

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Гнатюк Сергей Иванович
Должность: Первый проректор
Дата подписания: 07.10.2025 09:37:42
Уникальный программный ключ:
5ede28fe5b714e680817c5c132d4ba793a5b4422

**ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ ФЕДЕРАЛЬНОГО
ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО
УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
«ЛУГАНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
ИМЕНИ К.Е. ВОРОШИЛОВА»**

РАБОЧАЯ ПРОГРАММА
учебной дисциплины
ОПД.04 Основы алгоритмизации и программирования
(наименование учебной дисциплины)

09.02.07 Информационные системы и программирование
(код, наименование профессии/специальности)

Рассмотрено и согласовано цикловой комиссией компьютерных дисциплин.

Протокол № 2 от «02» сентября 2024 г.

Разработана на основе ФГОС СПО РФ и ПООП СПО для специальности 09.02.07 Информационные системы и программирование (утвержден Приказом Минпросвещения России от 9 декабря 2016 года № 1547).

Организация разработчик: Политехнический колледж ЛГАУ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

ОПД.04 Основы алгоритмизации и программирования

1.1. Область применения программы учебной дисциплины

Рабочая программа учебной дисциплины (далее – рабочая программа) является частью освоения программ подготовки специалистов среднего звена (далее ППССЗ) в соответствии с ФГОС СПО РФ и ПООП СПО для специальности 09.02.07 Информационные системы и программирование.

Рабочая программа учебной дисциплины ОПД.04 Основы алгоритмизации и программирования по специальности 09.02.07 Информационные системы и программирование может быть использована на базе среднего (полного общего) образования, в профессиональном обучении и дополнительном профессиональном образовании.

1.2. Цели и задачи учебной дисциплины, требования к результатам освоения учебной дисциплины

Учебная дисциплина ОПД.04 Основы алгоритмизации и программирования относится к общепрофессиональному циклу.

Целью реализации основной образовательной программы среднего общего образования по предмету ОПД.04 Основы алгоритмизации и программирования является освоение содержания предмета Основы алгоритмизации и программирования и достижение обучающимися результатов изучения в соответствии с требованиями, установленными ФГОС СПО РФ и ПООП СПО.

В результате освоения 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, инкапсуляция и полиморфизма, наследования и переопределения.

В результате освоения учебной дисциплины обучающийся должен

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.

2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

ОПД.04 Основы алгоритмизации и программирования

Код ПК, ОК	Умения	Знания
ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5	Разрабатывать алгоритмы для конкретных задач. Использовать программы для графического отображения алгоритмов. Определять сложность работы алгоритмов. Работать в среде программирования. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования. Оформлять код программы в соответствии со стандартом кодирования. Выполнять проверку, отладку кода программы. .	Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции. Эволюцию языков программирования, их классификацию, понятие системы программирования. Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти. Подпрограммы, составление библиотек подпрограмм Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения

3. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Тематический план учебной дисциплины

ОПД.04 Основы алгоритмизации и программирования

Вид учебной работы	Количество часов
1	2
Максимальная учебная нагрузка (всего)	217
Обязательная аудиторная учебная нагрузка (всего)	152
<i>в т. ч.:</i>	
теоретическое обучение	59
практические занятия	91
Самостоятельная работа обучающегося	65
Промежуточная аттестация: дифференцированный зачет	2
ИТОГО	217

3.2. Содержание обучения по учебной дисциплине ОПД.04 Основы алгоритмизации и программирования

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем в часах	Осваиваемые элементы компетенций
Раздел 1. Введение в программирование		30	
Тема 1.1. Языки программирования	Содержание учебного материала	15	
	Развитие языков программирования. Обзор языков программирования. Жизненный цикл программы. Области применения языков программирования. Стандарты языков программирования. Среда проектирования. Компиляторы и интерпретаторы.	4	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	Практическое занятие. Инструктаж по ТБ. Знакомство со средой программирования. Составление программ линейной структуры. Составление программ разветвляющейся структуры. Составление программ циклической структуры	6	
	Самостоятельная работа обучающихся Программа. Программный продукт и его характеристики. Основные этапы решения задач на компьютере.	5	
Тема 1.2. Типы данных	Содержание учебного материала	15	
	Типы данных. Простые типы данных. Производные типы данных.	4	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	Практическое занятие. Инструктаж по ТБ. Работа со строками. Работа с данными типа множество. Файлы последовательного доступа.	6	
	Самостоятельная работа обучающихся Структурированные типы данных. Типизированные файлы. Нетипизированные файлы.	5	
Раздел 2. Содержание учебного материала		15	
Тема 2.1. Операторы языка программирования	Содержание учебного материала	15	
	Операции и выражения. Правила формирования и вычисления выражений. Цикл с постусловием. Цикл с предусловием. Цикл с параметром. Вложенные циклы. Массивы. Двумерные массивы. Строки. Стандартные процедуры и функции для работы со строками. Структурированный тип данных – множество. Операции над множествами.	4	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10 ПК 1.1- ПК 1.5
	Практическое занятие. Инструктаж по ТБ. Структура программы. Ввод и вывод данных. Оператор присваивания.	6	

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем в часах	Осваиваемые элементы компетенций
	Самостоятельная работа обучающихся Составной оператор. Условный оператор. Оператор выбора.	5	ПК 2.4, 2.5
Раздел 3. Содержание учебного материала		45	
Тема 3.1. Процедуры и функции	Содержание учебного материала	15	ОК 1
	Общие сведения о подпрограммах. Определение и вызов подпрограмм	4	ОК 2
	Практическое занятие. Инструктаж по ТБ. Область видимости и время жизни переменных. Механизм передачи параметров. Организация функций.	6	ОК 4 ОК 5 ОК 9 ОК 10
	Самостоятельная работа обучающихся Рекурсия. Программирование рекурсивных алгоритмов.	5	ПК 1.1- ПК 1.5 ПК 2.4, 2.5
Тема 3.2. Структуризация в программировании	Содержание учебного материала	15	
	Основы структурного программирования. Методы структурного программирования.	4	ОК 1
	Практическое занятие. Инструктаж по ТБ. Разработка функциональной схемы работы приложения. Разработка оконного приложения с несколькими формами. Разработка игрового приложения.	6	ОК 2 ОК 4 ОК 5 ОК 9 ОК 10
	Самостоятельная работа обучающихся Программирование модуля. Создание библиотеки подпрограмм.	5	ПК 1.1- ПК 1.5 ПК 2.4, 2.5
Тема 3.3. Модульное программирование	Содержание учебного материала	15	
	Модульное программирование. Понятие модуля.	4	ОК 1
	Практическое занятие. Инструктаж по ТБ. Программирование модуля. Разработка оконного приложения с несколькими формами. Разработка игрового приложения.	6	ОК 2 ОК 4 ОК 5 ОК 9 ОК 10
	Самостоятельная работа обучающихся Структура модуля. Компиляция и компоновка программы. Стандартные модули.	5	ПК 1.1- ПК 1.5 ПК 2.4, 2.5
Раздел 4. Основные конструкции языков программирования		16	
Тема 4.1. Указатели.	Содержание учебного материала	16	ОК 1
	Указатели. Описание указателей. Задача о стеке.	4	ОК 2
	Практическое занятие. Инструктаж по ТБ. Использование указателей для организации связанных списков. Основные понятия и применение динамически распределяемой памяти. Создание и удаление динамических	7	ОК 4 ОК 5 ОК 9

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем в часах	Осваиваемые элементы компетенций
	переменных.		ОК 10
	Самостоятельная работа обучающихся Структуры данных на основе указателей. Изучение интегрированной среды разработчика.	5	ПК 1.1- ПК 1.5 ПК 2.4, 2.5
Раздел 5. Содержание учебного материала		109	
Тема 5.1. Основные принципы объектно-ориентированного программирования (ООП)	Содержание учебного материала	17	ОК 1
	История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс.	4	ОК 2 ОК 4
	Практическое занятие. Инструктаж по ТБ. Основные принципы ООП: инкапсуляция, наследование, полиморфизм. Классы объектов. Компоненты и их свойства. Событийно-управляемая модель программирования.	8	ОК 5 ОК 9 ОК 10
	Самостоятельная работа обучающихся Компонентно-ориентированный подход. Создание процедур на основе событий.	5	ПК 1.1- ПК 1.5 ПК 2.4, 2.5
Тема 5.2. Интегрированная среда разработчика.	Содержание учебного материала	17	
	Требования к аппаратным и программным средствам интегрированной среды разработчика. Интерфейс среды разработчика: характеристика, основные окна, инструменты, объекты. Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта.	4	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10
	Практическое занятие. Инструктаж по ТБ. Форма и размещение на ней управляющих элементов. Панель компонентов и их свойства. Окно кода проекта. Состав и характеристика проекта. Окно кода проекта.	8	ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	Самостоятельная работа обучающихся Выполнение проекта. Настройка среды и параметров проекта. Панель компонентов и их свойства. Настройка среды и параметров проекта.	5	
Тема 5.3. Визуальное событийно-управляемое программирование	Содержание учебного материала	18	
	Основные компоненты (элементы управления) интегрированной среды разработки, их состав и назначение. Дополнительные элементы управления.	5	ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10
	Практическое занятие. Инструктаж по ТБ. Виды свойств. Синтаксис определения свойств. Назначения свойств и их влияние на результат. Управление объектом через свойства.	8	ПК 1.1- ПК 1.5 ПК 2.4, 2.5
	Самостоятельная работа обучающихся Свойства компонентов. События компонентов (элементов управления), их сущность и назначение. Создание процедур на основе событий.	5	

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем в часах	Осваиваемые элементы компетенций
Тема 5.4. Разработка оконного приложения	Содержание учебного материала	19	
	Разработка функционального интерфейса приложения. Создание интерфейса приложения. Разработка функциональной схемы работы приложения.	6	ОК 1 ОК 2
	Практическое занятие. Инструктаж по ТБ. Разработка игрового приложения. Разработка функциональной схемы работы приложения. Разработка оконного приложения с несколькими формами. Разработка игрового приложения.	8	ОК 4 ОК 5 ОК 9 ОК 10
	Самостоятельная работа обучающихся Создание проекта с использованием кнопочных компонентов. Создание проекта с использованием компонентов стандартных диалогов и системы меню.	5	ПК 1.1- ПК 1.5 ПК 2.4, 2.5
Тема 5.5. Этапы разработки приложений	Содержание учебного материала	19	
	Разработка приложения. Проектирование объектно-ориентированного приложения. Создание интерфейса пользователя.	6	ОК 1 ОК 2
	Практическое занятие. Инструктаж по ТБ. Тестирование, отладка приложения. Этапы разработки приложений. Разработка интерфейса приложения. Решение задач.	8	ОК 4 ОК 5 ОК 9 ОК 10
	Самостоятельная работа обучающихся Создание процедур обработки событий. Компиляция и запуск приложения.	5	ПК 1.1- ПК 1.5 ПК 2.4, 2.5
Тема 5.6. Иерархия классов.	Содержание учебного материала	19	
	Классы ООП: виды, назначение, свойства, методы, события. Перегрузка методов. Тестирование и отладка приложения.	6	ОК 1 ОК 2
	Практическое занятие. Инструктаж по ТБ. Объявления класса. Создание наследованного класса. Программирование приложений. Перегрузка методов.	8	ОК 4 ОК 5 ОК 9 ОК 10
	Самостоятельная работа обучающихся Комбинированный тип данных – запись. Файлы последовательного доступа. Файлы прямого доступа.	5	ПК 1.1- ПК 1.5 ПК 2.4, 2.5

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем в часах	Осваиваемые элементы компетенций
	Всего:	217	
	из них практических занятий	91	
	лекций	59	
	самостоятельная работа	65	
	зачет	2	
	экзамен	-	

4. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

4.1. Требования к материально-техническому обеспечению

Реализация программы дисциплины требует наличия лаборатории «Программирования и баз данных».

Эффективность преподавания курса Основ алгоритмизации и программирования зависит от наличия соответствующего материально-технического оснащения. Это объясняется особенностями курса, в первую очередь его многопрофильностью и практической направленностью.

Оборудование учебного кабинета:

- рабочее место преподавателя и рабочие места по количеству обучающихся;
- аудиовизуальные средства – схемы и рисунки к занятиям в виде слайдов и электронных презентаций;
- - операционная система;
- файловый менеджер (в составе операционной системы или др.).
- антивирусная программа;
- программа-архиватор;
- программа – переводчик;
- интегрированное офисное приложение, включающее текстовый редактор, программу разработки презентаций и электронные таблицы;
- системы автоматизированного проектирования;
- простая система управления базами данных;
- мультимедиа проигрыватель (входит в состав операционных систем или др.);
- браузер (входит в состав операционных систем или др.);
- электронные средства образовательного назначения;
- программное обеспечение локальных сетей.

Приводится перечень средств обучения, включая тренажеры, модели, макеты, оборудование, технические средства, в т. ч. аудиовизуальные, компьютерные и телекоммуникационные и т. п. (количество не указывается)

Требования к квалификации педагогических кадров, осуществляющих реализацию ППССЗ по специальности, должны обеспечиваться педагогическими кадрами, имеющими среднее профессиональное, высшее образование, соответствующее профилю преподаваемой учебной дисциплины. Опыт деятельности в организациях соответствующей профессиональной сферы является обязательным для преподавателей, отвечающих за освоение обучающимися профессионального учебного цикла.

Преподаватели получают дополнительное профессиональное образование по программам повышения квалификации, в том числе в форме стажировки в профильных организациях не реже одного раза в 5 лет.

4.2. Информационное обеспечение обучения. Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Основные печатные издания

1. Основы алгоритмизации и программирования / И.Г. Семакин, А.П. Шестаков. – Москва: Академия, 2021. – 304 с.

Основные электронные издания

2. Трофимов, В. В. Основы алгоритмизации и программирования: учебник для среднего профессионального образования / В. В. Трофимов, Т. А. Павловская; под редакцией В. В. Трофимова. — Москва: Издательство Юрайт, 2021. — 137 с. — (Профессиональное образование). — ISBN 978-5-534-07321-8. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/473347> (дата обращения: 13.12.2021).

Дополнительные источники

3. Кудрина, Е. В. Основы алгоритмизации и программирования на языке С#: учебное пособие для среднего профессионального образования / Е. В. Кудрина, М. В. Огнева. — Москва: Издательство Юрайт, 2021. — 322 с. — (Профессиональное образование). — ISBN 978-5-534-10772-2. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/475228> (дата обращения: 13.12.2021).

4. Шандриков, А. С. Информационные технологии: учебное пособие / А. С. Шандриков. - 3-е изд., стер. - Минск: РИПО, 2019. - 443 с. - ISBN 978-985-503-887-1. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1088261> (дата обращения: 13.12.2021). — Режим доступа: по подписке.

5. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Контроль и оценка результатов освоения учебной дисциплины осуществляется преподавателем при проведении лабораторных работ, практических занятий, тестирования, а также выполнения обучающимися индивидуальных заданий, проектов, исследований, практических занятий, тестирования, а также выполнения обучающимися индивидуальных заданий, проектов, исследований.

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
1	2
Умения	
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.	Оценка результатов выполнения заданий, приемов, упражнений. Оценка выполненных самостоятельных работ.
Знания:	
Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции. Эволюцию языков программирования, их классификацию, понятие системы программирования. Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти. Подпрограммы, составление библиотек подпрограмм Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения.	Контрольная работа. Самостоятельная работа. Защита реферата. Выполнение проекта. Наблюдение за выполнением практического задания (деятельностью студента). Оценка выполнения практического задания (работы). Подготовка и выступление с докладом, сообщением, презентацией

ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ ФЕДЕРАЛЬНОГО
ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО
УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
«ЛУГАНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
ИМЕНИ К.Е. ВОРОШИЛОВА»

КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА
учебной дисциплины

ОПД.04 Основы алгоритмизации и программирования
(наименование учебной дисциплины)

09.02.07 Информационные системы и программирование
(код, наименование профессии/специальности)

2024

**Контрольно-оценочные средства
для выполнения промежуточной аттестации в форме
дифференцированного зачета**

Базовые конструкции языка программирования

Теоретические вопросы:

1. Оператор цикла с постусловием Do While.

Какими будут значения переменных *s* и *i* после выполнения следующих операторов: `int s=20;`

`int i= 0; while (i<=10) {s+= 5; i+= 1;}`

2. Оператор цикла с предусловием While.

Определить значения переменных *a* и *b* после выполнения следующих операторов: `a= 1; b= 2;`

`while (b<20) {a= a + 2; b= b +2;}`

3. Оператор цикла с параметром For.

Определить значение переменной *s* после выполнения следующих операторов:

`int s=0; int`

`n=5; for (int i=2; i< n; i++) s+= 100 / i;`

Определить значение переменной *s* после выполнения следующих операторов:

`int s=0; int`

`n=5; for (int i=2; i<= n; i++) s+= 100 / i;`

4. Оператор выбора switch.

Определить значение переменной *p* после выполнения следующего фрагмента программы: `int`

`m= 13; int n= 21; n= 2*m – n;`

`if (m<=n) t p = m + n ; else p= 4 – m*n;`

5. Операторы switch и If, назначение и отличие.

Определить значение переменной *p* после выполнения следующего фрагмента программы: `int`

`m = 13; int n= 21; n= 2*m – n;`

`if (m<=n) p= m + n; else p= 4 – m*n;`

6. Операторы цикла For , While и Do While, блок-схемы.

Какими будут значения переменных *s* после выполнения следующих операторов: `int s=2; int`

`i= 1; while (i<10) {s+= i; i+= 1;}`

7. Условный оператор IF (IF-....., IF-.....-ELSE).

Определить значение переменной *p* после выполнения следующего фрагмента программы: `int`

`m= 13; int n= 21; n= 2*m – n;`

`if (m<=n ) p= m + n; else p= 4 – m*n;`

8. Операторы цикла While и Do While, их отличие.

Какими будут значения переменных *a* и *b* после выполнения следующих операторов: `int a= 1;`

`int b= 2; while (b<20 ) { a+= 2; b+= 2;}`

9. Оператор многовероятностного выбора. Форма записи и пример.

Определить значение переменной p после выполнения следующего фрагмента программы: `int`

`m= 13; int n= 21; n= 2*m – n;`

`if (m<=n) p= m + n; else p= 4 – m*n`

10. Ввод одномерных массивов случайным образом. Пример.

Определить значение переменной s после выполнения следующих операторов:

`int s=0; int`

`n=5; for (int i=2 ; i< n; i++) s+= 100 / i;`

11. Алгоритм сортировки массива.

Какими будут значения переменных s и i после выполнения следующих операторов: `int s=20;`

`int i= 0; while (i<=10) { s+= 5; i+= 1;}`

12. Ввод двумерных массивов.

Какими будут значения переменных a и b после выполнения следующих операторов:

`int a= 1; int b= 2; while (b<20) { a+= 2; b+=2;}`

30 Вложенные условные операторы. Пример алгоритма.

Определить значение переменной p после выполнения следующего фрагмента программы:

`int m= 13; int n= 21; n= 2*m – n; if (m<=n ) p= m + n; else p= 4 – m*n;`

13. Вывод двумерных массивов.

Определить значение переменной s после выполнения следующих операторов:

`int s =0; int n`

`=4; for (int i=2; i<= n; i++) s+= 10 / i;`

14. Алгоритм нахождения максимального (минимального) элемента в массиве (одномерный массив, матрица).

15. Алгоритм нахождения порядкового номера максимального (минимального) элемента в массиве (одномерный массив, матрица).

16. Функции деления с остатком. Определить значения переменных a и b после выполнения

последовательности действий:

`a= 4*5 / 3 %2; b= 4*5 /(3 %2);`

***Практические задания**

1 Разработать и произвести отладку программы: Найти сумму бесконечного ряда. Суммировать до тех пор, пока сумма не станет больше заданного $p>0$. Вывести эти числа.

2 Разработать и произвести отладку программы для определения $N!-M!$. $N! = 1*2*3*4*.....*n$

3 Разработать и произвести отладку программы: Вычислить сумму квадратов всех целых чисел, пока сумма квадратов меньше заданного числа A . Вывести эти числа.

- 4 Разработать и произвести отладку программы: Произведение первых четных чисел равно P , сколько сомножителей взято.
- 5 Разработать и произвести отладку программы: Определить все двузначные числа, сумма квадратов цифр которых кратны числу 15
- 6 Разработать и произвести отладку программы: Даны два одномерных массива одинаковой длины. Получить третий массив такой же размерности, каждый элемент которого равен сумме соответствующих элементов данных массивов.
- 7 Разработать и произвести отладку программы: ан одномерный массив чисел. Определите сумму элементов, принадлежащих промежутку от A до B (A и B водить с клавиатуры).
- 8 Разработать и произвести отладку программы определения количества элементов массива, больших среднего арифметического всех его элементов.
- 9 Разработать и произвести отладку программы: Дан массив P целых чисел из n элементов, заполненный случайным образом числами из промежутка $[-10,10]$. Из элементов массива P сформировать массив M той же размерности по правилу: если номер четный, то $M_i = i * P_i$, если нечетный, то $M_i = -P_i$. Исходный и скорректированный массив вывести на экран.
- 10 Разработать и произвести отладку программы: ан массив P целых чисел из n элементов, заполненный случайным образом числами из промежутка $[-30,30]$. Из элементов массива P сформировать массив M из четных чисел. Исходный и скорректированный массивы вывести на экран.
- 11 Разработать и произвести отладку программы: ан массив P целых чисел из n элементов, заполненный случайным образом числами из промежутка $[-10,10]$. Из элементов массива P сформировать массив M той же размерности по возрастанию. Исходный и скорректированный массивы вывести на экран.
- 12 Разработать и произвести отладку программы, печатающей все делители целого числа в порядке убывания.
- 13 Разработать и произвести отладку программы, печатающей все делители целого числа в порядке возрастания

Раздел 3 Структурное и модульное программирование

Теоретические вопросы:

- 35 Назначение процедуры. Описание и вызов процедуры. Какими будут значения переменных s и i после выполнения следующих операторов: `int s=20; int i= 0; while (i<=10) { s+= 5; i+= 1;}`
- 36 Назначение функции. Описание и вызов функции. Отличие процедуры от функции. Чему равны значения переменных a и b после выполнения последовательности действий: `int a= 4*5 /3 % 2; b= 4*5/ (3 %2)`
- 37 Подпрограммы (структура подпрограммы, схема алгоритма). Чему равны значения переменных a и b после выполнения последовательности действий: `int a= 4*5 / 3 % 2; int b= 4*5 /(3 %2);`

Раздел 4 Основные конструкции языков программирования

Теоретические вопросы:

- 38 Файлы (общие сведения, характеристики, описание файловых переменных).

Определить значение переменной *s* после выполнения следующих операторов:

```
int s=0; int n=5; for (int i=2 ; i< n; i++) s+= 100 / i;
```

39 Основные процедуры работы с файлами.

Какими будут значения переменных *a* и *b* после выполнения следующих операторов:

```
int a= 1; int b= 2; while (b<20) { a+= 2; b+= 2;}
```

40 Процедуры и функции работы с текстовыми файлами.

Какими будут значения переменных *a* и *b* после выполнения следующих операторов:

```
int a= 1; int b= 1; while (a<=3) { a+= 1; b+= 1;}
```

41 Основные команды работы с графикой.

Чему равны значения переменных *a* и *b* после выполнения последовательности действий:

```
int a= 15 / (16 % 7); int b=34 % a*5 – 29 % 5*2;
```

42 Текстовые файлы (назначение, описание в программе).

Какими будут значения переменных *s* и *i* после выполнения следующих операторов:

```
int s=20; int i= 0; while (i<=10) { s+= 5; i+= 1;}
```

***Практические задания:**

14 Разработать и произвести отладку программы: Пользователь угадывает число задуманное компьютером, при помощи подсказок больше или меньше, компьютер выдаёт количество шагов, за которые пользователь угадал число.

15 Разработать и произвести отладку программы: Произведение *N* первых нечетных чисел равно *p*. Сколько сомножителей взято?

16 Дана последовательность из *N* целых чисел. Определить произведение максимального и минимального элементов этой последовательности.

17 Составьте программу- "перевертыш" (так называются слова, читающиеся одинаково слева направо и справа налево, например: ПОТОП, КАЗАК).

18 Дан текст со скобками, удалить текст в скобках вместе со скобками и вывести его отдельно.

19 Вывести все двухзначные числа, сумма цифр которых равна *N*.

20 В тексте имеется одна точка с запятой «;». Подсчитать количество символов до точки с запятой и после нее.

21 В тексте имеются запятые. Подсчитать количество запятых.