


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

Факультет информационных технологий, экономики и управления

Кафедра «Системы управления»

УТВЕРЖДАЮ  
Декан ФИТЭУ

 /И.А. Рычка/

« 31 » января 2024 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

**«Программирование»**

направление подготовки:  
09.03.01 «Информатика и вычислительная техника»  
(уровень бакалавриата)

направленность (профиль):  
«Программное обеспечение средств вычислительной техники и  
автоматизированных систем»

Петропавловск-Камчатский  
2024

Рабочая программа разработана в соответствии с ФГОС ВО по направлению подготовки 09.03.01 «Информатика и вычислительная техника», профиль «Программное обеспечение средств вычислительной техники и автоматизированных систем», и учебного плана ФГБОУ ВО «КамчатГТУ».

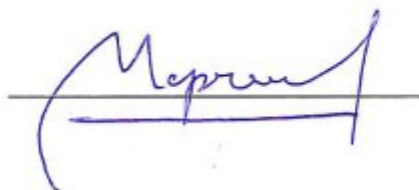
Составитель рабочей программы:

Профессор кафедры СУ, доцент, д.ф.-м.н. Марапулец Марапулец Ю.В

Рабочая программа рассмотрена на заседании кафедры «Системы управления»

Протокол № 6 от « 31 » января 2024 года.

« 31 » января 2024 г.



Заведующий кафедрой  
«Системы управления»  
А.А. Марченко

## 1. ЦЕЛЬ И ЗАДАЧИ УЧЕБНОЙ ДИСЦИПЛИНЫ

**Целью** дисциплины является ознакомление студентов с основными принципами программирования на современном языке высокого уровня C++, а так же - обучение студентов принципам построения алгоритмов с учетом архитектуры современных ЭВМ.

**Задачей** дисциплины является дать студенту теоретические и практические знания по принципам разработки программного обеспечения на языке C++ в соответствии со стандартом ISO/IEC 14882.

В результате изучения дисциплины:

- студент должен знать этапы разработки программы на ПЭВМ, уметь разрабатывать алгоритмы решения прикладных задач. Знать лексические и синтаксические основы языка программирования C++.

- студент должен уметь составлять программу решения прикладной задачи на языке C++, производить отладку программного кода в среде разработчика Visual C++.

- студент должен приобрести навыки по разработке программного кода в современных операционных системах.

## 2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина «Программирование» направлена для освоения следующих компетенций основной образовательной программы подготовки бакалавра по направлению подготовки 09.03.01 "Информатика и вычислительная техника" федерального государственного образовательного стандарта высшего образования:

- способностью разрабатывать алгоритмы и программы, пригодные для практического применения (ОПК-8).

Планируемые результаты обучения при изучении дисциплины, соотнесенные с планируемыми результатами освоения образовательной программы, представлены в таблице.

Таблица - Планируемые результаты обучения при изучении дисциплины, соотнесенные с планируемыми результатами освоения образовательной программы

Код компетенции	Наименование компетенции	Код и наименование индикатора достижения	Планируемый результат обучения по дисциплине	Код показателя освоения
ОПК-8	Обладать способностью разрабатывать алгоритмы и программы, пригодные для практического применения	ИД-1 <sub>ОПК-8</sub> Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий	Знать: ☑ основные принципы разработки программного обеспечения, методики анализа и проектирования программного кода;	3(ОПК-8)1
			☑ лексические и синтаксические основы языка программирования C++;	3(ОПК-8)2
			☑ основные концепции объектно-ориентированного программирования, методику объектно-ориентированного	3(ОПК-8)3

			анализа и проектирования.	
		<b>ИД-2</b> <sub>ОПК-8</sub> Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ	<b>Уметь:</b> спроектировать алгоритм и структуру ПО; разрабатывать программный код на языке C++; спроектировать алгоритм и структуру объектно-ориентированного ПО.	<b>У(ОПК-8)1</b> <b>У(ОПК-8)2</b> <b>У(ОПК-8)3</b>
		<b>ИД-3</b> <sub>ОПК-8</sub> Имеет навыки программирования, отладки и тестирования прототипов программно-технических комплексов задач	<b>Владеть:</b> навыками по разработке структуры ПО; навыками по разработке ПО на языке C++; навыками по разработке структуры объектно-ориентированного ПО.	<b>В(ОПК-8)1</b> <b>В(ОПК-8)2</b> <b>В(ОПК-8)3</b>

### 3. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Дисциплина Б1.О.14 "Программирование" относится к обязательной части в структуре образовательной программы.

### 4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

#### 4.1. Тематический план дисциплины

Наименование разделов и тем	Все го часов	Контактная работа	Контактная работа по видам учебных занятий			Самостоятельная работа	Формы текущего контроля	Итоговый контроль знаний по дисциплине
			Лекции	Семинары (практические занятия)	Лабораторные работы			
<b>2 семестр</b>								

<b>Тема 1:</b> Теоретические основы разработки программного обеспечения	50	<b>18</b>	10	8	-	32	Опрос, ПЗ	
<b>Тема 2:</b> Основы программирования на языке C++	154	<b>84</b>	14	34	36	70	Опрос, ПЗ, РЗ	
<b>Тема 3:</b> Принципы разработки программ с использованием средств визуального программирования в среде разработчика Visual C++	84	<b>42</b>	12	12	18	42	Опрос, ПЗ, РЗ	
Экзамен	36							36
Всего	<b>324</b>	<b>144</b>	<b>36</b>	<b>54</b>	<b>54</b>	<b>144</b>		<b>36</b>
<b>3 семестр</b>								
<b>Тема 4:</b> Основы разработки объектно-ориентированного программного кода	108	<b>51</b>	17	-	34	57	Опрос, ПЗ, РЗ, Тест	
Экзамен, КР	36			-	-	-	-	36
Всего	<b>144</b>	<b>51</b>	<b>17</b>		<b>34</b>	<b>57</b>		<b>36</b>

\*ПЗ – практическое задание, РЗ – решение задач, КС – конкретная ситуация

## 4.2. Описание содержания дисциплины

### Тема 1: Теоретические основы разработки программного обеспечения

Лекция 1.1 **Введение.** Предмет дисциплины и ее задачи. Роль и место задач разработки программного обеспечения в интегрированных производственных комплексах, автоматизированных системах управления техническими объектами. Структура и содержание дисциплины, ее связь с другими дисциплинами специальности. Обзор рекомендуемой литературы.

Лекция 1.2 **Этапы разработки и решение задач на ЭВМ.** Формулировка и постановка задачи. Разработка алгоритма. Виды алгоритмов. Принципы программирования. Ввод и отладка программ. Решение задачи на ЭВМ и анализ полученных результатов.

Практическая работа 1.1 **Разработка алгоритма программы.** Разработка структуры программ, использующих основные типы алгоритмов.

Задание:

- Разработать структуру программы, использующей условие.
- Разработать структуру программы, использующей цикл на фиксированное число шагов.
- Разработать структуру программы, использующей итерацию.

Лекция 1.3 **Арифметические и логические основы программирования.** Системы счисления, перевод из одной системы счисления в другую. Выполнение арифметических операций в различных системах счисления. Формы представления чисел в ПЭВМ. Элементарные функции алгебры логики.

Практическая работа 1.2 **Системы счисления.** Основные типы систем счисления. Взаимосвязь десятичной, двоичной, восьмеричной и шестнадцатеричной систем счисления. Правила перевода чисел из одной системы счисления в другую.

Задание:

- Перевести число из двоичной системы в десятичную.
- Перевести число из десятичную системы в двоичной.
- Перевести число из двоичной системы в шестнадцатеричную.

Практическая работа 1.3 **Логические основы ЭВМ.** Основные функции булевой алгебры. Обозначение функций булевой алгебры. Основные законы.

Задание:

- Посчитать логическую операцию **ИЛИ** двух чисел в двоичном представлении.
- Посчитать логическую операцию **И** двух чисел в двоичном представлении.
- Посчитать логическую операцию **Исключающее ИЛИ** двух чисел в двоичном представлении.

Лекция 1.4 **Проектирование программного обеспечения.** Основные фазы проектирования. Структура жизненного цикла программного обеспечения. Модели жизненного цикла. Особенности, достоинства и недостатки каскадной модели. Особенности, достоинства и недостатки спиральной модели. Критерии оценки качества программного обеспечения.

Лекция 1.5 **Стандарты на изготовление программного обеспечения.** Основные мировые и отечественный стандарты. Отечественный стандарт ГОСТ 19 ЕСПД, основные положения, особенности, достоинства, недостатки. ГОСТ 34, основные положения, особенности, достоинства, недостатки. Международный стандарт ISO/IEC 12207, основные положения, особенности, достоинства, недостатки.

Практическая работа 1.4 **Принципы разработки программной документации.** Использование стандарта ГОСТ 19 ЕСПД. Использование стандарта ГОСТ 34. Использование стандарта ISO/IEC 12207.

Задание:

- Рассмотреть основные документы, создаваемые при разработке программной документации в соответствии со стандартом ГОСТ 19 ЕСПД.
- Рассмотреть основные документы, создаваемые при разработке программной документации в соответствии со стандартом ГОСТ 34.
- Рассмотреть основные документы, создаваемые при разработке программной документации в соответствии со стандартом ISO/IEC 12207.

**СРС по теме 1.** Проработка теоретического материала по следующим темам:

1. Этапы разработки и решение задач на ЭВМ [1,2,7].
2. Арифметические и логические основы программирования [1,2].
3. Проектирование программного обеспечения [1].
4. Стандарты на изготовление программного обеспечения [1].

**Тема 2:** Основы программирования на языке C++

Лекция 2.1 **Базовые элементы языка C++.** История возникновения языка C++. Общий синтаксис. Разделители. Использование комментариев. Структура идентификаторов. Зарезервированные слова. Препроцессор. Основные типы данных. Размер данных. Определение локальных и глобальных переменных. Операции языка: арифметические, присваивания, отношения, логические, побитовые, адресные и др. типы. Старшинство операций.

Практическая работа 2.1 **Структура программы.** Состав основных файлов программы. Включение препроцессора. Главная функция программы. Разработка элементарной программы на языке C+.

Лабораторная работа 2.1 **Разработка программы, осуществляющей ввод информации с клавиатуры и вывод на экран.** Разработка программ, использующих ввод данных с клавиатуры, вычисление простых математических выражений в соответствии с заданием и вывод результата на экран.

Цель работы: изучение принципов разработки программ, использующих ввод данных с клавиатуры, вычисление математических выражений в соответствии с заданием и вывод результата на экран.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №1, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.2 **Операторы языка C++**. Составной оператор. Оператор - выражение. Операторы условия, цикла. Оператор переключения. Другие типы операторов.

Практическая работа 2.2 **Использование операторов**. Разработка программы, включающей операторы проверки условия, цикла, оператор-переключатель.

Лабораторная работа 2.2 **Разработка программы, использующей оператор проверки условия и оператор переключатель**. Разработка программ, использующих операторы проверки условия и оператор переключатель.

Цель работы: изучение принципов разработки программы, использующей операторы проверки условия и оператор переключатель.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №2, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лабораторная работа 2.3 **Разработка программы, использующей операторы циклов**. Разработка программ, использующих операторы цикла.

Цель работы: изучение принципов разработки программы, использующей операторы цикла.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №2, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.3 **Указатели, массивы и функции**. Указатели и адреса. Описание массива. Массивы в качестве параметров. Строки. Массивы указателей. Определение функции. Вызов функции. Использование и определение параметров. Передача значений через список параметров. Функция main.

Практическая работа 2.3 **Включение в программу указателей**. Разработка программы, использующей адресную структуру. Поиск элемента по адресу. Обращение через указатели.

Практическая работа 2.4 **Использование массивов**. Включение массивов данных в программу. Одномерные и многомерные массивы. Обращению к массиву с помощью указателя.

Лабораторная работа 2.4 **Разработка программы, рассчитывающей элементы массива**. Разработка программ, использующих массивы.

Цель работы: изучение принципов разработки программы, использующей массивы.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №3, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Практическая работа 2.5 **Использование функций**. Использование функций в программе. Внешнее описание функций. Параметры, передаваемые функциям. Использование функции main.

Лабораторная работа 2.5 **Разработка программы, использующей внешние функции**. Разработка программ с внешними функциями.

Цель работы: изучение принципов разработки программы, использующей внешние функции.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №4, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.4 **Библиотеки ввода-вывода.** Подключение библиотек. Функции ввода-вывода. Описание параметров функций. Ввод символов с клавиатуры. Ввод-вывод в файл, доступ к файлам. Форматы записи данных. Ввод-вывод в строку.

Практическая работа 2.6 **Функции ввода-вывода.** Использование функций ввода-вывода. Форматы ввода-вывода. Ввод-вывод в файл.

Лабораторная работа 2.6 **Разработка программы, осуществляющей запись/чтение данных из файла.** Разработка программы, использующей различные форматы записи информации на диск.

Цель работы: изучение принципов разработки программы, использующей ввод-вывод в файл.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №5, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.5 **Другие библиотеки.** Математические функции. Функции работы с данными с плавающей точкой. Функции работы с символами и строками. Функции общего назначения. Функции выделения памяти. Функции проверки ошибок. Функции генерации случайной последовательности. Функции для работы с прерываниями БИОС. Функции дат и времени. Функции восстановления контекста программы. Функции поддержки интернациональной среды.

Практическая работа 2.7 **Функции работы с символами и строками.** Использование функций работы с символами и строками. Поиск символа в строке. Преобразование символов и строк. Определение длины строки.

Лабораторная работа 2.7 **Разработка программы, использующей функции для обработки символов и строк.** Разработка программ, использующих функции для работы с символами и строками.

Цель работы: изучение принципов разработки программы, использующей работу с символами и строками.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №6, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Практическая работа 2.8 **Выделение памяти.** Функции выделения памяти. Освобождение памяти. Выделение памяти под массив.

Лабораторная работа 2.8 **Разработка программы, использующей технологии динамического выделения памяти.** Разработка программ, использующих функции для динамического выделения памяти.

Цель работы: изучение принципов разработки программы, динамически выделяющей память.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №7, Ч.1.



Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Практическая работа 2.9 **Генерация случайной последовательности с использованием функций времени.** Функции для генерации случайной последовательности. Функции дат и времени.

Лабораторная работа 2.10 **Разработка программы, с использованием функций времени и генерации псевдослучайной последовательности.** Разработка программы, использующей функции времени и генерации случайной последовательности.

Цель работы: изучение принципов разработки программы, использующей генератор случайных чисел и функции времени.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №8, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Практическая работа 2.10 **Алгоритмы сортировки и поиска.** Алгоритмы и примеры программного кода, осуществляющего поиск в массиве. Алгоритмы и примеры программного кода, осуществляющего сортировку массива.

Лабораторная работа 2.10 **Разработка программы, использующей сортировку и поиск в массиве.** Разработка программы, использующей сортировку и поиск в массиве.

Цель работы: изучение принципов разработки программы, использующей сортировку и поиск в массиве.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №9, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.6 **Структуры и объединения.** Общие положения. Доступ к элементам структуры. Объединения. Битовые поля структур и объединений.

Практическая работа 2.11 **Структуры и объединения.** Создание программы с использованием структур и объединений. Использование внешних функций для работы с полями структур. Создание объектов и указателей на объекты структур.

Лабораторная работа 2.11 **Разработка программы, использующей структуры.** Разработка программ, использующих структуры.

Цель работы: изучение принципов разработки программы, использующей структуры.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №10, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.7 **Динамические структуры данных.** Основные виды динамических структур: стек, очередь, дэж, список, бинарное дерево. Основные особенности, принципы реализации на языке C++.

Лабораторная работа 2.12 **Разработка программы, использующей динамически расширяемые массивы.** Разработка программ, использующих динамически расширяемый массив.

Цель работы: изучение принципов разработки программы, использующей динамически расширяемый массив.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №11, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Практическая работа 2.12 **Принципы создания динамических структур данных – стек и очередь.** Стеки: особенности построения, примеры реализаций на языке C++. Очереди: особенности построения, примеры реализаций на языке C++.

Лабораторная работа 2.13 **Разработка программы, использующей стек и очередь.** Разработка программ, использующих динамические структуры стек и очередь.

Цель работы: изучение принципов разработки программы, использующей динамическую динамические структуры стек и очередь.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №12, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Практическая работа 2.13 **Принципы создания динамической структуры данных – список.** Списки: особенности построения, примеры реализаций на языке Си.

Лабораторная работа 2.14 **Разработка программы, использующей список.** Разработка программ, использующих динамическую структуру список.

Цель работы: изучение принципов разработки программы, использующей динамическую структуру список.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №13, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Практическая работа 2.14 **Принципы создания динамической структуры данных – бинарное дерево.** Бинарные деревья: особенности построения, примеры реализаций на языке C++.

Лабораторная работа 2.15 **Разработка программы, использующей бинарное дерево.** Разработка программ, использующих динамическую структуру бинарное дерево.

Цель работы: изучение принципов разработки программы, использующей динамическую структуру бинарное дерево.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №14, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.8 **Основы построения рекурсивных алгоритмов.** Общие принципы рекурсии. Принципы разработки рекурсивных функций на языке C++.

Практическая работа 2.15 **Принципы создания программ с рекурсивными функциями.** Пример для вычисления факториала. Многократный вывод на экран. Обход бинарного дерева.

Лабораторная работа 2.16 **Разработка программы, использующей рекурсивные функции.** Разработка программ, использующих рекурсивные функции.

Цель работы: изучение принципов разработки программы, использующей рекурсивную функцию.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №15, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Практическая работа 2.16 **Знакомство с общими принципами построения классов.** Использование классов в программе. Добавление полей и методов в классы. Права доступа к элементам класса.

Лабораторная работа 2.17 **Разработка программы, использующей элементарные классы.** Разработка программы, использующей элементарные классы.

Цель работы: изучение принципов разработки программы, использующей элементарные классы.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №16, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Практическая работа 2.17 **Принципы отладки программного кода в C++.** Запуск программы Debugger. Установка точки останова. Пошаговое выполнение программы.

Лабораторная работа 2.18 **Отладка программ с помощью встроенных средств отладчика.** Разработка программы для изучения принципов отладки программ в C++.

Цель работы: изучение принципов отладки программ в C++.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №17, Ч.1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

**СРС по теме 2.** Проработка теоретического материала по следующим темам:

1. Базовые элементы языка C++ [1, 3-8].
2. Операции, выражения и операторы [1, 3-8].
3. Указатели, массивы и функции [1, 3-8].
4. Библиотеки ввода-вывода [1, 3].
5. Другие библиотеки [1, 3].
6. Структуры и объединения [1, 4-8].
7. Динамические структуры данных [1, 7].
8. Основы построения рекурсивных алгоритмов [1, 7].
9. Разработка многопрограммного проекта [1].
10. Отладка программного кода [1].

**Тема 3:** Принципы разработки программ с использованием средств визуального программирования в среде разработчика Visual C++

Лекция 3.1 **Разработка проектов на основе классов MFC.** Изменение установок проекта. Иерархия классов MFC. Функции MFC. Обработка сообщений в MFC. Анализ классов проекта. Создание ресурсов. Типы диалогов. Введение элементов в диалоговое окно. Виды применяемых элементов. Управление мышью. Поддержка вывода на принтер. Запись в файл.

Практическая работа 3.1 **Принципы разработки программы на основе классов MFC.** Разработка простой оконной программы на основе классов MFC.

Лабораторная работа 3.1 **Разработка программы с выводом текста на экран на основе классов MFC.** Разработка программы с выводом текста на экран.

Цель работы: изучение принципов разработки программы с выводом текста на экран на основе классов MFC.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №1, Ч.2.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 3.2 **Разработка проекта с однодокументным интерфейсом.** Создание проекта с однодокументным интерфейсом на основе MFC. Основные классы и методы проекта. Введение в проект возможности обработки данных с клавиатуры, применение мыши. Документирование данных на диске. Введение ресурсов в приложение на базе однодокументного интерфейса. Добавление элементов меню. Создание подменю. Введение дополнительных кнопок. Использование строки подсказок. Связывание ресурсных элементов с кодом программы.

Практическая работа 3.2 **Принципы разработки программы с однодокументным интерфейсом.** Разработка программы на основе однодокументного интерфейса.

Лабораторная работа 3.2 **Разработка программы с функционалом текстового редактора.** Разработка программы-прототипа текстового редактора.

Цель работы: изучение принципов разработки текстового редактора в среде Visual C++.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №2, Ч.2.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лабораторная работа 3.3 **Включение ресурсов в приложение.** Разработка программы-прототипа с различными ресурсами.

Цель работы: изучение принципов включения ресурсов в разрабатываемое приложение.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №3, Ч.2.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 3.3 **Разработка проекта с применением графических функций.** Использование графических функций в приложении на базе однодокументного интерфейса. Рисование линий, прямоугольников, эллипсов. Выбор цвета для рисования. Закраска фона. Сохранение рисунка на диске.

Лабораторная работа 3.4 **Использование графических функций в приложениях с интерфейсом SDI.** Разработка программы-прототипа с использованием графических функций.

Цель работы: изучение принципов использования графических функций в приложениях с интерфейсом SDI.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №4, Ч.2.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 3.4 **Разработка проекта на базе диалогового окна.** Создание проекта на базе диалогового окна. Основные классы проекта. Добавление в проект кнопок и переключателей. Добавление списков. Добавление ползунков, индикаторов и др. элементов.

Практическая работа 3.3 **Принципы разработки программы на базе диалогового окна.** Разработка программы на базе диалогового окна. Добавление элементов с панели. Создание переменных и методов-обработчиков, связанных с данными элементами.

Лабораторная работа 3.5 **Создание приложения на базе диалогового окна.** Разработка программы на базе диалогового окна.

Цель работы: изучение принципов создания приложения на базе диалогового окна.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №5, Ч.2.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 3.5 **Разработка проекта с многодокументным интерфейсом.** Создание проекта с многодокументным интерфейсом на основе MFC. Основные классы и методы проекта.

Практическая работа 3.4 **Принципы разработки программы с многодокументным интерфейсом.** Разработка программы с многодокументным интерфейсом. Введение в проект возможности работы с полосой прокрутки. Разработка многодокументного приложения на основе класса CEditView.

Лабораторная работа 3.6 **Создание приложения с интерфейсом MDI.** Разработка программы на базе интерфейса MDI.

Цель работы: изучение принципов создания приложения на базе интерфейса MDI.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №6, Ч.2.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 3.6 **Разработка проекта на основе ActiveX.** Создание проекта на основе ActiveX. Принципы создания элементов с нестандартной прорисовкой. Основные классы и методы проекта. Использование в приложении внешних элементов ActiveX.

Практическая работа 3.5 **Принципы разработки программы на основе ActiveX.** Разработка программы на основе ActiveX. Введение в проект возможности по работе с мышью. Принципы использования в приложении элемента ActiveX на примере элемента “Web browser” для разработки элементарного Web-браузера.

Лабораторная работа 3.7 **Создание приложения на основе ActiveX.** Разработка программы на базе ActiveX.

Цель работы: изучение принципов создания приложения на основе ActiveX.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №7, Ч.2.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лабораторная работа 3.8 **Создание приложения для работы с сетью Internet.** Разработка программы для работы с сетью Internet.

Цель работы: изучение принципов создания приложения для работы с сетью Internet.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №8, Ч.2.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Практическая работа 3.6 **Отладка программ с помощью встроенных средств отладчика.** Разработка программы для изучения принципов отладки программ на основе MFC.

Лабораторная работа 3.9 **Отладка программ с помощью встроенных средств отладчика.** Разработка программы для изучения принципов отладки программ на основе MFC (4 часа).

Цель работы: изучение принципов отладки программ на основе MFC.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №10, Ч.2.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

**СРС по теме 3.** Проработка теоретического материала по следующим темам:

1. Разработка проектов на основе MFC [1, 9-13].
2. Разработка проекта с однодокументным интерфейсом [10, 11, 13].
3. Введение ресурсов в приложение [10, 11, 13].
4. Разработка проекта на базе диалогового окна [10, 11, 13].
5. Разработка проекта с многодокументным интерфейсом [10, 11, 13].
6. Разработка проекта на основе ActiveX [10, 11, 13].

**Тема 4:** Основы разработки объектно-ориентированного программного кода

Лекция 4.1 **Объектно-ориентированные языки программирования.** История возникновения объектно-ориентированного программирования. Факторы, обусловившие появление и содержание концепции ООП. Основные идеи ООП: использование объекта в качестве основной компоненты программы и децентрализация управления, реализуемое представлением программы как описания взаимодействия объектов. Инкапсуляция,

наследование, полиморфизм. Объектно-ориентированный подход к разработке программ. Место и роль ООП в теории и практике разработки программных систем. Языки Delphi, C++, Java - базовые основы, сходство и различия. Обзор литературы по объектно-ориентированному программированию.

Лекция 4.2 **Объекты и классы в языке C++**. Общие положения. Понятие и определение класса. Члены классов. Методы классов. Доступ к элементам класса. Друзья классов. Статические члены объектов класса. Примеры описания и использования классов.

Лабораторная работа 4.1 **Разработка программы, использующей общие принципы построения классов**. Разработка программы, использующей элементарные классы. (4 часа).

Цель работы: изучение принципов разработки программы, использующей элементарные классы.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №1, Ч.3.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 4.3 **Конструкторы и деструкторы**.

Принципы создания и уничтожения объектов. Свойства конструкторов и деструкторов. Конструкторы по умолчанию, с параметрами и копирования. Выделение и уничтожение динамических объектов класса.

Лабораторная работа 4.2 **Разработка программы, использующей конструкторы и деструкторы**. Разработка программы, использующей конструкторы и деструкторы классов.

Цель работы: изучение принципов разработки программы, использующей конструкторы и деструкторы классов.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №2, Ч.3.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 4.4 **Наследование классов**.

Базовый и производный классы. Функции-элементы и функции-друзья. Правила доступа к элементам производного класса. Иерархия классов. Одиночное и множественное наследование. Виртуальные базовые классы. Особенности доступа при множественном наследовании. Виртуальные функции. Чистые виртуальные функции и абстрактные базовые классы.

Лабораторная работа 4.3 **Разработка программы, использующей механизм наследования классов**. Разработка программы, использующей механизм наследования классов. (4 часа).

Цель работы: изучение принципов разработки программы, использующей механизм наследования классов.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №3, Ч.3.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

#### Лекция 4.5 **Шаблоны классов.**

Шаблоны классов и функций. Наследование шаблонных классов. Правила отождествления параметров шаблона. Контейнерные классы. Применение шаблонных классов для создания контейнерных классов.

Лабораторная работа 4.4 **Разработка программы, использующей шаблоны классов.** Разработка программы, использующей шаблоны классов.

Цель работы: изучение принципов разработки программы, использующей шаблоны классов.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №4, Ч.3.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

#### Лекция 4.6 **Потоковые классы ввода-вывода.**

Использование функций ввода-вывода языка C++. Форматы ввода-вывода. Стандартный ввод-вывод. Ввод-вывод в файл. Строковый ввод-вывод. Ошибочные состояния потоков ввода-вывода.

Лабораторная работа 4.5 **Разработка программы, использующей потоковые классы ввода-вывода в файл.** Разработка программы, осуществляющей ввод-вывод в файл с помощью потоков. (4 часа).

Цель работы: изучение принципов разработки программы, осуществляющей ввод-вывод в файл с помощью потоков.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №5, Ч.3.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 4.7 **Библиотека стандартных шаблонов: структура, компоненты и основные контейнеры.** Контейнеры библиотеки STL. Принцип использования контейнера string для объявления строк в языке C++. Контейнеры vector, stack, queue, list, set, map: описание, методы, принципы использования.

Лабораторная работа 4.6 **Разработка программы, использующей класс string для работы со строками.** Разработка программ, использующих класс string работы со строками.

Цель работы: изучение принципов разработки программы, использующей работу со строками с использованием класса string.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №6, Ч.3.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лабораторная работа 4.7 **Разработка программы, осуществляющей использование библиотеки STL на примере использования контейнера vector.** Разработка программ, использующих контейнер vector. (4 часа).

Цель работы: изучение принципов разработки программы, осуществляющей использование библиотеки STL на примере использования контейнера vector.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №7, Ч.3.



Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 4.8 **Библиотека стандартных шаблонов: аллокаторы, итераторы и алгоритмы.** Аллокаторы, итераторы и алгоритмы в библиотеке STL: описание, методы, принципы использования. Обсуждение примеров программного кода с использованием аллокаторов, итераторов и алгоритмов.

Лабораторная работа 4.8 **Разработка программы, осуществляющей использование контейнеров stack и queue.** Разработка программ, использующих контейнеры stack и queue. (4 часа).

Цель работы: изучение принципов разработки программы, осуществляющей использование контейнеров stack и queue.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №8, Ч.3.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лабораторная работа 4.9 **Разработка программы, осуществляющей использование контейнеров deque и list.** Разработка программ, использующих контейнеры deque и list. (4 часа).

Цель работы: изучение принципов разработки программы, осуществляющей использование контейнеров deque и list.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №9, Ч.3.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лабораторная работа 4.10 **Разработка программы, осуществляющей использование контейнеров, описанных в заголовках <set> и <map>.** Разработка программ, использующих контейнеры, описанные в заголовках <set> и <map>. (4 часа).

Цель работы: изучение принципов разработки программы, осуществляющей использование контейнеров, описанных в заголовках <set> и <map>.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №10, Ч.3.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 4.9 **Обработка исключительных ситуаций.**

Анализ различных моделей обработки исключительных ситуаций. Стандартные средства контроля подтверждений. Проверка предусловий и постусловий, вычисления инвариантов. Контроль асинхронных событий. Реализация модели обработки синхронных ситуаций с завершением в C++. Возбуждение ситуации. Описание блоков с контролем и реакций на ситуации. Система классов для описания исключительных ситуаций. (1 час).

Лабораторная работа 4.11 **Разработка программы, использующей технологии обработки исключительных ситуаций.** Разработка программы, осуществляющей обработку исключений.

Цель работы: Изучение принципов разработки программы, осуществляющей обработку исключений.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №10, Ч.3.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

**СРС по теме 4.** Проработка теоретического материала по следующим темам:

1. Основные принципы объектно-ориентированного программирования [1].
2. Объекты и классы в языке C++ [1, 3-8].
3. Поточные классы ввода-вывода [1].

## **5. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ**

В целом внеаудиторная самостоятельная работа обучающегося при изучении курса включает в себя следующие виды работ:

- ☑ проработка (изучение) материалов лекций;
- ☑ чтение и проработка рекомендованной основной и дополнительной литературы;
- ☑ подготовка к практическим занятиям и лабораторным работам;
- ☑ поиск и проработка материалов из Интернет-ресурсов, периодической печати;
- ☑ выполнение домашних заданий в форме творческих (проблемно-поисковых, групповых) заданий, докладов;
- ☑ подготовка к текущему и итоговому (промежуточная аттестация) контролю знаний по дисциплине.

Основная доля самостоятельной работы обучающихся приходится на подготовку к практическим занятиям и лабораторным работам, тематика которых полностью охватывает содержание курса. Самостоятельная работа по подготовке к практическим занятиям и лабораторным работам предполагает умение работать с первичной информацией.

## **6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ**

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине «Программирование» представлен в приложении к рабочей программе дисциплины и включает в себя:

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы;
2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания;
3. Типовые контрольные задания или материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций;
4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Фонд оценочных средств содержит:

1. Тестовые задания для контроля усвоения материала.
2. Перечень вопросов к промежуточной аттестации.

### 3. Комплект билетов к промежуточной аттестации.

#### **Примерный перечень вопросов к промежуточной аттестации**

1. История возникновения языка C++. Общий синтаксис языка C++. Разделители. Использование комментариев. Структура идентификаторов.
2. Этапы подготовки программы на языке C++ к исполнению на ПЭВМ.
3. Резервированные слова.
4. Директивы препроцессора.
5. Основные типы данных. Размер данных.
6. Определение переменных. Локальные, статические и др. типы переменных.
7. Арифметические операции. Операции присваивания.
8. Операции отношения. Логические операции. Побитовые операции.
9. Оператор-выражение. Основные виды операторов.
10. Условные операторы.
11. Операторы цикла.
12. Оператор переключения.
13. Указатели и адреса.
14. Описание массива. Массивы в качестве параметров. Строки. Массивы указателей.
15. Определение внешней функции. Вызов функции. Использование и определение параметров. Функция main.
16. Подключение динамических библиотек. Основные библиотеки языка C++.
17. Библиотеки ввода-вывода. Основные функции ввода-вывода.
18. Ввод-вывод в файл, доступ к файлам. Форматы записи данных.
19. Математическая библиотека. Основные функции.
20. Библиотеки для работы с символами и строками. Основные функции.
21. Библиотека общего назначения. Основные функции.
22. Технологии и функции для динамического выделения памяти.
23. Системные библиотеки. Основные функции.
24. Перечисления.
25. Структуры и объединения.
26. Основные виды алгоритмов, осуществляющих поиск в массиве. Алгоритм определения места включения нового элемента в упорядоченный массив.
27. Основные виды алгоритмов, осуществляющих сортировку в массиве.
28. Виды динамических структур данных. Основные особенности и различия.
29. Динамические структуры данных. Динамически расширяемые массивы. Структура, основные операции.
30. Динамические структуры данных. Списки. Структура, основные операции.
31. Динамические структуры данных. Стеки. Структура, основные операции.
32. Динамические структуры данных. Очереди. Структура, основные операции.
33. Динамические структуры данных. Бинарные деревья. Структура, основные операции.
34. Основные принципы разработки рекурсивных функций.
35. Классы: определение, описание, члены класса. Управление доступом к элементам классов, область определения элементов.
36. Конструкторы и деструкторы. Дружественные функции класса.
37. Основы механизма наследования классов.
38. Шаблоны классов.
39. Средства обработки исключений в языке C++.
40. Основные принципы отладки программного кода на языке C++.

## **7. РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА**

### **7.1. Основная литература**

1. *Маранулец Ю.В.* Язык C++. Основы программирования. (Издание второе, исправленное и дополненное, рекомендовано ДВ РУМЦ в качестве учебного пособия). Петропавловск-Камчатский: КамГУ им. Витуса Беринга, 2019. – 158 с.

2. *Страуструп Б.* Язык программирования C++. Четвертое издание. М.:БИНОМ. Лаборатория знаний, 2023. – 1216 с.

### **7.2. Дополнительная литература**

3. *Карпов Б., Баранова Т.* C++. Специальный справочник. С.-Пб.:Питер, 2001. – 479 с.
4. *Климова Л.М.* C++. Практическое программирование. М.:КУДИЦ-ОБРАЗ, 2001. – 587 с.
5. *Маранулец Ю.В.* Программирование на языке высокого уровня. Рекомендовано ДВ РУМЦ в качестве учебного пособия. Петропавловск-Камчатский: Издательство КамчатГТУ, 2008 - 189 с.
6. *Лафоре Р.* Объектно-ориентированное программирование в C++. Классика Computer Science. Четвертое издание. СПб.: Питер, 2004. – 924 с.
7. *Романов Е.Л.* Практикум по программированию на C++. С.-Пб.:БХВ-Петербург, 2004. – 427 с.
8. *Кнут Д.* Искусство программирования для ЭВМ. Ч.1. М.:Мир, 1976. – 726 с.
9. *Финогенов К.Г.* Win32. Основы программирования. М.: Диалог-МИФИ, 2004. - 416 с.
10. *Рихтер Д.* Windows. Создание эффективных Win32-приложений с учетом специфики 64-разрядной версии Windows. С.-Пб.: Питер, 2005. - 624 с.
11. *Д.Круглински, С.Уингоу, Д.Шефферд.* Программирование на Microsoft Visual C++ для профессионалов. С.-Пб.: Питер, 2004 г.
12. *Шупак Ю.А.* Win32 API. Эффективная разработка приложений. С.-Пб.: Питер, 2007.-572с.
13. *Черносвитов А.* Visual C++ и MFC. С.-Пб.: Питер, 2005 г.

## **8. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ»**

1. Электронно-библиотечная система «eLibrary»: [Электронный ресурс]. - Режим доступа: <http://www.elibrary.ru>
2. Официальный сайт компании Microsoft по изучению языка C++: [Электронный ресурс]. - Режим доступа: <https://learn.microsoft.com/ru-ru/cpp/cpp/?view=msvc-160>
3. Руководства и описание языка C++ от его создателя Бьярна Страуструпа: [Электронный ресурс]. - Режим доступа: <https://www.stroustrup.com/C++.html>
4. Руководства и описание языка C++ на интернет ресурсе для программистов Хабр: [Электронный ресурс]. - Режим доступа: <https://habr.com/ru/hub/cpp/>
5. Онлайн справочник программиста на языке C++: [Электронный ресурс]. - Режим доступа: <http://www.c-cpp.ru/>

## **9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ**

### **Подготовка к лекционным занятиям**

Лекции составляют основу теоретического обучения и дают систематизированные основы научных знаний по дисциплине, концентрируют внимание обучающихся на наиболее сложных и узловых вопросах, стимулируют их активную познавательную деятельность и способствуют формированию творческого мышления.

При подготовке к лекции следует предварительно ознакомиться с учебным материалом по теме занятия и при конспектировании лекции акцентировать внимание на новых теоретических положениях и иных данных, не нашедших отражения в учебной литературе.

Для успешного изучения дисциплины студенту рекомендуется систематически готовиться к каждому занятию по следующей схеме:

- повторить материал предыдущей лекции, используя конспекты, учебную и специальную литературу

- ответить на контрольные вопросы по изучаемой теме.

### **Подготовка к лабораторным занятиям**

Лабораторная работа – это выполнение студентами под руководством преподавателя или по инструкции заданий (решение задач, написание программ) с применением персонального компьютера.

В ходе лабораторных работ студенты воспринимают и осмысливают новый учебный материал. Лабораторные занятия носят систематический характер, регулярно следуя за лекционными занятиями. Лабораторные работы выполняются согласно графику, при этом соблюдается принцип индивидуального выполнения работ.

Обучающийся должен подготовить отчет к каждой лабораторной работе, предусмотренной планом.

При подготовке к лабораторным занятиям необходимо заранее изучить методические рекомендации по его проведению. Обратит внимание на цель занятия, на основные вопросы для подготовки к занятию, на содержание темы занятия.

### **Перечень методических указаний к проведению учебных, лабораторных занятий и самостоятельной работы студентов**

1. Марапулец Ю.В. Программирование. Методическое пособие к выполнению лабораторных работ по направлению подготовки бакалавров 09.03.01 «Информатика и вычислительная техника» по дисциплине «Программирование» для студентов очной формы обучения, Часть 1 – Петропавловск-Камчатский: КамчатГТУ, 2015. – 78 с. (электронная форма).
2. Марапулец Ю.В. Программирование. Методическое пособие к выполнению лабораторных работ по направлению подготовки бакалавров 09.03.01 «Информатика и вычислительная техника» по дисциплине «Программирование» для студентов очной формы обучения, Часть 2 – Петропавловск-Камчатский: КамчатГТУ, 2015. – 71 с. (электронная форма).
3. Марапулец Ю.В. Программирование. Методическое пособие к выполнению лабораторных работ по направлению подготовки бакалавров 09.03.01 «Информатика и вычислительная техника» по дисциплине «Программирование» для студентов очной формы обучения, Часть 3 – Петропавловск-Камчатский: КамчатГТУ, 2023. – 81 с. (электронная форма).

### **10. КУРСОВОЙ ПРОЕКТ (РАБОТА)**

Рекомендуется следующая тематика курсовых проектов:

1. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Круг и порожденного класса Кольцо.

2. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Круг и порожденного класса Дуга окружности.
3. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Прямоугольник и порожденного класса Прямоугольник с вписанным кругом.
4. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Треугольник и порожденного класса Четырехугольник.
5. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Квадрат и порожденного класса Квадрат с описанным кругом.
6. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Прямоугольник и порожденного класса Прямоугольник со скругленными углами.
7. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Вектор на плоскости и порожденного класса Вектор и точка на плоскости.
8. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Вектор на плоскости и порожденного класса Два вектора с общим началом с определением координаты вектора суммы двух векторов.
9. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Вектор на плоскости и порожденного класса Два вектора с общим началом с определением координаты вектора разности двух векторов.
10. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Вектор на плоскости и порожденного класса Два параллельных вектора на плоскости.
11. Реализация возможностей объектно-ориентированного подхода с использованием структуры Комплексное число и класса Произведение комплексных чисел.
12. Реализация возможностей объектно-ориентированного подхода с использованием класса Время и дочернего класса Расписание движения поездов.

## **11. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ, ВКЛЮЧАЯ ПЕРЕЧЕНЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННО-СПРАВОЧНЫХ СИСТЕМ**

### ***11.1 Перечень информационных технологий, используемых при осуществлении образовательного процесса:***

- электронные образовательные ресурсы, представленные в п. 8 рабочей программы;
- использование слайд-презентаций;
- интерактивное общение с обучающимися и консультирование в электронной информационной образовательной среде.

### ***11.2 Перечень программного обеспечения, используемого при осуществлении образовательного процесса***

При освоении дисциплины используется лицензионное программное обеспечение:

- операционные системы AstraLinux (или иная операционная система, включенная в реестр отечественного программного обеспечения);
- комплект офисных программ Р-7 Офис (в составе текстового процессора, программы работы с электронными таблицами, программные средства редактирования и демонстрации презентаций);
- компилятор языка С++;
- программа проверки текстов на предмет заимствования «Антиплагиат».

## **12. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

Лекционные занятия по данной дисциплине проводятся в учебной аудитории 7-518 с комплектом учебной мебели. Лабораторные и практические занятия проводятся в лаборатории разработки программного обеспечения (учебная аудитория 7-513), оборудованной 10 рабочими станциями с программным обеспечением, представленным в п. 11.2, доступом к сети «Интернет», в электронную информационно-образовательную среду и комплектом учебной мебели.

### 13 ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ В РАБОЧЕЙ ПРОГРАММЕ

Дополнения и изменения в рабочей программе за \_\_\_\_/\_\_\_\_ учебный год

В рабочую программу \_\_\_\_\_  
(наименование дисциплины)

для специальности (тей) \_\_\_\_\_  
(номер специальности)

вносятся следующие дополнения и изменения:

Дополнения и изменения внес \_\_\_\_\_  
(должность, Ф.И.О., подпись)

Рабочая программа пересмотрена и одобрена на заседании кафедры  
\_\_\_\_\_  
(наименование кафедры)

« \_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

Заведующий кафедрой \_\_\_\_\_ (подпись) \_\_\_\_\_ (Ф.И.О.)