

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
(МИНОБРНАУКИ РОССИИ)

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ДОНБАССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ДонГТУ»)

Факультет информационных технологий и автоматизации
производственных процессов

Кафедра интеллектуальных систем и информационной
безопасности



И.о. проректора по учебной работе
Д.В. Мулов

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Основы программирования

(наименование дисциплины)

02.03.01 Математика и компьютерные науки

(код, наименование направления/специальности)

09.03.01 Информатика и вычислительная техника

(код, наименование направления/специальности)

10.05.03 Информационная безопасность автоматизированных систем

(код, наименование направления/специальности)

38.03.05 Бизнес-информатика

(код, наименование направления/специальности)

Квалификация бакалавр, специалист по защите информации
(бакалавр/специалист)

Форма обучения очная
(очная, очно-заочная, заочная)

Алчевск, 2024

1 Цели и задачи изучения дисциплины

Цели дисциплины. Целью изучения дисциплины «Основы программирования» является выработка у студентов передовых научно-технических воззрений, ориентации их на мировой уровень производительности труда, подготовке специалистов, которые должны обеспечить разработку качественных компьютерных программ, снижение материальных затрат, сокращение сроков проектирования.

Задачи изучения дисциплины:

- приобретение студентами знаний, умений и навыков в области алгоритмизации, программирования, основным этапам решения задач на ЭВМ;
- ознакомление с основами программирования на языке С;
- приобретение навыков использования полученных знаний в прикладных исследованиях, проектировании и эксплуатации информационных систем и технологий;
- умение создавать программные приложения для проведения инженерных расчетов, обработки числовой, текстовой и графической информации.

2 Место дисциплины в структуре ОПОП ВО

Логико-структурный анализ дисциплины – курс входит в обязательную часть БЛОКА 1 «Дисциплины (модули)» по направлениям 02.03.01 Математика и компьютерные науки, 09.03.01 Информатика и вычислительная техника, 38.03.05 Бизнес-информатика и специальности 10.05.03 Информационная безопасность автоматизированных систем.

Дисциплина реализуется кафедрой интеллектуальных систем и информационной безопасности.

Основывается на базе дисциплин: «Высшая математика».

Является основой для изучения следующих дисциплин: «Объектно-ориентированное программирование», «Вычислительная математика», «Базы данных», «Безопасность баз данных», «Учебно-лабораторный практикум».

Для изучения дисциплины необходимы компетенции, сформированные у студента для решения профессиональных задач деятельности, связанных с разработкой программного обеспечения.

Курс является фундаментом для ориентации студентов в сфере разработки программного обеспечения информационных систем.

Общая трудоемкость освоения дисциплины составляет 12 зачетных единицы, 432 ак.ч. Программой дисциплины предусмотрены лекционные (72 ак.ч.), лабораторные (72 ак.ч.), практические (54 ак.ч.) занятия и самостоятельная работа студента (234 ак.ч.).

Дисциплина изучается на 1,2 курсе в 1-3 семестрах. Форма промежуточной аттестации – экзамен и дифференцированный зачет.

3 Перечень результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОПОП ВО

Процесс изучения дисциплины «Основы программирования» направлен на формирование компетенций, представленных в таблице 1.

Таблица 1 – Компетенции, обязательные к освоению

Код	Наименование специальности, направления подготовки	Компетенция (код, содержание)	Индикатор (код, наименование)
02.03.01	Математика и компьютерные науки	ОПК-6 Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК-6.3. Имеет практические навыки разработки ПО
09.03.01	Информатика и вычислительная техника	ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-8.3. Составляет алгоритмы, пишет и отлаживает коды на заданном языке программирования, тестирует работоспособность программ, интегрирует программные модули
10.05.03	Информационная безопасность автоматизированных систем	ОПК-7 Способен создавать программы на языках общего назначения, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ	ОПК-7.1. Создает программы на языках общего назначения
38.03.05	Бизнес-информатика	ОПК-3 Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации	ОПК-3.3. Разрабатывает алгоритмы и приложения на бизнес-ориентированных языках программирования

4 Объём и виды занятий по дисциплине

Общая трудоёмкость учебной дисциплины составляет 12 зачётных единиц, 432 ак.ч.

Самостоятельная работа студента (СРС) включает проработку материалов лекций, подготовку к практическим занятиям, текущему контролю, выполнение индивидуального задания, выполнение курсовой работы, самостоятельное изучение материала и подготовку к экзамену.

При организации внеаудиторной самостоятельной работы по данной дисциплине используются формы и распределение бюджета времени на СРС для очной формы обучения в соответствии с таблицей 2.

Таблица 2 – Распределение бюджета времени на СРС

Вид учебной работы	Всего ак.ч.	Ак.ч. по семестрам		
		1	2	3
Аудиторная работа, в том числе:	162	90	90	18
Лекции (Л)	72	36	36	-
Практические занятия (ПЗ)	36	18	18	-
Лабораторные работы (ЛР)	72	36	36	-
Курсовая работа	18	-	-	18
Самостоятельная работа студентов (СРС), в том числе:	234	90	126	18
Подготовка к лекциям	36	9	9	-
Подготовка к лабораторным работам	36	27	27	-
Подготовка к практическим занятиям / семинарам	18	9	9	-
Выполнение курсовой работы / проекта	10	-	-	10
Расчетно-графическая работа (РГР)	-	-	-	-
Реферат (индивидуальное задание)	12	-	12	-
Домашнее задание	6	-	15	-
Подготовка к контрольным работам	9	9	-	-
Подготовка к коллоквиуму	-	-	-	-
Аналитический информационный поиск	18	9	18	-
Работа в библиотеке	14	-	10	4
Подготовка к экзамену (диф.зачету)	75	27	26	4
Промежуточная аттестация – экзамен (Э), диф.зачет (ДЗ)	Э, ДЗ	Э	Э	ДЗ
Общая трудоемкость дисциплины				
ак.ч.	432	180	216	36
з.е.	12	5	6	1

5 Содержание дисциплины

С целью освоения компетенции, приведенной в п.3 дисциплина разбита на 9 тем:

- тема 1 (Начальные сведения о языках программирования);
- тема 2 (Язык Си – базовый синтаксис);
- тема 3 (Массивы, указатели и строки);
- тема 4 (Функции);
- тема 5 (Поразрядные логические операции и операции сдвига);
- тема 6 (Типы данных, определяемые пользователем);
- тема 7 (Файлы текстовые и бинарные);
- тема 8 (Стадии препроцессорной обработки);
- тема 9 (Основные понятия объектно-ориентированного программирования).

Виды занятий по дисциплине и распределение аудиторных часов для очной формы приведены в таблице 3.

В таблице 4 приведено распределение видов занятий и распределение аудиторных часов для выполнения курсовой работы.

Таблица 3 – Виды занятий по дисциплине и распределение аудиторных часов (очная форма обучения)

№ п/п	Наименование темы (раздела) дисциплины	Содержание лекционных занятий	Трудоемкость в ак.ч.	Темы практических занятий	Трудоемкость в ак.ч.	Тема лабораторных занятий	Трудоемкость в ак.ч.
1	2	3	4	5	6	7	8
1	Начальные сведения о языках программирования	Характеристики и свойства языков программирования. Понятие среды разработки. Алгоритмы, их свойства, виды и способы записи. Программные средства представления алгоритмов. Основные этапы решения задачи на ЭВМ. Принципы структурного программирования.	2	Постановка задач, алгоритмы	2	Алгоритмы. Виды и способы представления. Знакомство со средой разработки	4
2	Язык Си – базовый синтаксис	Основные программные объекты. Структура программы. Стандартные типы данных. Операторы описания. Стандартный ввод и вывод. Ввод вывод из текстовых файлов.	4	Работа с переменными и константами, ввод-вывод	2	Переменные и константы. Ввод-вывод	6
		Вычисления, математические функции и библиотека math.h	4	Программирование алгоритмов линейной структуры	4	Программирование алгоритмов линейной структуры	6
		Условные операторы как инструмент языка для управления процессом обработки информации. Две формы условного оператора. Оператор безусловного перехода.	4	Разветвления и циклы	2	Программирование алгоритмов разветвляющейся структуры и циклов	8

Продолжение таблицы 3

1	2	3	4	5	6	7	8
2	Язык Си – базовый синтаксис	Три формы оператора цикла. Примеры применения операторов цикла в программах.	4	Разветвления и циклы	2	–	–
3	Массивы, указатели и строки	Одномерные массивы. Базовые алгоритмы их обработки. Многомерные массивы. Указатель. Адресная арифметика. Связь указателей с массивами. Доступ к данным через указатели. Строковые константы, стандартные функции для работы со строками. Массивы строк, массив указателей.	6	Обработка массивов Указатели и строки	4	Обработка массивов	6
4	Функции	Определение, описание и вызов функции. Способы передачи параметров в функцию. Передача массивов как параметров. Классы памяти. Рекурсивные функции. Понятие динамической памяти и использование динамических переменных. Указатели на функции. Передача функций функциям в качестве параметров	4	Функции Работа с памятью программ Указатели на функции	2	Функции	6

Окончание таблицы 3

1	2	3	4	5	6	7	8
5	Поразрядные логические операции и операции сдвига	Побитовые операции и области их применения	4	Побитовые операции	4	Побитовые операции	8
6	Типы данных, определяемые пользователем	Структуры, объединения, битовые поля. Переименование типов. Перечисления. Использование битовых полей	8	Структуры	6	Использование структур	10
7	Файлы текстовые и бинарные.	Файлы последовательного и прямого доступа. Режим работы в файлах прямого доступа. Функции для работы с файлами и директориями. Файловый ввод-вывод.	12	Файлы	4	Обработка файлов	10
8	Стадии препроцессорной обработки.	Стадии препроцессорной обработки. Директивы. Макроподстановки. Условная компиляция	6	Макросы	4	Преобразование типов	8
9	Основные понятия объектно-ориентированного программирования (ООП).	Отличия ООП от процедурного программирования. Принципы ООП. ООП в языках программирования.	6	–	–	-	-
Всего аудиторных часов			72	36		72	

Таблица 4 – Виды занятий по дисциплине и распределение аудиторных часов по курсовой работе (очная форма обучения)

№ п/п	Тема занятия	Трудоемкость в ак.ч.
1	Описание задачи, разработка содержательной блок-схемы алгоритма программы	2
2	Определение и описание переменных, констант и подпрограмм (функций)	2
3	Разработка интерфейса программы	2
4	Разработка программного кода основного алгоритма	4
5	Разработка программного кода обработчиков событий	4
6	Выполнение тестового примера	2
7	Оформление пояснительной записки	2
Всего аудиторных часов		18

6 Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации по итогам освоения дисциплины

6.1 Критерии оценивания

В соответствии с Положением о кредитно-модульной системе организации образовательного процесса ФГБОУ ВО «ДонГТУ» (https://www.dstu.education/images/structure/license_certificate/polog_kred_modul.pdf) при оценивании сформированности компетенций по дисциплине используется 100-балльная шкала.

Перечень работ по дисциплине и способы оценивания знаний приведены в таблице 5.

Таблица 5 – Перечень работ по дисциплине и способы оценивания знаний

Вид учебной работы	Способ оценивания	Количество баллов
Выполнение лабораторных работ	Предоставление отчетов	42 - 70
Выполнение контрольных работ	Более 50% правильных ответов	12-20
Выполнение домашнего задания	Предоставление материалов домашнего задания	6 - 10
Итого	-	60-100
Выполнение курсовой работы	Предоставление материалов согласно календарного плана	36 – 60
Защита курсовой работы	Защита курсовой работы	24 – 40
Итого по курсовой работе	–	60 - 100

Экзамен проставляется автоматически, если студент набрал в течении семестра не менее 60 баллов и отчитался за каждую контрольную точку. Минимальное количество баллов по каждому из видов текущей работы составляет 60% от максимального.

Экзамен по дисциплине «Основы программирования» проводится по результатам работы в семестре. В случае, если полученная в семестре сумма баллов не устраивает студента, во время зачетной сессии студент имеет право повысить итоговую оценку либо в форме устного собеседования по приведенным ниже вопросам (п.п. 6.5), либо в результате тестирования.

Шкала оценивания знаний при проведении промежуточной аттестации приведена в таблице 6.

Таблица 6 – Шкала оценивания знаний

Сумма баллов за все виды учебной деятельности	Оценка по национальной шкале зачёт/экзамен
0-59	Не зачтено/неудовлетворительно
60-73	Зачтено/удовлетворительно
74-89	Зачтено/хорошо
90-100	Зачтено/отлично

6.2 Домашние задания

Задание 1. Создать структуру согласно своего варианта задания к лабораторной работе № 2 (2 семестр). Написать программу, обеспечивающую автоматическую генерацию 10 переменных типа структур. Диапазон граничных значений вводить с клавиатуры.

Задание 2. Написать программу, которая заносит в файл и считывает из него структуры из задания 1.

Задание 3. Обеспечить вывод из файла, сформированного в результате выполнения задания 2, в виде таблицы на экран с окаймлением границ ячеек. Для окаймления ячеек использовать символы псевдографики в кодировке UTF-8.

6.3 Темы для рефератов (презентаций) – индивидуальное задание

- 1) Типы данных Си, определяемые пользователем.
- 2) Работа с текстовыми файлами в Си.
- 3) Работа с файлами, содержащими типы данных пользователя в Си.
- 4) Вывод данных в Си в виде таблиц.
- 5) Как функция Си работает с локальными и глобальными переменными.
- 6) Передача параметров в функции Си различными способами.
- 7) Динамические массивы в Си.
- 8) Рекурсивные функции в Си.
- 9) Многопоточный ввод-вывод в Си.
- 10) Возможности компиляторов Си различных производителей.
- 11) Компиляторы Си, поставляемые по программе OpenSource.
- 12) Массивы строк в Си: поиск и замена символов и фрагментов строк.
- 13) Рекурсивная обработка строк в Си.
- 14) Семейство Си-подобных языков программирования.
- 15) Особенности использования языка Си в Windows и Linux.
- 16) Вывод таблиц на экран в программах Си с использованием псевдографики.
- 17) Использование моделей памяти в программах Си.

- 18) Свободные массивы в языке Си.
- 19) Управление версиями программы на Си для Windows/Linux с помощью директив условной компиляции.
- 20) Сортировка данных в файлах Си.

6.4 Оценочные средства для самостоятельной работы и текущего контроля успеваемости

Контрольная работа № 1 (1 семестр). Выполняется в виде решения тестовых задач:

1) Для переменных заданы максимальные, минимальные и начальные значения. Выберите для них подходящие типы данных и запишите объявления для программы.

2) Какие значения будут иметь переменные после выполнения приведенного ниже фрагмента программы? Покажите ход решения.

Контрольная работа № 2 (1 семестр). Выполняется в виде решения тестовых задач вида:

1) Дана программа с ошибками. Найти и исправить ошибки.

2) Дана программа с использованием операторов цикла и условия. Переписать её с использованием других операторов, использующих аналогичные функции.

Контрольная работа № 3 (1 семестр). Выполняется в виде решения тестовых задач:

Дан массив размерностью $n \times m$. Выполнить выборку элементов k -й строки (j -го столбца), рассчитать среднее значение элементов массива, выделить в отдельный вектор элементы массива согласно признакам.

Контрольная работа итоговая (1 семестр). Выполняется в виде решения тестовых задач, собранных из контрольных работ №№ 1 – 3.

Также используются контрольные вопросы к защите лабораторных работ <http://library.dstu.education/download.php?rec=132257>.

Во втором семестре контрольные работы не проводятся. Оценка знаний производится на лабораторных занятиях. Используются контрольные вопросы к защите лабораторных работ <http://library.dstu.education/download.php?rec=103329>

6.5 Вопросы для подготовки к экзамену

I семестр

- 1) Какова классификация языков программирования?
- 2) Какие известны этапы и особенности решения задач с помощью ЭВМ?
- 3) Какие известны алгоритмы, их свойства, способы описания алгоритмов?
- 4) Что такое блок-схемы алгоритмов?
- 5) Какова структура программы?

- 6) Что такое структурное программирование? Его основные принципы? Как формулируется теорема о структурировании?
- 7) Какова классификация языков программирования?
- 8) Какие свойства языков программирования, характеризуют качество программ?
- 9) Что такое алфавит, синтаксис и семантика языка программирования?
- 10) Какие есть способы описания синтаксиса?
- 11) Что такое идентификатор в языках программирования? Как определить идентификатор с помощью синтаксической диаграммы и языка БНФ?
- 12) Каковы основные характеристики языка Си?
- 13) Что такое структура программы в языке Си?
- 14) Что такое тип данных?
- 15) Каковы основные типы данных в языке Си? Что такое квалификаторы и спецификаторы?
- 16) Как задаются константы в языке Си?
- 17) Что такое переменные в языке Си, их описание и инициализация?
- 18) Что такое стандартный ввод данных в языке Си?
- 19) Что такое стандартный вывод данных в языке Си?
- 20) Какие есть арифметические операции в языке Си?
- 21) Какие есть математические функции библиотеки `math.h`?
- 22) Какие есть операции присваивания и корректирующие операции присваивания в языке Си?
- 23) Что такое явные и неявные преобразования типов в языке Си?
- 24) Какие есть операции отношения в языке Си?
- 25) Какие есть логические операции в языке Си? Каковы правила вычислений логических выражений?
- 26) Какие есть условные выражения в языке Си?
- 27) Как организовать бинарное и множественное ветвления в языке Си?
- 28) Как организованы циклы в языке Си?
- 29) Какие есть операторы разрыва, продолжения, перехода и возврата в языке Си?
- 30) Что такое функции в языке Си, как их объявить и определить? Что такое аргументы функций формальные и фактические?
- 31) Какие есть классы памяти в языке Си?
- 32) Как производится инициализация переменных, принадлежащих различным классам памяти в языке Си?
- 33) Какие есть модели памяти?
- 34) Что такое одномерные массивы в языке Си, как производится их описание и инициализация?
- 35) Как передать одномерные массивы функциям в качестве параметров в языке Си?
- 36) Что такое многомерные массивы в языке Си, их описание и инициализация?

37) Как передать многомерные массивы функциям в качестве параметров в языке Си?

38) Каков принцип рекурсии? Какие Вы знаете примеры формирования рекурсивных функций?

39) Что такое строки в языке Си?

40) Какие есть стандартные функции работы со строками в языке Си?

41) Что такое указатели в языке Си, их описание, инициализация? Какие есть операции над указателями?

42) Какова связь массивов с указателями в языке Си?

43) Какие есть функции для работы с динамической памятью?

44) Как создать одномерный динамический массив?

45) Что такое свободные массивы в языке Си?

46) Что такое указатели на функции в языке Си (описание типа указатель на функцию, передача указателя на функцию в качестве параметров)?

II семестр

1) Какие есть побитовые операции в языке Си?

2) Что такое структуры в языке Си? Их описание и использование, передача структур в качестве параметра переменных.

3) Как создать в динамической памяти структуры для работы с матрицами произвольного порядка?

4) Что такое поля битов в структурах в языке Си?

5) Что такое перечисляемый тип в языке Си?

6) Что такое Оператор определения типов в языке Си?

7) Что такое объединения в языке Си?

8) Что такое файлы в языке Си? Что такое функции ввода-вывода потоком?

9) Как организован прямой доступ к потокам в языке Си?

10) Какие функции содержит библиотека `stdio.h`?

11) Как организована работа с файлами на нижнем уровне?

12) Что такое и зачем нужен препроцессор языка Си?

13) Что такое условная компиляция в языке Си?

14) Что такое модульное программирование?

15) Создание программ на языке Си из нескольких файлов?

16) Каков принцип инкапсуляции? Что такое наследование и полиморфизм?

17) Что такое конструкторы и деструкторы? Что такое дружественные классы и функции?

18) Что такое статические элементы класса? Что такое константные методы и объекты?

19) Что такое указатели и ссылки на объекты, массивы объектов?

20) Какие есть классы памяти в языке Си?

21) Какие есть модели памяти в Си?

22) Что такое одномерные массивы в языке Си, как производится их описание инициализация?

23) Как передать одномерные массивы функциям в качестве параметров в языке Си?

24) Что такое многомерные массивы в языке Си, их описание инициализация?

25) Как передать многомерные массивы функциям в качестве параметров в языке Си?

26) Что такое указатели в языке Си, их описание, инициализация? Операции над указателями?

27) Как создать одномерный динамический массив?

28) Что такое свободные массивы в языке Си?

29) Что такое указатели на функции в языке Си (описание типа указатель на функцию, передача указателя на функцию в качестве параметров)?

30) Что такое структуры в языке Си? Их описание и использование, передача структур в качестве параметра переменных.

31) Как создать в динамической памяти структуры для работы с матрицами произвольного порядка?

32) Что такое роля битов в структурах в языке Си?

33) Какова организация прямого доступа к потокам в языке Си?

34) Как работать с файлами на нижнем уровне?

35) Что такое препроцессор языка Си?

36) Как происходит условная компиляция в языке Си?

6.6 Примерная тематика курсовых работ

Разработать программу с использованием языка программирования Си.

Задача решается согласно индивидуального варианта, приведенного в таблице ниже. Индивидуальное задание студент выбирает самостоятельно (таблица 7).

Таблица 7 – Индивидуальные варианты заданий

№	Задача	Примечание
1	2	3
1	Статистическая обработка данных *	1) Ввод данных из файла 2) Вычисление для массива данных: среднего значения, дисперсии, математического ожидания, среднеквадратичного отклонения, среднегеометричного значения 3) Запись результатов в отдельный файл *.html
2	Аппроксимация, интерполяция и экстраполяция функций, заданных таблицей данных *	1) Ввод данных из файла *.txt 2) Выбор наилучшего метода аппроксимации по критерию наименьшей ошибки 3) Интерполяция и экстраполяция за пределами диапазона исходных данных 4) Вывод результатов на форму и в файл *.html.

Продолжение таблицы 7

1	2	3
3	Решение систем линейных уравнений произвольной размерности методом Гаусса *	1) Ввод данных из файла *.txt 2) Анализ исходных данных 3) Выбор метода решения 4) Вывод результатов на экран и в файл 5) Предусмотреть решение недоопределенных и переопределенных уравнений 6) Вывод результатов в файл *.html.
4	Поиск экстремумов функций первого и второго порядка, заданных таблицей данных *	1) Ввод данных из файла 2) Выбор метода поиска *.txt 3) Поиск глобальных и локальных экстремумов 4) Вывод результатов на экран и в файл *.html.
5	Интегрирование функций, заданных таблицей данных *	1) Ввод данных из файла *.txt 2) Выбор метода интегрирования (3) 3) Вывод результатов на форму и в файл *.html.
6	Интегрирование функций, заданных формулой **	1) Ввод данных из файла *.txt 2) Выбор метода интегрирования (2) 3) Вывод результатов на форму и в файл *.html.
7	Дифференцирование функций, заданных формулой **	1) Ввод данных из файла *.txt 2) Выбор метода дифференцирования (2) 3) Вывод результатов на форму и в файл *.html.
8	Работа с квадратными матрицами произвольного размера	1) Ввод данных из файла *.txt 2) Вычисление детерминанта 3) Вычисление суммы элементов матрицы 4) Поиск минимального и максимального элементов с определением индексов строки и столбца. 5) Поиск нулевых элементов с определением индексов строки и столбца. 6) Вывод исходных данных и результатов в файл *.html.
9	Работа с массивами произвольного размера	1) Ввод данных из файла *.txt 2) Вычисление суммы, разности, произведения и частного двух массивов размером $n \times k$ и $k \times \ell$. 3) Поиск минимального и максимального элементов массивов с определением индексов строки и столбца. 4) Вывод исходных данных и результатов в файл *.html.
10	Вычисление производных функций, заданных таблицей данных *	1) Ввод данных из файла *.txt 2) Выбор метода дифференцирования (2) 3) Вывод результатов на экран и в файл
11	Сортировка данных в файлах	1) Файл формата *.txt может содержать: - один или несколько столбцов из данными - числа или слова 2) Сортировки осуществляется за одним или несколькими столбцами 3) Для сортировки может задаваться дополнительный признак 4) Результат вывести в файл формата *.html.

Продолжение таблицы 7

1	2	3
12	Вычисление функций, заданных аналитически **	1) Функция задается аналитически, например, так: $y = 20 * \text{Cos}(\text{Sqrt}(x) * 2)$ 2) Функция может содержать только допустимые символы, знаки операций 3) Предусмотреть возможность записи таблицы функции в файл *.html. 4) Предусмотреть использование функций $\sin, \cos, \text{tg}, \sqrt{\quad}, \sqrt[x]{\quad}, e^x$
13	Поиск экстремумов функций, заданных аналитически **	1) Функция задается аналитически, например, так: $y = 20 * \text{Cos}(\text{Sqrt}(x) * 2)$ 2) Функция может содержать только допустимые символы, знаки операций 3) Предусмотреть возможность записи таблицы функции в файл *.html. 4) Предусмотреть использование функций $\sin, \cos, \text{tg}, \sqrt{\quad}, \sqrt[x]{\quad}, e^x$ 5) Поиск глобального и локальных экстремумов функции
14	Регрессионный анализ функции, заданной таблицей данных *	1) Ввод данных из файла *.txt 2) Подбор наилучшей регрессионной зависимости 3) Вычисление коэффициентов регрессии 4) Запись исходных данных и результатов в отдельный файл *.html
15	Решение систем линейных уравнений произвольной размерности итерационным методом *	1) Ввод данных из файла *.txt 2) Анализ исходных данных 3) Выбор метода решения 4) Вывод результатов на экран и в файл 5) Предусмотреть решение недоопределенных и переопределенных уравнений 6) Вывод результатов в файл *.html.
16	Регрессионный анализ многомерной функции, заданной таблицей данных **	1) Ввод данных из файла *.txt 2) Количество независимых переменных – от 2 до 5 3) Количество зависимых переменных – от 1 до 3 3) Подбор наилучшей регрессионной зависимости 4) Вычисление коэффициентов регрессии 5) Построение таблицы исходной функции и таблицы функции регрессии. 6) Вычисление ошибки регрессии 7) Запись исходных данных и результатов в файл *.html
17	Описание многомерной функции, заданной таблицей данных, с помощью системы одновременных уравнений **	1) Ввод данных из файла *.txt 2) Количество независимых переменных – от 2 до 5 3) Количество зависимых переменных – от 1 до 3 3) Подбор наилучшей регрессионной зависимости 4) Вычисление коэффициентов регрессии 5) Построение таблицы исходной функции и таблицы функции регрессии. 6) Вычисление ошибки регрессии 7) Запись исходных данных и результатов в файл *.html

Окончание таблицы 7

1	2	3
18	Представление функции, заданной таблицей данных, с помощью рядов Фурье **	<ol style="list-style-type: none"> 1) Ввод данных из файла *.txt 2) Количество гармоник определяется заданной ошибкой вычисления: от 0,01% до 1%. 3) Запись исходных данных, результатов в виде таблицы в отдельный файл *.html
19	Аппроксимация, интерполяция и экстраполяция функций, заданных таблицей данных за методом Лагранжа *	<ol style="list-style-type: none"> 1) Ввод данных из файла *.txt 2) Интерполяция и экстраполяция за пределами диапазона исходных данных 3) Вычисление ошибок интерполяции и экстраполяции 4) Вывод результатов на форму и в файл *.html.
20	Решение систем нелинейных уравнений произвольной размерности методом Ньютона *	<ol style="list-style-type: none"> 1) Ввод данных из файла *.txt 2) Анализ исходных данных 3) Выбор метода решения 4) Вывод результатов на форму и в файл 5) Предусмотреть использование функций \sin, \cos, tg, $\sqrt{\quad}$, $\sqrt[3]{\quad}$, e^x 6) Вывод результатов в файл *.html.
21	Решение систем нелинейных уравнений произвольной размерности методом итераций *	<ol style="list-style-type: none"> 1) Ввод данных из файла *.txt 2) Анализ исходных данных 3) Выбор метода решения 4) Вывод результатов на форму и в файл 5) Предусмотреть использование функций \sin, \cos, tg, $\sqrt{\quad}$, $\sqrt[3]{\quad}$, e^x 6) Вывод результатов в файл *.html.
22	Решение систем нелинейных уравнений произвольной размерности методом покоординатного спуска *	<ol style="list-style-type: none"> 1) Ввод данных из файла *.txt 2) Анализ исходных данных и выбор метода решения 3) Вывод результатов на форму и в файл 4) Предусмотреть использование функций \sin, \cos, tg, $\sqrt{\quad}$, $\sqrt[3]{\quad}$, e^x 5) Вывод результатов в файл *.html.
23	Решение систем нелинейных уравнений произвольной размерности методом градиентного спуска *	<ol style="list-style-type: none"> 1) Ввод данных из файла *.txt 2) Анализ исходных данных 3) Выбор метода решения 4) Вывод результатов на форму и в файл 5) Предусмотреть использование функций \sin, \cos, tg, $\sqrt{\quad}$, $\sqrt[3]{\quad}$, e^x 6) Вывод результатов в файл *.html.
24	Решение систем линейных уравнений произвольной размерности методом обращения *	<ol style="list-style-type: none"> 1) Ввод данных из файла *.txt 2) Анализ исходных данных 3) Выбор метода решения 4) Вывод результатов на экран и в файл 5) Предусмотреть решение недоопределенных и переопределенных уравнений 6) Вывод результатов в файл *.html.

Примечания:

- * – на повышенную оценку (от 74 до 90 баллов);
- ** – на повышенную оценку (от 90 до 100 баллов).

В таблице 8 представлена структура пояснительной записки.

Таблица 8 – Структура пояснительной записки

Раздел	Количество страниц
1	2
Титульный лист	1
Задание и календарный план	1
РЕФЕРАТ	1
ВВЕДЕНИЕ	2-3
1 АНАЛИЗ ЗАДАНИЯ	
1.1 Теоретические сведения, необходимые для выполнения работы	5-6
1.2 Концепция решения задачи	1-2
1.3 Описание переменных и констант	2-3
2 РАЗРАБОТКА ТРЕБОВАНИЙ К ПРОГРАММЕ И АЛГОРИТМОВ	
2.1 Разработка общих требований к программе и ее структуры	3-5
2.2 Разработка интерфейса пользователя	3-5
2.3 Разработка алгоритмов расчетов	5-7
3 ПРОГРАММНАЯ РЕАЛИЗАЦИЯ АЛГОРИТМОВ И УПРАВЛЯЮЩЕЙ ПРОГРАММЫ	
3.1 Создание подпрограмм для реализации алгоритмов	2-4
3.2 Реализация программного кода для управляющей программы	3-4
3.3 Тестирование программы	1-2
ВЫВОДЫ	1
ПЕРЕЧЕНЬ ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	1
ПРИЛОЖЕНИЕ А ФОРМЫ ПРОГРАММЫ	2-4
ПРИЛОЖЕНИЕ Б ПРОГРАММНОЕ КОД	5-6
ПРИЛОЖЕНИЕ В ТЕСТОВЫЙ ПРИМЕР	2-4
ВСЕГО	40-60

После завершения работы преподавателю предоставляется пояснительная записка в объеме 40-60 страниц в печатном и электронном виде в формате Word 2003/2007, оформленная по требованиям стандартов, а также рабочая программа (файлы проекта и скомпилированный *exe-файл*), записанную на CD/DVD – диск.

Исходные данные для тестирования программы студент создает самостоятельно. Для заданий, которые связаны с обработкой статистических данных, размер выборки должен быть не менее 50 записей.

7 Учебно-методическое и информационное обеспечение дисциплины

7.1 Рекомендуемая литература

Основная литература

1. Зыков, С. В. Программирование: учебник и практикум для вузов / С. В. Зыков. — Текст : Москва: Издательство Юрайт, 2021. — 320 с. — (Высшее образование). — непосредственный. Режим доступа: <https://ntb.tti.sfedu.ru/wp-content/uploads/2023/03/Программирование.pdf> (Дата обращения 26.08.2024).

Дополнительная литература

2. Брайан Керниган, Деннис Ритчи. Язык программирования С. — Москва: Вильямс, 1992. — 272 с. — 4 экз.

3. Стенли Липпман Язык программирования С++ [Электронный ресурс]: полное руководство/ Стенли Липпман, ЖозиЛажойе— Электрон. текстовые данные.— М.: ДМК Пресс, 2019.— 1104 с. Режим доступа: <http://www.iprbookshop.ru/6899.html> (Дата обращения 26.08.2024).

Учебно-методическое обеспечение

1. Сборник задач к практическим занятиям по курсу «Программирование» : практикум (для студ. специальности 10.05.03 «Информационная безопасность автоматизированных систем» 1 курса очной формы обучения) / сост.: Е.Е. Бизянов, А.А. Гутник, Л.В. Толстова ; Каф. Специализированных компьютерных систем . — Алчевск : ФГБОУ ВО ДонГТУ, 2023 . — 29 с. — URL: <http://library.dstu.education/download.php?rec=132304>. (Дата обращения 26.08.2024).

2. Бизянов, Е.Е. Основы программирования. Часть 1 : лабораторный практикум / Е.Е. Бизянов, Л.В. Толстова. — Алчевск :ФГБОУ ВО "ДонГТУ", 2023. — 134 с. — URL: <http://library.dstu.education/download.php?rec=103329>.

3. Бизянов, Е.Е. Основы программирования. Часть 2 : лабораторный практикум / Е.Е. Бизянов, Л.В. Толстова. — Алчевск : ФГБОУ ВО "ДонГТУ", 2023) — 92 с. — URL: <http://library.dstu.education/download.php?rec=132257>. (Дата обращения 26.08.2024).

7.2 Базы данных, электронно-библиотечные системы, информационно-справочные и поисковые системы

1. Научная библиотека ДонГТУ : официальный сайт.— Алчевск. — URL: library.dstu.education.—Текст : электронный.

2. Научно-техническая библиотека БГТУ им. Шухова : официальный сайт. — Белгород. — URL: <http://ntb.bstu.ru/jirbis2/>.—Текст : электронный.

4. Университетская библиотека онлайн :электронно-библиотечная система.— URL: http://biblioclub.ru/index.php?page=main_ub_red.—Текст : электронный.

5. Сайт кафедры ИСИБ <http://scs.dstu.education>.

8 Материально-техническое обеспечение дисциплины

Материально-техническая база обеспечивает проведение всех видов деятельности в процессе обучения, соответствует требованиям ФГОС ВО.

Материально-техническое обеспечение представлено в таблице 9.

Таблица 9 – Материально-техническое обеспечение

Наименование оборудованных учебных кабинетов	Адрес (местоположение) учебных кабинетов
<p>Аудитория для проведения лекций</p> <p><i>Компьютерные классы (22 посадочных места), оборудованный учебной мебелью, компьютерами с неограниченным доступом к сети Интернет, включая доступ к ЭБС:</i> ПК – 11шт.; Интерактивная доска– 1 шт.</p> <p>ПК - 11 шт.; Доска– 1 шт.</p>	<p>ауд. <u>309</u> корп. <u>4</u></p> <p>ауд. <u>208</u> корп. <u>4</u></p> <p>ауд. <u>211</u> корп. <u>4</u></p>

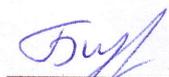
Лист согласования рабочей программы дисциплины

Разработал
И.о. заведующего кафедрой
интеллектуальных систем и
информационной безопасности
(должность)


(подпись)

Е.Е. БИЗЯНОВ
(Ф.И.О.)

И.о. заведующего кафедрой
интеллектуальных систем и
информационной безопасности
(наименование кафедры)

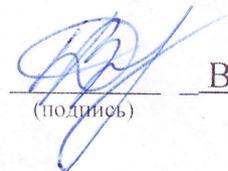

(подпись)

Е.Е. БИЗЯНОВ
(Ф.И.О.)

Протокол № 1 заседания кафедры
интеллектуальных систем и
информационной безопасности

от 27.08. 20 24 г.

И.о. декана факультета
информационных технологий и
автоматизации производственных
процессов


(подпись)

В.В. Дьячкова.
(Ф.И.О.)

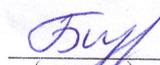
Согласовано

Председатель методической
комиссии по направлению подготовки 02.03.01
Математика и компьютерные науки


(подпись)

Н.Н. Лепило
(Ф.И.О.)

Председатель методической
комиссии по направлению подготовки 09.03.01
Информатика и вычислительная техника


(подпись)

Е.Е. БИЗЯНОВ
(Ф.И.О.)

Председатель методической
комиссии по специальности 10.05.03
Информационная безопасность
автоматизированных систем


(подпись)

Е.Е. БИЗЯНОВ
(Ф.И.О.)

Председатель методической
комиссии по направлению подготовки 38.03.05
Бизнес-информатика


(подпись)

Н.Н. Лепило
(Ф.И.О.)

Начальник учебно-методического центра


(подпись)

О.А.Коваленко
(Ф.И.О.)

Лист изменений и дополнений

Номер изменения, дата внесения изменения, номер страницы для внесения изменений	
ДО ВНЕСЕНИЯ ИЗМЕНЕНИЙ:	ПОСЛЕ ВНЕСЕНИЯ ИЗМЕНЕНИЙ:
Основание:	
Подпись лица, ответственного за внесение изменений	