор цикла с условием.
9. Pascal. Оператор цикла с постусловием.
10. Pascal. Оператор цикла с параметром.
11. Pascal. Оператор множественного выбора.
12. Pascal. Строки. Действия над строками.
13. Pascal. Множества. Операции над множествами.
14. Pascal. Одномерные массивы.
15. Pascal. Двумерные массивы. Свойства квадратных матриц.
16. Pascal. Записи. Оператор присоединения.
17. Pascal. Файлы. Функции для работы с файлами.
18. Pascal. Процедуры и функции.
19. Pascal. Работа с графикой.
20. Pascal. Анимация.
21. C. Основные типы данных. Стандартные функции. Математические функции. Структура программы.
22. C. Операторы форматного ввода и вывода. Спецификаторы и модификаторы формата.
23. C. Оператор присваивания. Арифметические выражения. Операторы инкремента и декремента. Сокращенный вид записи арифметических выражений.
24. C. Условные выражения. Оператор условия.
25. C. Оператор множественного выбора.
26. C. Циклы с условием.
27. C. Циклы с постусловием.
28. C. Циклы с параметром.
29. C. Массивы.
30. C. Строки.
31. C. Функции.
32. C. Структуры.
33. C. Файлы.

Перечень практических заданий

- 1) Составить программу для вычисления значения выражения:

$$y = \cos^2 2x + \left| \sin \frac{x}{2} \right|$$

Входные данные – x (вводится с клавиатуры). Выходные данные – y.

- 2) Составьте блок-схему алгоритма и программу, определяющую является ли введенное число двузначным. Результатом работы программы должно быть сообщение 'является' или 'не является'.

3) Составьте блок-схему и программу вычисления значения функции, используя алгоритм полного ветвления:

$$y = \begin{cases} \frac{9x^2 + 5}{3x + 12}, & \text{если } x < -4, \\ 4x^2 - 7, & \text{если } x \geq -4. \end{cases}$$

4) Составьте блок-схему алгоритма и программу вычисления площади и периметра

прямоугольника, если известны его стороны (вводятся с клавиатуры).

5) Составьте блок-схему алгоритма и программу вычисления площади и периметра треугольника, если известны его стороны и высота (вводятся с клавиатуры).

6) Составьте блок-схему алгоритма и программу, определяющую является ли введенное целое число положительным. Результатом работы программы должно быть сообщение 'является' или 'не является'.

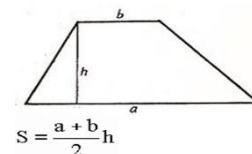
7) Составьте блок-схему алгоритма и программу, определяющую является ли введенное целое число нечетным. Результатом работы программы должно быть сообщение 'является' или 'не является'.

8) Составьте блок-схему алгоритма и программу вычисления количества букв в строке. Строка вводится с клавиатуры.

9) Составьте блок-схему алгоритма и программу вычисления площади ромба по его диагоналям (вводятся с клавиатуры).

$$S = \frac{1}{2} d_1 \cdot d_2$$

10) Составьте блок-схему алгоритма и программу вычисления площади трапеции по ее основаниям и высоте (вводятся с клавиатуры).



11) Составить блок-схему алгоритма и программу для вычисления значения выражения:

$$a = \sqrt{\frac{b+c}{5bc}} - \frac{b^2}{2c};$$

Входные данные – с, b (вводятся с клавиатуры). Выходные данные – a.

12) Составьте блок-схему алгоритма и программу, которая по введенному с клавиатуры целому числу в диапазоне 0 – 9 выводит строку — название соответствующей цифры на русском языке (0 — "ноль", 1 — "один", 2 — "два", ...).

13) Составьте блок-схему алгоритма и программу вычисления среднего арифметического трех целых чисел, введенных с клавиатуры, с точностью до трех знаков после запятой.

14) Составить блок-схему алгоритма и программу для вычисления значения выражения:

$$s = \left| \frac{x + \sqrt{y}}{x^2 + y^2} \right|$$

Входные данные – x, y (вводятся с клавиатуры). Выходные данные – s.

- 15) Составить блок-схему алгоритма и программу, которая по введенному номеру дня недели (вводится с клавиатуры), выдает его название.
- 16) Составьте блок-схему алгоритма и программу вычисления площади круга и длины окружности по введенному радиусу (вводится с клавиатуры).
- 17) Составьте блок-схему алгоритма и программу вычисления площади и периметра квадрата, если известны его стороны (вводятся с клавиатуры).
- 18) Составьте блок-схему алгоритма и программу вывода на экран всех чисел, которые кратны числу 5, в интервале от a до b (значения a и b вводятся, $a < b$).
- 19) Составьте блок-схему алгоритма и программу вычисления произведения целых чисел из промежутка $[-6; 5)$.
- 20) Составьте блок-схему алгоритма и программу вывода на экран всех трехзначных чисел, кратных трем. Решите данную задачу, используя цикл с параметром, цикл с предусловием и цикл с постусловием.
- 21) Составьте блок-схему алгоритма и программу расположения трех чисел в порядке возрастания.
- 22) Составьте блок-схему алгоритма и программу вывода на экран таблицы степеней 2^n , где $0 \leq n \leq 10$.
- 23) Составьте блок-схему алгоритма и программу вывода таблицы значений функции $y = 4x^2 + 5x - 10$ на отрезке $[-9; 9]$ с шагом $h = 3$. Вывод результатов оформите в виде таблицы с границами.
- 24) Составьте блок-схему алгоритма и программу вычисления среднего арифметического целых чисел из отрезка $[-4; 15]$ с точностью до трех знаков после запятой.
- 25) Составьте блок-схему алгоритма и программу вывода таблицы значений функции $y = x^2 - 5x - 2$ на отрезке $[1; 20]$ с шагом $h = 2$. Вывод результатов оформите в виде таблицы с границами.
- 26) Составьте блок-схему алгоритма и программу вычисления значения выражения для данного натурального числа N : $1/1^2 + 1/2^2 + 1/3^2 + \dots + 1/N^2$. Значение N вводится с клавиатуры.
- 27) Составьте блок-схему алгоритма и программу заполнения массива, не используя клавиатуру, числами: 2, 4, 6, 8, 10, 12, 14, 16, 18, 20. Найти сумму элементов полученного массива.
- 28) Составьте блок-схему алгоритма и программу, определяющую, на какую букву начинается второе слово в строке, введенной с клавиатуры.
- 29) Составить программу и блок-схему алгоритма. Сгенерировать массив A из 10 целых чисел, которые берутся из промежутка $[-15, 15]$. Подсчитать количество нечетных элементов массива.
- 30) Составить программу и блок-схему алгоритма. Сгенерировать массив C из 15 целых чисел, которые берутся из промежутка $[-10, 10]$. Подсчитать сумму четных элементов массива.

Комплект примерного билета для промежуточной аттестации:

1. Вопрос № 1

Программирование на языке С. Файлы.

2. Задание № 2

Составить программу для вычисления значения выражения:

$$y = \cos^2 2x + \left| \sin \frac{x}{2} \right|$$

Входные данные – x (вводится с клавиатуры). Выходные данные – y.