



Автономная некоммерческая образовательная организация высшего образования

Международный институт компьютерных технологий

Кафедра Информатики и вычислительной техники

УТВЕРЖДАЮ

Декан факультета *высшего образования*

\_\_\_\_\_ *Хоряков О.С.*

«23» января 2026 г.

## РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

*Б1.В.ДВ.03.02 «Программирование на C/C++»*

Рекомендуется для направления подготовки (специальности) 54.03.01 «Дизайн»

Профиль подготовки / специализации: Дизайн программных интерфейсов и Веб-приложений

Квалификация (степень) выпускника: бакалавр

Факультет высшего образования

Наименование факультета или факультетов

Воронеж  
2026

Рабочая программа составлена на основании требований:

- Федерального государственного образовательного стандарта высшего образования № 1015, утвержденного Министерством образования РФ «13» августа 2020 г.
- учебного плана МИКТ по направлению/специальности 54.03.01 «Дизайн», одобренного Учёным советом вуза 23.01.2026, протокол №4.

Индекс- 54.03.01 Д

Рецензент: доцент кафедры «Автоматизированных и вычислительных систем» Воронежского государственного технического университета, канд. техн. наук, доцент  
А. М. Нужный

Составитель (составители):

канд. техн. наук, доцент

\_\_\_\_\_ С.А. Мишин

Рабочая программа обсуждена на заседании кафедры «Информатики и вычислительной техники» «10» января 2026 г., протокол № 6

Рабочая программа одобрена методическим советом МИКТ

«21» января 2026 г., протокол № 4

## **1. Цель и задачи учебной дисциплины:**

Цель дисциплины "Программирование на C/C++" состоит в получении теоретических знаний, умений и практических навыков в области разработки программ на языках C и C++ на современной технологической основе.

Для достижения поставленной цели студенты должны получить знания по программированию на языках C/C++ в системах программирования для решения различного типа задач.

Задачами учебной дисциплины являются:

- сформировать у студента представление о роли и месте языков программирования C и C++, типовых алгоритмических решениях, технологиях программирования в современных инструментальных средах разработки компонентов программно-аппаратных комплексов;
- научить студента строить на C и C++ программные приложения, используя различные структуры данных, в том числе, базы данных, шаблоны приложений и стандартные библиотеки классов.

## **2. Место учебной дисциплины в структуре ООП:**

Дисциплина относится к разделу, формируемому участниками образовательных отношений *Б1.В.ДВ.03.02*.

Для усвоения дисциплины «Программирование на C/C++» необходимо знание дисциплин «Информатика», «Алгоритмизация и программирование», «Объектно-ориентированное программирование».

В результате освоения данных дисциплин студент должен:

1. Знать:

- приемы работы с современными программными оболочками и операционными системами;
- типовые алгоритмы, способы и методы построения алгоритмов;
- основные языки программирования и принципы объектно-ориентированного проектирования и программирования.

2. Уметь:

- работать с разными структурами данных, в том числе и базами данных
- применять языки программирования для автоматизации бизнес-процессов, решения различных практических задач;
- разрабатывать классы, интерфейс пользователя.

3. Владеть:

- навыками программирования в современных системах разработки программ;
- навыками отладки и тестирования программно-технических комплексов;
- способностью совершенствования навыков в разработке программ.

Дисциплина предшествует изучению дисциплины (дисциплин) «Базы данных», «Моделирование», «Теория и технологии поиска эффективных решений», выполнению курсовых проектов, выпускной квалификационной работы и занятиям самостоятельной профессиональной деятельностью.

## **3. Перечень планируемых результатов обучения по дисциплине**

### **3.1. Формируемые дисциплиной УК и ОПК в соответствии с ФГОС**

Рабочим учебным планом формирование УК и ОПК не предусмотрены.

### **3.2. Профессиональные компетенции выпускников, формируемые дисциплиной и планируемые результаты обучения по дисциплине**

Таблица 3.2

Основание (профстандарты, справочник, анализ опыта)	Код и содержание профессиональной компетенции	Планируемые результаты обучения по дисциплине
Профессиональный стандарт 06.019 «ТЕХНИЧЕСКИЙ ПИСАТЕЛЬ (СПЕЦИАЛИСТ ПО ТЕХНИЧЕСКОЙ ДОКУМЕНТАЦИИ В ОБЛАСТИ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ)» (утвержден приказом 03 октября 2022 № 609н, регистрационный № 197)	<b>ПК-5.</b> Способен разрабатывать требования и проектировать графические программные интерфейсы, интерфейсы веб-приложений и интернет-сайтов, обеспечивающие удобство использования, высокую интерактивность и соответствие требованиям дизайна	<b>Знать:</b> современные системы программирования для C/C++, синтаксис языка C/C++ и технологии построения программных приложений, назначение стандартных библиотек классов и шаблонов приложений. <b>Уметь:</b> проводить анализ предметной области, строить различные виды приложений в системе программирования Microsoft Visual Studio, применять библиотеки классов для быстрого создания эффективного программного обеспечения на C/C++. <b>Владеть:</b> способностью разрабатывать на C/C++, документировать компоненты программного обеспечения обработки различных структур данных, в том числе баз данных, с использованием современных технологий и систем программирования

#### 4. Объем дисциплины и виды учебной работы

Таблица 4

Распределение трудоемкости, час	Форма обучения	
	очная	заочная
<b>1. Контактная работа по видам учебных занятий:</b>		
Лекционные	36	8
Практические	-	-
Лабораторные работы	36	12
<b>2. Самостоятельная работа</b>	27	115
<b>3. Курсовой проект</b>	-	-
<b>4. Экзамен</b>	45	9
<b>Общая трудоемкость</b>	<b>144</b>	<b>144</b>

## 5. Содержание дисциплины и распределение трудоемкости по видам занятий

Таблица 5.1. Очная форма обучения

№	Наименование модуля (раздела) дисциплины	Лекции	Лаб	Практ	СРС	Всего
1	Принципы построения программных приложений на С#	2	2	–	1	5
2	Классы, стандартные библиотеки классов	18	16	–	14	48
3	Технологии разработки программных приложений на С/С++. Шаблоны приложений	16	18	–	12	46
<b>Итого по дисциплине</b>		<b>36</b>	<b>36</b>	<b>–</b>	<b>27</b>	<b>99</b>
<b>Контроль</b>						<b>45</b>
<b>Всего</b>		<b>36</b>	<b>36</b>	<b>–</b>	<b>27</b>	<b>144</b>

Таблица 5.2. Заочная форма обучения

№	Наименование модуля (раздела) дисциплины	Лекции	Лаб	Практ	СРС	Всего
1	Принципы построения программных приложений на С#	1	–	–	18	19
2	Классы, стандартные библиотеки классов	4	6	–	50	60
3	Технологии разработки программных приложений на С/С++. Шаблоны приложений	3	6	–	47	56
<b>Итого по дисциплине</b>		<b>8</b>	<b>12</b>	<b>–</b>	<b>115</b>	<b>135</b>
<b>Контроль</b>						<b>9</b>
<b>Всего</b>		<b>8</b>	<b>12</b>	<b>–</b>	<b>115</b>	<b>144</b>

## 6. Тематический план аудиторных занятий для дневной формы обучения

Содержание занятия	Вид занятия	Трудоемкость в часах	Формируемые компетенции
<b>Модуль 1 – Принципы построения программных приложений на С#</b>		<b>4</b>	<b>ПК-5</b>
<i>Тема 1: Особенности С/С++</i>			
1. Особенности С и С++. История, область применения, особенности языков, стандарты	Лекция	0,5	ПК-5
<i>Тема 2: Операции и операторы</i>			
2. Унарные, бинарные, тернарная операции. Приоритет операций. Операторы:	Лекция	0,5	ПК-5

присваивания, условный, цикла, выбора, прерывания, продолжения цикла, безусловного перехода. Приемы применения			
<b>Тема 3: Типы данных C/C++</b>			
3. Классификация, явное и неявное преобразование типов. Операции преобразования типов. Указатели.	Лекция лаб. раб.	0,5 1	ПК-5
<b>Тема 4. Функции</b>			
4. Формальные и фактические параметры, тип возвращаемого значения. Способы передачи параметров. Функция main(). Функции ввода/вывода	Лекция лаб. раб.	0,5 1	ПК-5
<b>Модуль 2 – Классы, стандартные библиотеки классов</b>		<b>34</b>	<b>ПК-5</b>
<b>Тема 1 – Классы. Содержимое классов</b>			
1. Поля, методы и другие члены класса. Доступ к членам класса. Конструкторы, деструкторы, создание и уничтожение объектов. Статические члены класса. Виды классов	Лекция, лаб. раб.	2 2	ПК-5
2. Отношения между классами. Доступ при наследовании. Проблема множественного наследования. Иерархия стандартной библиотеки классов	Лекция, лаб. раб.	2 2	ПК-5
3. Механизмы реализации полиморфизма. Перегрузка, переопределение, сокрытие. Сигнатура метода. Виртуальные методы и классы. Раннее и позднее связывание	Лекция , лаб. раб.	2 2	ПК-5
<b>Тема 2: Универсализация</b>			
4. Механизм универсализации, область его действия. Шаблоны	Лекция , лаб. раб.	2 2	ПК-5
<b>Тема 3. Стандартная библиотека классов.</b>			
5. Библиотека STL: итераторы и контейнеры. Классы ввода/вывода. Поточный ввод/вывод	Лекция , лаб. раб.	2 2	ПК-5
<b>Тема 4. Исключения</b>			
6. Обработка исключений в программе	Лекция , лаб. раб.	2 2	ПК-5
<b>Тема 5. Массивы и строки</b>			
7. Массивы: размерность, размер, размещение в памяти. Технологии работы со строками	Лекция , лаб. раб.	2 2	ПК-5
<b>Тема 6. Лямбда-выражения</b>			
8. Лямбда-выражения. Анонимные методы, лямбда-выражения. События	Лекция , лаб. раб.	4 2	ПК-5
<b>Модуль 3 – Технологии разработки программных приложений на C/C++. Шаблоны приложений</b>		<b>34</b>	<b>ПК-5</b>
<b>Тема 1. Системы программирования для C/C++</b>			
1. Обзор, сравнительная характеристика, область применения. Система Microsoft Visual Studio. Платформа .NET. Проект, решение, сборка, модуль. Типы приложений на C#. Технологии проектирования,	Лекция , лаб. раб.	4 4	ПК-5

построения, отладки программных приложений			
2. Шаблоны Windows-приложений. Библиотека компонентов для Windows-приложений. Графические библиотеки	Лекция , лаб. раб.	4 6	ПК-5
3. Шаблоны Web-приложений. Библиотека компонентов для Web-приложений. Компоненты для работы с базами данных	Лекция , лаб. раб.	4 4	ПК-5
4. Шаблоны настольных приложений. Технологии WPF. Возможности, преимущества. Язык XAML. Изменение поведения элемента управления	Лекция , лаб. раб.	4 4	ПК-5

## 7. Тематический план аудиторных занятий для заочной формы обучения

Содержание занятия	Вид занятия	Трудоемкость в часах	Формируемые компетенции
<b>Модуль 1 – Принципы построения программных приложений на C#</b>		<b>1</b>	<b>ПК-5</b>
<i>Тема: Особенности C/C++. Операции и операторы.</i>			
1. Область применения, история, особенности языков C и C++, стандарты. Операции, операторы, типы данных. Указатели. Формальные и фактические параметры, тип возвращаемого значения функции. Способы передачи параметров. Функция main(). Функции ввода/вывода	лекция	1	ПК-5
<b>Модуль 2 – Классы, стандартные библиотеки классов</b>		<b>10</b>	<b>ПК-5</b>
<i>Тема: Классы. Содержимое классов</i>			
1. Поля, свойства, методы и другие члены класса. Доступ к членам класса. Конструкторы, деструкторы, создание и уничтожение объектов. Статические члены класса.	Лекция	1	ПК-5
2. Наследование. Проблема множественного наследования. Полиморфизм. Перегрузка, переопределение, сокрытие. Сигнатура метода. Раннее и позднее связывание	Лекция лаб. раб.	1 2	ПК-5
3. Универсализм. Шаблоны. Библиотека STL. Массивы и строки. Обработка исключений	Лекция лаб. раб.	1 2	ПК-5
4. Лямбда-выражения	Лекция лаб. раб.	1 2	ПК-5
<b>Модуль 3 – Технологии разработки программных приложений на C/C++. Шаблоны приложений</b>		<b>9</b>	<b>ПК-5</b>
<i>Тема 1. Системы программирования для C/C++</i>			
1. Обзор, сравнительная характеристика, область применения. Система Microsoft Visual Studio. Платформа .NET. Проект, решение, сборка, модуль. Типы приложений на C#.	Лекция	0,5	ПК-5

Технологии проектирования, построения, отладки программных приложений			
2. Шаблоны Windows-приложений. Библиотека компонентов для Windows-приложений. Графические библиотеки	Лекция, лаб. раб	0,5 2	ПК-5
3. Шаблоны Web-приложений. Библиотека компонентов для Web-приложений. Компоненты для работы с базами данных	Лекция, лаб. раб	1 2	ПК-5
4. Шаблоны настольных приложений. Технологии WPF. Возможности, преимущества. Язык XAML. Изменение поведения элемента управления.	Лекция, лаб. раб	1 2	ПК-5

**8. Примерная тематика курсовых работ, проектов (при наличии) – не предусмотрены учебным планом**

**9. Расчётно-графические задания – не предусмотрены учебным планом**

**10. Контрольные работы - не предусмотрены учебным планом**

**11. Учебно-методическое и информационное обеспечение дисциплины**

Перечень учебно-методического и информационного обеспечения учебной дисциплины представлен в Приложении 2.

**12. Материально-техническое обеспечение дисциплины**

Перечень материально-технического обеспечения учебной дисциплины представлен в Приложении 3.

**13. Методические рекомендации по организации преподавания дисциплины**

В начале аудиторных занятий желательно напомнить студентам сведения теоретического и методического характера, необходимые для успешной работы с текущим материалом и полученные ими ранее по данной дисциплине. Желательно также после изложения слушателям теоретических и методических сведений провести краткий выборочный опрос по основным положениям изложенного материала.

В отчёте по выполненной лабораторной работе должны быть кратко представлены: необходимые теоретические сведения, ответы на контрольные вопросы, результаты работы и краткие выводы.

Лекционные занятия (лекции) реализуются в традиционной форме в соответствии с календарным планом-графиком чтения лекций. Целесообразно лекции сопровождать практическими примерами для лучшего понимания материала и формирования навыков и умений для решения задач, относящихся к различным разделам «Программирование на C/C++». Следует уделить внимание вопросам эффективного использования инструментальных программных средств. В рамках данной дисциплины студенты должны ознакомиться с принципами построения программных приложений, но и научиться проектировать приложения для решения прикладных задач.

По результатам проведённых лекций и лабораторных занятий по возможности проводить контрольные работы один-два раза в семестр.

Конспекты лекций и материалы к лабораторным занятиям, как и список рекомендуемой литературы, выставлены в соответствующем разделе на официальном сайте института.

## **14. Воспитательная работа**

### **Духовно-нравственное воспитание**

- развитие способности к развитию творческого потенциала в учебно-профессиональной деятельности на основе нравственных установок и моральных норм, непрерывного самообразования и самовоспитания;

- развитие способности к сотрудничеству с окружающими в образовательной, общественно полезной, проектной и других видах деятельности.

### **Гражданско-правовое воспитание**

- развитие студенческого самоуправления, совершенствование у обучающихся организаторских умений и навыков.

### **Патриотическое воспитание**

- формирование чувств патриотизма, гражданственности, уважения к памяти защитников Отечества и подвигам Героев Отечества;

- формирование социально значимых и патриотических качеств обучающихся.

### **Экологическое воспитание**

- формирование бережного и ответственного отношения к своему здоровью (физическому и психологическому) и здоровью других людей, живой природе, окружающей среде.

### **Профессиональное воспитание**

- формирование творческого подхода к самосовершенствованию в контексте будущей профессии;

- повышение мотивации профессионального самосовершенствования обучающихся средствами изучаемых учебных дисциплин, практик, научно-исследовательской и других видов деятельности.

### **Приложения:**

Приложение 1 – Фонд оценочных средств учебной дисциплины

Приложение 2 – Учебно-методическое и информационное обеспечение дисциплины

Приложение 3 – Материально-техническое обеспечение учебной дисциплины



Автономная некоммерческая образовательная организация высшего образования

Международный институт компьютерных технологий

Кафедра Информатики и вычислительной техники

УТВЕРЖДЕН

на заседании кафедры ИВТ

« 10 » января 2026 г., протокол № 6

Заведующий кафедрой

\_\_\_\_\_ Слепокуров Ю.С.

## ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

*Б1.В.ДВ.03.01 «Программирование на C/C++»*

Рекомендуется для направления подготовки (специальности) 54.03.01 «Дизайн»

Профиль подготовки / специализации: Дизайн программных интерфейсов и Web приложений

Квалификация (степень) выпускника: бакалавр

Факультет высшего образования

Наименование факультета или факультетов

Составитель (составители):

канд. техн. наук, доцент

\_\_\_\_\_ С.А. Мишин

Экспертиза проведена доцентом кафедры «Автоматизированных и вычислительных систем» Воронежского государственного технического университета, канд. техн. наук, доцентом А.М. Нужным

Воронеж  
2026

## 1. Требования к результатам освоения дисциплины:

1.1. В результате изучения дисциплины студент должен демонстрировать следующие индикаторы достижения универсальных и (или) общепрофессиональных компетенций – универсальные и общепрофессиональные компетенции учебным планом не предусмотрены.

1.2. В результате изучения дисциплины студент должен демонстрировать следующие индикаторы достижения профессиональных компетенций:

1.2.1. ПК-5 – способен разрабатывать компоненты программно-аппаратных комплексов и информационных систем

Таблица 1.2

Профессиональный стандарт	Наименование трудовой функции	Необходимо знать	Необходимо уметь	Индикаторы достижения компетенции
Профессиональный стандарт 06.019 «ТЕХНИЧЕСКИЙ ПИСАТЕЛЬ (СПЕЦИАЛИСТ ПО ТЕХНИЧЕСКОЙ ДОКУМЕНТАЦИИ В ОБЛАСТИ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ)» (утвержден приказом 03 октября 2022 № 609Н, регистрационный № 197)	Разработка пользовательских документов, а также стандартных технических документов на основе предоставленного материала	Синтаксис и особенности языка программирования С/С++	Разрабатывать компоненты программных приложений на С/С++ и документировать их Использовать стандартные классы и шаблоны системы программирования	ПК-5.1. Способен проектировать интерфейсы веб-приложений, сайтов и графических программ с учётом требований, обеспечивая удобство, интерактивность и соответствие дизайн-стандартам
		Стандартные библиотеки, шаблоны приложений системы программирования и способы создания приложений различных типов	Использовать выбранную систему программирования для эффективного создания компонентов Windows-приложений, настольных и веб-приложений, а также их документирования	

## 2. Программа оценивания контролируемых частей компетенций

### 2.1.1 Текущая аттестация

Таблица 2.1 – ПК-5

№	Контролируемый раздел (тема)	Код компетенции	Код индикатора достижения профессиональной компетенции	Наименование оценочного средства
1	2	3	4	5
1	Принципы построения программных приложений на C/C++	ПК-5	ПК-5.1	Выполнение и защита лабораторных работ Тесты
2	Классы, стандартные библиотеки классов	ПК-5	ПК-5.1	Выполнение и защита лабораторных работ Тесты
3	Технологии разработки программных приложений на C/C++. Шаблоны приложений	ПК-5	ПК-5.1	Выполнение и защита лабораторных работ Тесты

### 2.1.2. Примеры оценочных средств для текущего контроля

#### 2.1.2.1. Примеры тестовых заданий (оценка сформированности компетенции ПК-5 на репродуктивном уровне (оценка знаний)):

1. Может ли функция возвращать значения любого типа?
  1. Да, кроме массива или другой функции
  2. Да, кроме другой функции
  3. Да.
  4. Нет.
2. Если оператор *return* отсутствует или не содержит выражения, то возвращаемое значение функции:
  1. Определено
  2. Не определено
  3. Равно нулю
  4. Будет вещественным числом
3. При определении функции, [*< спецификация типа >*] в совокупности с описателем определяет:
  1. Тип возвращаемого значения и имя функции
  2. Только тип возвращаемого значения
  3. Тип возвращаемого значения и класс памяти функции
  4. Тип возвращаемого значения и объявления параметров
4. *Формальные параметры* – это переменные, которые:
  1. Зависят от класса памяти функции
  2. Принимают значения, переданные функции при вызове
  3. Вырабатываются при выполнении оператора возврата типа *return*
  4. Задаются в теле функции
5. Могут ли формальные параметры иметь базовый тип?
  1. Да, а также могут быть структурой, объединением, указателем или массивом

2. Да и ничего более
3. Нет
4. Нет, но могут быть структурой, объединением, указателем или массивом
6. Что может содержать тело функции?
  1. Операторы, и объявления переменных, используемых в этих операторах
  2. Любые операторы, кроме операторов *switch* и *goto*
  3. Только переменные
  4. Любые переменные, кроме структур, указателей и массивов
7. Неявное объявление функции имеет место всякий раз, когда функция вызывается:
  1. Без предварительного объявления или определения
  2. Без предварительного объявления
  3. С отсутствием формальных параметров
  4. С формальными параметрами
8. При неявном объявлении функции компилятор языка C считает, что вызываемая функция имеет:
  1. Тип возвращаемого значения *int* и класс памяти *extern*
  2. Тип возвращаемого значения *int* и класс памяти *static*
  3. Тип возвращаемого значения *double* и класс памяти *static*
  4. Тип возвращаемого значения *float* и класс памяти *static*
9. Чему передается управление и фактические аргументы (если они есть) при вызове функции?
  1. Заданной функции
  2. Оператору *return*
  3. Оператору *switch* или *goto*
  4. Операционной системе
10. Что требуется для вызова функции с переменным числом аргументов?
  1. В вызове функции просто задается то число аргументов, которое нужно
  2. Требуется установка класса памяти *extern*
  3. Должен присутствовать оператор *return*
  4. Количество операторов *return* должно соответствовать числу аргументов
11. Где размещаются аргументы, заданные при вызове функции?
  1. В стеке
  2. В теле функции
  3. В параметрах функции *main*
  4. В заголовочном файле *stdio.h*
12. Ограничивает ли компилятор число рекурсивных вызовов одной функции?
  1. Нет
  2. Да, при неявном объявлении функции
  3. Да, но при использовании класса памяти *extern*
  4. Да, при наличии формальных параметров
13. Какие функции могут иметь одно и то же имя, но отличаться друга количеством или типом параметров?
  1. Перегруженные функции
  2. Дружественные функции
  3. Подставляемые функции
  4. Таких функций нет
14. Какие функции имеют доступ к *private* части другой функции, не являющейся членом этого класса?
  1. Перегруженные функции
  2. Дружественные функции
  3. Подставляемые функции
  4. Таких функций нет

15. Каким образом при вызове функции можно установить значения параметров по умолчанию?

1. Значения функции не задаются
2. Надо задать тип возвращаемого значения
3. Передается нулевое значение
4. Установить значение параметров по умолчанию нельзя

16. Какая ситуация, не препятствует созданию *inline* – кода?

1. В функции значения параметров установлены по умолчанию
2. В функции имеются циклы, операторы *switch*, *goto*
3. Функция возвращает значение более одного раза
4. Функция рекурсивная или содержит статические (*static*) переменные

**2.1.2.2 Примеры заданий для самостоятельной работы** (оценка сформированности компетенции **ПК-5** на реконструктивном уровне (оценка умений)):

1. Каково значение переменной *k* будет после выполнения операторов:

```
int j=0;
++j; int k=j++;
```

2. Сколько раз будет выполнено тело цикла:

```
while(100)
{ ... }
```

3. Есть ли ошибки в следующем фрагменте:

```
for (int i=0, j=1; i<10, j<15 )
{ ... }
```

4. Есть ли ошибки в следующем фрагменте:

```
void a(int k, double d)
{ ... }
```

```
void a()
```

```
{ ... }
```

```
void b(int k, double d)
```

```
{ ... }
```

5. Какое значение примет переменная *d* :

```
double d;
int m=3, n=2;
d=m/n;
```

6. Как получить набор равномерно распределенных случайных чисел целого типа на отрезке от -10 до 50?

7. Есть ли ошибки в следующем фрагменте:

```
class A
{ ... }
class B
{ ... }
class C : A, B
{ ... }
```

8. Создайте 2 класса: точка на плоскости, окружность на плоскости – наследник класса точка на плоскости. Включите в классы необходимые члены.

**2.1.2.3 Примеры тем реферативных сообщений** для практических занятий (оценка сформированности компетенции **ПК-5** на творческом уровне (оценка владения навыками, методиками)):

1. Построение оконных приложений, использующих графические библиотеки представления 3D объектов.

2. Построение приложений, использующих базы данных в Visual Studio.

3. Построение приложений, использующих CLR .Net Framework.
4. Межъязыковое взаимодействие в приложениях на C++ Visual Studio.
5. Технологии создания библиотек графических примитивов на C++.

### **2.1.3 Критерии и шкалы оценивания**

Тесты представляют собой программу, выдающую вопросы, выбранные из категорий случайным образом и выдающую оценку по пятибалльной системе:

- Оценка «отлично» – более 85% правильных ответов
- «хорошо» – более 70%, но менее 85% правильных ответов
- «удовлетворительно» – от 50% до 70% правильных ответа
- «неудовлетворительно» – менее 50% правильных ответов.

### **2.1.4 Оценивание защиты лабораторных работ:**

- 3 балла – неполное решение поставленной задачи (некорректные форматы вывода результатов, не выполнены все требования при разработке программного приложения, полученные результаты не полностью соответствуют заданию);
- 4 балла - правильное решение поставленной задачи с недостатками (неправильный выбор инструментов представления данных задачи, неточности в проектировании классов, отсутствуют необходимые комментарии в программе);
- 5 баллов – правильное и полное решение поставленной задачи.

## **2.2. Промежуточная аттестация**

### **2.2.1. Оценочные средства для курсового проекта (курсовой работы)**

Курсовое проектирование не предусмотрено учебным планом.

### **2.2.2 Примеры вопросов подготовки к промежуточной аттестации для оценки сформированности компетенции ПК-5 на уровне знаний**

1. Стандарты языков C/C++.
2. Основные операторы языков C/C++.
3. Типы данных языков C/C++.
4. Реализация принципов объектно-ориентированного программирования в языке C++.
5. Определение класса.
6. Преобразование типов в C/C++.
7. Обработка исключительных ситуаций.
8. Присваивание и сравнение объектов.
9. Конструкторы и деструкторы.
10. Члены классов.
11. Метод main().
12. Полиморфизм в языке C++, способы реализации.
13. Индексаторы класса и их определение.
14. Реализация инкапсуляции в классах.
15. Отношения между классами.
16. Абстрактные классы.
17. Проблемы безопасного кода.
18. Лямбда-выражения. Область применения.
19. Неявный тип.
20. Анонимные методы.
21. Стандартная библиотека классов. Основные пространства имен.
22. Файловые операции.
23. Работа с потоками ввода/вывода.

24. Реализация событий и обработки событий в языке C++.
25. Системы программирования для C/C++.
26. Особенности разработки приложений на языке C++ в Microsoft Visual Studio.
27. Основные шаблоны приложений на языке C++ в Microsoft Visual Studio.
28. Компоненты для разработки приложений на языке C++ в Microsoft Visual Studio.
29. Графические библиотеки для разработки приложений на языке C++ в Microsoft Visual Studio.
30. Компоненты баз данных для разработки приложений на языке C++ в Microsoft Visual Studio.

### 2.2.3 Примеры вопросов подготовки к промежуточной аттестации для оценки сформированности компетенции ПК-5 на уровне умений

1. В C++ создайте список объектов, находящихся в отношении наследования.
2. Дана квадратная матрица целых чисел. Найти среднее арифметическое элементов каждой строки матрицы.
3. Перечислите отличия перегрузки метода от его переопределения. Приведите примеры.
4. Создайте оконное приложение на C++.
5. Создайте в классе конструктор по умолчанию.
6. Проверьте корректность значения вещественного числа, вводимого пользователем с клавиатуры.
7. Перегрузить операцию  $>$  для экземпляров класса в программе на C++.
8. Создайте строку в программе на C++, используя STL/

### 2.2.4 Оценка сформированности компетенции ПК-5 на уровне владений определяется по результатам выполнения лабораторного практикума

#### 2.2.5 Оценивание на промежуточной аттестации

Индикаторы достижения компетенции	Пороговый уровень (удовлетворительно или 3 балла)	Базовый (хорошо или 4 балла)	Повышенный (отлично или 5 баллов)
ПК-5.1	Знание понятийного аппарата изучаемой дисциплины на уровне не менее 50% правильных ответов, полученных при промежуточном тестировании	Знание понятийного аппарата изучаемой дисциплины на уровне не менее 70% правильных ответов, полученных при промежуточном тестировании	Знание понятийного аппарата изучаемой дисциплины на уровне не менее 85% правильных ответов, полученных при промежуточном тестировании

**Оценивание результатов промежуточной аттестации с учётом критериев по компетенции**  
**Критерии оценивания промежуточной аттестации по дисциплине «Программирование на C/C++»**

Наименование критерия
1. Полнота ответов на заданные вопросы.
2. Обоснованность ответов на заданные вопросы.
3. Правильность ответов на вопросы к промежуточной аттестации с учётом требований сформированности компетенции ПК-5 на уровне знаний.
4. Правильность ответов на вопросы к промежуточной аттестации с учётом требований сформированности компетенции ПК-5 на уровне умений применения знаний.
5. Выполнение программы лабораторного практикума.

Принципы формирования оценок на основе критериев

«отлично» – все критерии соблюдены полностью,

«хорошо» – все критерии соблюдены, но недостаточна полнота и обоснованность критериев 1 и 2,

«удовлетворительно» – не соблюден хотя бы один из критериев 1 или 2,

«неудовлетворительно» – не соблюден хотя бы один из критериев 3, 4 или 5.

### Учебно-методическое и информационное обеспечение учебной дисциплины

#### 1 Рекомендуемая литература

##### 1.1 Основная литература

1.1.1. Лебеденко, Л. Ф. Программирование на C++ : учебное пособие для СПО / Л. Ф. Лебеденко, О. И. Моренкова. — Саратов : Профобразование, 2024. — 199 с. — ISBN 978-5-4488-1696-3. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/133497.html>

2. Косарева Е.А., Учебное пособие по дисциплине "Основы программирования на языке C#". / Е.А. Косарева. – Воронеж: АНОО ВО "Международный институт компьютерных технологий". – 2016. – 184 с.

##### 1.2 Дополнительная литература

1.2.1. Белева, Л. Ф. Программирование на языке C++ : учебное пособие / Л. Ф. Белева. — Москва : Ай Пи Ар Медиа, 2024. — 81 с. — ISBN 978-5-4497-2560-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/134887.html>

1.2.2. Демяненко, Я. М. Программирование на языке C++ : учебное пособие / Я. М. Демяненко, М. И. Чердынцева. — Москва : Ай Пи Ар Медиа, 2022. — 398 с. — ISBN 978-5-4497-2008-5. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/128007.html>

1.2.3. Иванов В.Б. Прикладное программирование на C/C++: с нуля до мультимедийных и сетевых приложений / Иванов В.Б.. — Москва : СОЛОН-Пресс, 2018. — 240 с. — ISBN 978-5-91359-308-5. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/90397.html>

#### 2 Рекомендуемое программное обеспечение

Microsoft Visual Studio 2013 и версии выше.

#### 3 Рекомендуемые базы данных, информационно-справочные и поисковые системы, периодика

1. Научная электронная библиотека. Режим доступа: [www.eLIBRARY.ru](http://www.eLIBRARY.ru)

2. Учебно-методическое обеспечение дисциплины «Программирование на C/C++». Электронные ресурсы МИКТ. Режим доступа: <http://sdo.iict.ru/course/view.php?id=1948>

## **Материально-техническое обеспечение учебной дисциплины**

1 В процессе преподавания дисциплины при проведении лекционных занятий используются презентации, выполненные в формате Microsoft Power Point, а также интерактивное обучение методам эффективной разработки программ и программных модулей, что вызывает необходимость применения мультимедийного проектора в специализированных аудиториях 501, 014, 015, 508.

2 Лабораторный работы выполняются с использованием программного обеспечения персональных компьютеров и предусматривают их проведение в специализированных компьютерных аудиториях.