

Минобрнауки России

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

«Оренбургский государственный университет»

Кафедра компьютерной безопасности и математического обеспечения информационных систем

## **РАБОЧАЯ ПРОГРАММА**

**ДИСЦИПЛИНЫ**

*«Б1.Д.Б.22 Языки программирования»*

Уровень высшего образования

**СПЕЦИАЛИТЕТ**

Специальность

*10.05.01 Компьютерная безопасность*

(код и наименование специальности)

*специализация №3 «Разработка защищенного программного обеспечения»*

(наименование направленности (профиля)/специализации образовательной программы)

Квалификация

*Специалист по защите информации*

Форма обучения

*Очная*

Год набора 2023

Рабочая программа дисциплины «Б1.Д.Б.22 Языки программирования» рассмотрена и утверждена на заседании кафедры

Кафедра компьютерной безопасности и математического обеспечения информационных систем  
наименование кафедры

протокол № 7 от "9" марта 2023г.

Заведующий кафедрой

Кафедра компьютерной безопасности и математического обеспечения информационных систем

наименование кафедры И.В. Влацкая  
подпись И.В. Влацкая расшифровка подписи

Исполнители:

доцент К.Р. Джукашев  
должность подпись расшифровка подписи

должность подпись расшифровка подписи

СОГЛАСОВАНО:

Председатель методической комиссии по специальности  
10.05.01 Компьютерная безопасность И.В. Влацкая  
код наименование личная подпись расшифровка подписи

Заведующий отделом формирования фонда и научной обработки документов

Н.Н. Бигалиева  
личная подпись расшифровка подписи

Уполномоченный по качеству факультета

И.В. Крючкова  
личная подпись расшифровка подписи

№ регистрации 158940

## 1 Цели и задачи освоения дисциплины

**Цель (цели)** освоения дисциплины:

Формирование у будущих специалистов практических навыков по основам алгоритмизации вычислительных процессов и программированию решения вычислительных, экономических и других задач.

**Задачи:**

- изучение основ алгоритмизации;
- приобретение базовых навыков создания консольных и оконных программ на языке C++;
- получение навыков разработки консольных и оконных приложений с управляемым кодом с помощью языка C#.

## 2 Место дисциплины в структуре образовательной программы

Дисциплина относится к базовой части блока Д «Дисциплины (модули)»

Пререквизиты дисциплины: *Б1.Д.Б.10 Информатика*

Постреквизиты дисциплины: *Б1.Д.Б.23 Методы программирования, Б1.Д.Б.25 Операционные системы, Б1.Д.Б.27 Системы управления базами данных, Б1.Д.Б.34 Защита программ и данных, Б1.Д.Б.38 Методы и средства криптографической защиты информации, Б1.Д.Б.44.2 Анализ программных реализаций, Б1.Д.Б.44.5 Технология создания прикладного программного обеспечения, Б1.Д.Б.44.6 Объектно-ориентированные языки и системы, Б1.Д.Б.44.7 Параллельное программирование, Б1.Д.Б.44.8 Структуры и алгоритмы компьютерной обработки данных, Б1.Д.В.3 Разработка и применение систем искусственного интеллекта, Б1.Д.В.7 Разработка мобильных приложений, Б1.Д.В.Э.2.1 Web-программирование, Б2.П.Б.У.1 Ознакомительная практика, Б2.П.Б.П.1 Научно-исследовательская работа, Б2.П.В.П.1 Производственная практика (по специализации)*

## 3 Требования к результатам обучения по дисциплине

Процесс изучения дисциплины направлен на формирование следующих результатов обучения

Код и наименование формируемых компетенций	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций
ОПК-7 Способен создавать программы на языках высокого и низкого уровня, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ	ОПК-7-В-1 Знает язык программирования высокого уровня ОПК-7-В-2 Умеет работать с интегрированной средой разработки программного обеспечения ОПК-7-В-3 Умеет разрабатывать и реализовывать на языке высокого уровня алгоритмы решения типовых профессиональных задач	<b>Знать:</b> - современные тенденции в области разработки программного обеспечения, включая концепции и методологии проектирования и кодирования программного обеспечения; - базовые структуры данных, основные алгоритмы сортировки и поиска данных, комбинаторные и

Код и наименование формируемых компетенций	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций
		<p>теоретико-графовые алгоритмы;</p> <ul style="list-style-type: none"> <li>- основные возможности языка программирования C++, C# и платформы .NET.</li> <li>- основные угрозы безопасности информации в разрабатываемых информационных</li> </ul> <p><b><u>Уметь:</u></b></p> <ul style="list-style-type: none"> <li>- разрабатывать приложения на языке программирования C++, C#;</li> <li>- использовать возможности языка C++, C# для реализации различных приложений с использованием объектно-ориентированного, функционального и структурного программирования</li> <li>- использовать возможности языка программирования C# и платформы .NET для устранения основных угроз, связанных с ошибками в разрабатываемом программном обеспечении.</li> </ul> <p><b><u>Владеть:</u></b></p> <ul style="list-style-type: none"> <li>- навыками проектирования и разработки приложений с использованием языков C#, C++;</li> <li>- навыками работы со средой разработки Visual Studio Code.</li> </ul>

## 4 Структура и содержание дисциплины

### 4.1 Структура дисциплины

Общая трудоемкость дисциплины составляет 12 зачетных единиц (432 академических часа).

Вид работы	Трудоемкость, академических часов			
	1 семестр	2 семестр	3 семестр	всего
<b>Общая трудоёмкость</b>	<b>144</b>	<b>144</b>	<b>144</b>	<b>432</b>
<b>Контактная работа:</b>	<b>84,25</b>	<b>85,25</b>	<b>85,5</b>	<b>255</b>
Лекции (Л)	34	34	34	102
Лабораторные работы (ЛР)	50	50	50	150
Консультации		1		1
Индивидуальная работа и инновационные формы учебных занятий			1	1
Промежуточная аттестация (зачет, экзамен)	0,25	0,25	0,5	1
<b>Самостоятельная работа:</b> - выполнение курсовой работы (КР); - самоподготовка (проработка и повторение лекционного материала и материала учебников и учебных пособий); - подготовка к лабораторным занятиям; - подготовка к рубежному контролю и т.п.)	<b>59,75</b>	<b>58,75</b>	<b>58,5</b> +	<b>177</b>
<b>Вид итогового контроля (зачет, экзамен, дифференцированный зачет)</b>	<b>зачет</b>	<b>экзамен</b>	<b>зачет</b>	

Разделы дисциплины, изучаемые в 1 семестре

№ раздела	Наименование разделов	Количество часов				
		всего	аудиторная работа			внеауд. работа
			Л	ПЗ	ЛР	
1	Введение в алгоритмизацию и программирование на языке C++	8	4			4
2	Основные понятия языка программирования. Операторы и алгоритмические конструкции языка C++	38	6		16	16
3	Работа с указателями, массивами и строками в языке C++	52	8		24	20
4	Структурное и модульное программирование с использованием языка C++	46	16		10	20
	Итого:	144	34		50	60

Разделы дисциплины, изучаемые в 2 семестре

№ раздела	Наименование разделов	Количество часов				
		всего	аудиторная работа			внеауд. работа
			Л	ПЗ	ЛР	
4	Структурное и модульное программирование с использованием языка C++ (продолжение)	88	12		36	40
5	Объектно-ориентированное программирование с использованием языка C++	56	22		14	20

№ раздела	Наименование разделов	Количество часов				
		всего	аудиторная работа			внеауд. работа
			Л	ПЗ	ЛР	
	Итого:	144	34		50	60

Разделы дисциплины, изучаемые в 3 семестре

№ раздела	Наименование разделов	Количество часов				
		всего	аудиторная работа			внеауд. работа
			Л	ПЗ	ЛР	
6	Основные понятия языка C#. Объектно-ориентированное программирование	96	18		38	40
7	Разработка приложений с помощью технологии Windows Presentation Foundation на языке C#	48	16		12	20
	Итого:	144	34		50	60
	Всего:	432	102		150	180

## 4.2 Содержание разделов дисциплины

**1 Введение в алгоритмизацию и программирование на языке C++.** Понятие программирования, программы, структуры данных. Основные этапы решения задачи на компьютере.

Понятие Алгоритма. Свойства алгоритмов. Формы записи алгоритма.

Блок-схемы. Основные типы блоков. Правила соединения блоков.

Базовые структуры алгоритмов. Дополнительные структуры алгоритмов.

Понятие языка программирования. Составляющие элементы языка программирования: алфавит, лексемы, синтаксис, семантика. Классификация языков программирования. Понятие компилируемых и интерпретируемых языков. Инструменты программирования.

Компилятор. Упрощенная модель компилятора.

**2 Основные понятия языка программирования.** Операторы и алгоритмические конструкции языка C++. Состав языка программирования C++: алфавит, лексемы, выражения, операторы. Виды лексем: идентификаторы, ключевые слова, знаки операций, константы, разделители.

Понятие типа данных. Классификация типов данных в языке C++: базовые и составные типы. Целочисленные, вещественные, символьный, логический и пустой типы данных.

Типичная структура программы на языке C++. Переменные, их описание, виды.

Выражения, операнды, операции языка C++. Унарные, бинарные и тернарные операции.

Классификация операций по назначению. Операция присваивания, преобразование типов в операции присваивания, сокращенные операции присваивания. Арифметические операции. Операции сравнения. Логические операции.

Выражения, операнды, операции языка C++. Побитовые операции. Специальные операции.

Операция приведения типов. Приоритеты операций. Преобразование типов в выражениях.

Понятие оператора. Виды операторов C++. Составной блок. Условный оператор. Оператор выбора. Понятие цикла. Виды операторов цикла: цикл с параметром, цикл с предусловие, цикл с постусловием. Операторы передачи управления.

**3 Работа с указателями, массивами и строками в языке C++.** Понятие указателя. Виды указателей: указатель на объект, указатель на void, указатель на функцию. Инициализация указателей. Константные указатели. Операции с указателями. Назначение указателей.

Структура памяти программы. Динамические переменные. Ссылки.

Понятие массива. Доступ к элементам массива по индексу. Инициализация массива.

Операции над элементами массива: ввод, вывод, нахождение минимального элемента, сортировка.

Многомерные массивы. Доступ к элементам многомерного массива по индексу.

Расположение многомерных массивов в памяти. Инициализация многомерных массивов. Операции над элементами двумерного массива: ввод, вывод, нахождение минимального элемента.

Динамические массивы. Выделение и освобождение памяти. Двумерные динамические массивы.

Понятие строки. Инициализация строк. Ввод строки с консоли. Печать строки на экран.

Основные функции для работы со строками: `strlen`, `strcmp`, `strcpy`, `strcat`, `strchr`, `strstr`. Ввод строки с клавиатуры. Печать строки на экран. Функции преобразования различных типов данных в строку и обратно.

**4 Структурное и модульное программирование с использованием языка C++.** Понятие функции. Объявление и определение функции. Вызов функции. Формальные и фактические параметры. Передача параметров в функцию по ссылке и по значению. Возврат значений из функции.

Функции, возвращающие `void`. Передача массивов в функцию.

Рекурсивные функции. Прямая и косвенная рекурсия. Назначение рекурсий, достоинства и недостатки.

Параметры функций со значениями по умолчанию. Перегрузка функций. Неоднозначность при перегрузке. Шаблоны функций. Функция `main()`.

Типы данных, определяемые пользователем. Переименование типов. Перечисления. Понятие структуры. Доступ к полям структуры. Инициализация структур. Присвоение структур.

Указатели на структуру. Битовые поля и объединения.

Понятие потока. Виды потоков. Способы работы с потоками в C++. Понятие файла.

Последовательность действий при работе с файлами. Текстовые и бинарные файлы.

Работа с файлами с помощью функций, унаследованных их языка C. Работа с файлами с помощью потоковых классов C++.

Динамические структуры данных. Линейные списки и их виды. Основные операции над односвязными и двусвязными списками на примере односвязного линейного списка.

Стеки. Операции для работы со стеками. Очереди. Операции для работы с очередями.

Модульное программирование. Основные понятия. Достоинства и недостатки. Объявление функций, переменных и констант, скрытых внутри модуля или доступных из других модулей.

Заголовочные файлы. Директивы препроцессора. Пространства имен. Примеры.

Структурный подход к разработке программ: нисходящее проектирование, структурное программирование, нисходящее тестирование.

**5 Объектно-ориентированное программирование с использованием языка C++.**

Объектно-ориентированное программирование. Основные принципы: инкапсуляция, наследование и полиморфизм. Понятие класса, объекта. Элементы класса, спецификаторы доступа к ним.

Конструкторы и деструкторы классов. Списки инициализаторов. Конструктор копирования класса. Указатель `this`. Статические элементы класса.

Константные методы класса. Понятие дружественных функций и дружественных классов.

Понятие перегрузки операций. Правила перегрузки операций. Перегрузка унарных и бинарных операций. Перегрузка операции присваивания, приведения типа, вызова функции и индексирования.

Наследование классов. Простое и множественное наследование. Переопределение методов, правила их наследования. Механизмы раннего и позднего связывания. Виртуальные методы, правила их использования. Абстрактные классы.

Шаблоны классов. Синтаксис и правила описания. Инстанцирование. Специализация отдельных методов и всего класса.

Исключения. Принципы их обработки. Понятия контролируемого блока, обработчика и генерации исключения. Стандартные исключения.

**6 Основные понятия языка C#. Объектно-ориентированное программирование.**

Платформа .NET. Особенности создания управляемого кода для CLR. Автоматический сборщик мусора. Межязыковое взаимодействие. Общезыковая среда выполнения и объектная модель.

Понятие сборки.

Основные принципы, инструменты, среды разработки приложений для платформы .NET.

Язык C#. Структура программы, алфавит, ключевые слова, система типов CLR (ссылочные типы и типы-значения), переменные, знаки операций, выражения, основные операторы (линейные, ветвления, выбора, программирования циклических конструкций, безусловного перехода), консольный ввод-вывод. Преобразование типов. Операторы as и is. Сравнение типов-значений и типов-ссылок.

Одномерные и многомерные массивы. Пользовательские типы данных. Перечисления, структуры.

Объектно-ориентированное программирование на языке C#. Классы, объекты, поля, методы, свойства, делегаты. Спецификаторы видимости элементов класса. Передача параметров по ссылке, по значению, по возврату. Статические поля и методы. Перегрузка методов.

Объект this. Конструкторы по умолчанию, статические, списки инициализаторов.

Жизненный цикл объектов, деструкторы.

Перегрузка унарных и бинарных операций. Перегрузка индексирования, инкремента и декремента, операций сравнений, преобразования типа.

Константные поля и поля readonly.

Наследование классов. Полиморфизм. Ранее и позднее связывание. Виртуальные методы.

Абстрактные классы. Соккрытие полей и методов. Sealed-классы.

Интерфейсы. Определение и реализация. Стандартные интерфейсы ICloneable, IEnumerator, IEnumerable, IComparable, IDisposable.

Базовый класс object. Механизмы boxing и unboxing.

Пространства имен.

Строки в .Net. Типы String и StringBuilder.

Основные контейнеры: ArrayList, Queue, Stack, Hashtable.

Работа с файлами и потоками в .NET. Классы File, Directory. Потоки StreamReader, StreamWriter, BinaryReader, BinaryWriter.

Принципы и механизмы обработки исключений.

**7 Разработка приложений с помощью технологии Windows Presentation Foundation на языке C#.** Основные понятия технологии Windows Presentation Foundation (WPF), разделение интерфейса и кода, язык XAML, архитектура WPF, принципы компиляции.

Элементы управления, стили, шаблоны расположения, анимации. Связывание с данными.

Двумерные и трехмерные изображения. Проигрывание аудио и видео.

### 4.3 Лабораторные работы

№ ЛР	№ раздела	Наименование лабораторных работ	Кол-во часов
1.	2	Основы алгоритмизации. Линейное программирование на языке C++	4
2.	2	Программирование конструкций ветвления языка C++	4
3.	2	Циклы с предусловием и постусловием в языке C++	4
4.	2	Счетные циклы в языке C++	4
5.	3	Одномерные массивы в языке C++	4
6.	3	Одномерные массивы в языке C++. Сортировка.	4
7.	3	Многомерные массивы в языке C++	4
8.	3	Работа с указателями в языке C++	4
9.	3	Работа с динамическими массивами в C++	4
10.	3	Работа со строками в языке C++	4
11.	4	Использование функций при создании программ с помощью языка C++	6
12.	4	Шаблоны функций и перегрузки	4
13.	4	Работа с файлами в языке C++	4
14.	4	Пользовательские типы данных. Работа со структурами в языке C++	4
15.	4	Программирование линейных односвязных списков	8
16.	4	Программирование линейных двухсвязных списков	8

№ ЛР	№ раздела	Наименование лабораторных работ	Кол-во часов
17.	4	Программирование стеков и очередей	8
18.	4	Использование стандартной библиотеки шаблонов STL	4
19.	5	Введение в объектно-ориентированное программирование с использованием языке C++	4
20.	5	Наследование классов. Ранее и позднее связывание. Абстрактные классы	4
21.	5	Работа с шаблонами классов	4
22.	5	Программирование и обработка исключений в языке C++	2
23.	6	Введение в язык программирования C#. Программирование различных алгоритмических конструкций	2
24.	6	Одномерные массивы в языке C#	4
25.	6	Двумерные массивы в языке C#. Строки	4
26.	6	Работа со стандартными контейнерными классами .NET. Стеки и очереди	4
27.	6	Работа со стандартными контейнерными классами .NET. Словари, списки, множества	4
28.	6	Введение в объектно-ориентированное программирование на языке C#. Перегрузка операций.	4
29.	6	Введение в объектно-ориентированное программирование на языке C#. Наследование классов.	4
30.	6	Введение в объектно-ориентированное программирование на языке C#. Интерфейсы, абстрактные и статические классы.	4
31.	6	Введение в объектно-ориентированное программирование на языке C#. Генерики и исключения	4
32.	6	Введение в объектно-ориентированное программирование на языке C#. Делегаты, события и исключения	4
33.	7	Введение в Windows Presentation Foundation. Оконный интерфейс	6
34.	7	Использование шаблонов, стилей и анимаций	6
		Итого:	150

#### 4.4 Курсовая работа (3 семестр)

Примерные темы курсовой работы:

##### 1. Обработка идентификаторов программы на языке C++

В файле содержится программа на языке C++. Напечатать в алфавитном порядке все не зарезервированные идентификаторы программы. Для каждого идентификатора указывать номера строк, где встречается данный идентификатор. Считать, что программа состоит только из одного модуля, включающего единственную функцию main.

##### 2. Частотный словарь идентификаторов программы на языке C++

В файле содержится программа на языке C++. Напечатать в алфавитном порядке все не зарезервированные идентификаторы программы, указывая для каждого из них число вхождений в текст программы. Считать, что программа состоит только из одного модуля, включающего единственную функцию main.

##### 3. Классификация идентификаторов программы на языке C++

В файле содержится программа на языке C++. Определить для каждого идентификатора его категорию. Для каждого идентификатора указывать номера строк, где встречается данный идентификатор. Считать, что программа состоит только из одного модуля, включающего единственную функцию main.

#### 4. Построение программы по арифметическому выражению

По заданному арифметическому выражению, записанному на языке C++, построить и записать в файл готовую линейную программу, в которой каждое действие реализуется определенным оператором присваивания, значения всех необходимых переменных читается с консоли, а результат вычисления выражения печатается на экран.

#### 5. Построение графиков стандартных функций

Написать программу для построения графиков стандартных математических функций:

$$Y = KX + B$$

$$Y = AX^2 + BX + C$$

$$Y = A \sin x + B \cos x$$

$$Y = A / (BX + C)$$

Программа должна отображать оси координат, предоставлять возможность масштабирования графиков.

#### 6. Нахождение корня нелинейного уравнения методом половинного деления

Найти корень уравнения  $f(x)=0$  на заданном интервале изоляции корня  $[a;b]$  методом половинного деления (бисекции). Для функции  $y=f(x)$  программа также строит ее график и отмечает на нем корень.

### 5 Учебно-методическое обеспечение дисциплины

#### 5.1 Основная литература

**1. Павлоградский, В. В. C++Builder. Учебный курс : учебное пособие / В. В. Павлоградский, В. В. Пальчиковский. — Пермь : ПНИПУ, 2014. — 293 с. — ISBN 978-5-398-01215-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/160810> (дата обращения: 20.04.2023). — Режим доступа: для авториз. пользователей.**

**2. Вафин, Р. Р. Программирование на C#.NET : учебное пособие / Р. Р. Вафин, А. Р. Бикмурзина. — Казань : КНИТУ-КАИ, 2020. — 108 с. — ISBN 978-5-7579-2430-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/264917> (дата обращения: 20.04.2023). — Режим доступа: для авториз. пользователей.**

#### 5.2 Дополнительная литература

**1. Павловская, Т. А. C/C++. Структурное программирование [Текст] : практикум / Т. А. Павловская, Ю. А. Шупак. - СПб. : Питер, 2005. - 239 с. - (Учебное пособие). - Прил.: с. 212-235. - Алф. указ.: с. 236-238. - ISBN 5-94723-967-1.**

**2. Павловская, Т. А. C/C++. Программирование на языке высокого уровня [Текст] : учеб. для вузов / Т. А. Павловская. - СПб. : Питер, 2006. - 461 с. : ил. - (Учебник для вузов). - Алф. указ.: с. 450-460. - ISBN 5-94723-568-4.**

#### 5.3 Периодические издания

1. Вестник компьютерных и информационных технологий : журнал. - М. : Агентство "Роспечать", 2018

2. Программные продукты и системы : журнал. - М. : Агентство "Роспечать", 2017.

#### **5.4 Интернет-ресурсы**

1. <https://www.coursera.org/specializations/hands-on-cpp> - «Coursera», MOOK: «Programming in C++: A Hands-on Introduction»
2. <https://www.coursera.org/learn/c-plus-plus-a> - «Coursera», MOOK: «C++ For C Programmers, Part A»
3. <https://www.coursera.org/specializations/coding-for-everyone> - «Coursera», MOOK: «Coding for Everyone: C and C++»
3. [https://openedu.ru/course/urfu/CSHARP/?session=spring\\_2023](https://openedu.ru/course/urfu/CSHARP/?session=spring_2023) - «Открытое образование», MOOK: «Программирование на C#»

#### **5.5 Программное обеспечение, профессиональные базы данных и информационные справочные системы**

1. Операционная система Astra Linux. «Astra Linux Special Edition» РУСБ.10015-01, лицензионный договор №А-2021-1374-ВУЗ от 28.05.2021
2. Пакет офисных приложений LibreOffice
3. Программная система для организации видеоконференц-связи Webinar.ru
4. Visual Studio Code – редактор исходного кода, распространяемый под лицензией MIT – Режим доступа: <https://github.com/microsoft/vscode>

#### **6 Материально-техническое обеспечение дисциплины**

Учебные аудитории для проведения занятий лекционного типа, семинарского типа, для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации.

Аудитории оснащены комплектами ученической мебели, техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Для проведения лабораторных занятий компьютерный класс, оснащенный компьютерами с операционной системой Astra Linux текущей версии с установленным пакетом офисных программ и интегрированной средой разработки ПО.

Помещение для самостоятельной работы обучающихся оснащены компьютерной техникой, подключенной к сети "Интернет", и обеспечением доступа в электронную информационно-образовательную среду ОГУ.