Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Костровец Лариса Борисовна
Должность: директор
Дата подписания: 02.10.2026 00:12:42
Уникальный программный ключ:
6882606104c36dbde41c4ab93a65382136a292d6

Приложение 4
к образовательной программе

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.О.02.07 Разработка программных приложений на C#/C++
(индекс, наименование дисциплины в соответствии с учебным планом)

09.03.03 Прикладная информатика
(код, наименование направления подготовки/специальности)

Прикладная информатика в управлении корпоративными информационными системами
(наименование образовательной программы)

Очная форма обучения
(форма обучения)

Год набора – 2026
Донецк

Автор(ы)-составитель(и) РПД:

Лебезова Элла Михайловна, старший преподаватель кафедры информационных технологий

Заведующий кафедрой:

Брадул Наталья Валерьевна, канд. физ.-мат. наук, заведующий кафедрой информационных технологий

Рабочая программа дисциплины Б1.О.02.07 Разработка программных приложений на C#/C++ одобрена на заседании кафедры информационных технологий администрирования факультета государственной службы и управления Донецкого филиала РАНХиГС.

Протокол № 7 от «05 » марта 2026 г.

СОДЕРЖАНИЕ

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

2. Объем и место дисциплины в структуре образовательной программы

3. Содержание и структура дисциплины

4. Типы оценочных материалов, показатели и критерии их оценивания

5. Формы аттестации, типовые оценочные материалы для текущего контроля успеваемости обучающихся, критерии и шкалы оценивания по контрольным точкам

6. Формы промежуточной аттестации, критерии и шкала оценивания, типовые оценочные материалы по дисциплине

7. Методические материалы по освоению дисциплины

8. Учебная литература и ресурсы информационно-телекоммуникационной сети «Интернет»

9. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Дисциплина Б1.О.02.07 Разработка программных приложений на C#/C++ обеспечивает формирование у обучающихся следующих общепрофессиональных компетенций*:

ОТФ /ТФ и реквизиты ПС *(при наличии)* **	**Код компетенции ****	**Наименование Компетенции ****	**Код индикатора достижения компетенций ****	**Наименование индикатора достижения компетенций ****	**Образовательный результат ****
-	ОПК-7.	Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.1.	Разрабатывает прикладные программные приложения на C#/C++, в том числе для корпоративного использования	**Знает** основы разработки программных приложений на языках C#/C++. **Умеет** разрабатывать и использовать базы данных в программных приложениях.

** Дисциплина может формировать компетенцию полностью или частично.*
***Должно соответствовать Приложению 1 к образовательной программе*

2. Объем и место дисциплины в структуре образовательной программы

Общий объем дисциплины:

3,00 з.е., 108 ак.час

Контактная работа обучающихся с преподавателем по видам учебных занятий: 63 ак. час на контактную работу с преподавателем, из них 18 ак. час на лекции и 36 ак. час на практические занятия. 45 ак. час на самостоятельную работу обучающихся.

Б1.О.02.07. Разработка программных приложений на C#/C++ реализуется в 4-м семестре 2-го курса после изучения дисциплин:

- Информатика и программирование
- Алгоритмизация игровых задач
- Программирование на Python
- Базы данных
- Объектно-ориентированное программирование.

.

3. Содержание и структура дисциплины

3.1. Структура дисциплины

Очнаяформа обучения

№ п/п	Наименование тем и (или) разделов		Объем дисциплины, ак.час												Форма текущего контроля успеваемости, промежуточной аттестации
		ВСЕГО	Контактная работа обучающихся с преподавателем по видам учебных занятий									Самостоятельная работа			
			Период теоретического обучения						Период промежуточной аттестации (сессия)						
			Занятия лекционного типа		Занятия семинарского типа		ИК	КСР	КЭ	Каттэк	Контроль	СРкр	СРэк	СР	
			Л	ВЛ	ЛР	ПЗ									
РАЗДЕЛ 1. Разработка классических настольных приложений в среде C# и .NET															
Тема 1	Современная среда разработки приложений на C#.	11	2	0	0	4	0	0	0	0	0	0	0	5	Контрольные вопросы, практические занятия, КР1
Тема 2	Разработка графического интерфейса пользователя	14	2	0	0	4	0	0	0	0	0	0	0	6	Контрольные вопросы, практические занятия, КР1

Тема 3	Работа с графикой и анимацией	15	2	0	0	6	0	0	0	0	0	0	0	7	Контрольные вопросы, практические занятия, КР 1
Тема 4	Архитектура приложений и организация взаимодействия компонентов	15	2	0	0	6	0	0	0	0	0	0	0	7	Контрольные вопросы, практические занятия, КР 1
РАЗДЕЛ 2. Интеграция приложений и работа с данными															
Тема 5	Работа с базами данных в приложе-ниях C#	15	4	0	0	6	0	0	0	0	0	0	0	5	Контрольные вопросы, практические занятия, КР 2
Тема 6	Интеграция приложений через API	16	4	0	0	6	0	0	0	0	0	0	0	6	Контрольные вопросы, практические занятия, КР 2
Тема 7	Разработка и сопровождение программных приложений	12	2	0	0	4	0	0	0	0	0	0	0	6	Контрольные вопросы, практические занятия, КР 2
Промежуточная аттестация		9	0	0	0	0	0	0	0	9	0	0	0	0	Зачет с оценкой
Итого		**108**	**18**	**0**	**0**	**36**	**0**	**0**	**0**	**9**	**0**	**0**	**0**	**45**	

Используемые сокращения:

Л – лекции - занятия, предусматривающие преимущественную передачу учебной информации обучающимся педагогическими работниками организации и (или) лицами, привлекаемыми организацией к реализации образовательных программ на иных условиях,).

ВЛ – видео лекции.

ЛР – лабораторные работы.

ПЗ – практические занятия (за исключением лабораторных работ).

ИК – индивидуальные консультации.

КСР – контроль самостоятельной работы
КЭ – консультации перед экзаменом
Каттэк – контактная работа на аттестацию в период экзаменационных сессий
Контроль - контактная работа на аттестацию в период экзаменационных сессий для заочной формы обучения
СРкр – самостоятельная работа на подготовку курсовой работы/ курсового проекта.
СРэк – самостоятельная работа на подготовку к экзамену.
СР – самостоятельная работа в семестре на подготовку к учебным занятиям.

3.2. Содержание дисциплины

РАЗДЕЛ 1. РАЗРАБОТКА КЛАССИЧЕСКИХ НАСТОЛЬНЫХ ПРИЛОЖЕНИЙ В СРЕДЕ C# И .NET

Тема 1. Современная среда разработки приложений на C#. ОПК-7.1.

Назначение и возможности среды разработки Microsoft Visual Studio. Платформа .NET и язык C#. Структура проекта. Типы приложений. Основы работы с системой сборки и отладки программ. Использование Git и средств контроля версий.

Тема 2. Разработка графического интерфейса пользователя. ОПК-7.1.

Технологии Windows Forms и WPF. Элементы пользовательского интерфейса. Обработка событий. Компоновка элементов управления. Работа с меню, формами, диалоговыми окнами и ресурсами приложения.

Тема 3. Работа с графикой и анимацией. ОПК-7.1.

Основы компьютерной графики в настольных приложениях. Отрисовка графических примитивов. Использование Canvas и Graphics. Создание простой анимации и визуальных эффектов. Работа с мультимедийными компонентами.

Тема 4. Архитектура приложений и организация взаимодействия компонентов. ОПК-7.1.

Структура проектов Windows Forms и WPF. Принципы построения многослойных приложений. Модель данных. Шаблон MVVM. Основы обработки исключений и логирования. Организация взаимодействия между компонентами приложения.

РАЗДЕЛ 2. ИНТЕГРАЦИЯ ПРИЛОЖЕНИЙ И РАБОТА С ДАННЫМИ

Тема 5. Работа с базами данных в приложениях C#. ОПК-7.1.

Подключение к базам данных. Выполнение SQL-запросов. Технологии ADO.NET и Entity Framework. Реализация операций добавления, изменения, удаления и поиска данных. Связывание пользовательского интерфейса с источниками данных.

Тема 6. Интеграция приложений через API. ОПК-7.1.

Понятие API и веб-сервисов. Работа с HTTP-запросами. Форматы JSON и XML. Использование REST API в приложениях C#. Получение, обработка и отображение данных из внешних сервисов.

Тема 7. Разработка и сопровождение программных приложений. ОПК-7.1.

Тестирование приложений. Основы развертывания и публикации программных продуктов. Создание установочных пакетов. Основы

информационной безопасности приложений. Документирование и сопровождение программного обеспечения.

4. Типы оценочных материалов, показатели и критерии оценивания

4.1. Оценочные материалы по дисциплине Б1.О.02.07. Разработка программных приложений на C#/C++ входят в состав оценочных материалов по образовательной программе. Совокупность оценочных материалов по всем дисциплинам (модулям) образовательной программы составляет фонд оценочных средств (далее – ФОС). ФОС используется при проведении текущего контроля успеваемости и промежуточной аттестации обучающихся с целью оценивания достижения обучающимися планируемых результатов обучения.

4.2. ФОС разработан как комплекс проверочных заданий различного типа и уровня сложности, включает критерии и шкалы оценивания, а также «ключи» правильных ответов. ФОС формируется как отдельный документ и хранится в электронном виде, доступ к ФОС предоставлен ограниченному кругу лиц.

4.3. Для самостоятельной работы обучающихся при подготовке к текущему контролю успеваемости и промежуточной аттестации в рабочих программах дисциплин размещены типовые проверочные задания, которые можно условно разделить на задания закрытого, комбинированного и открытого типов.

Задания закрытого типа – это тестовые задания, в которых каждый вопрос сопровождается готовыми вариантами ответов, из которых необходимо выбрать один или несколько правильных.

Задания комбинированного типа – это тестовые задания, в которых каждый вопрос сопровождается готовыми вариантами ответов, из которых необходимо выбрать один или несколько правильных и обосновать свой выбор.

Задания открытого типа – это задания, в которых на каждый вопрос должен быть предложен развернутый обоснованный ответ.

В зависимости от типа задания рекомендованы определенная последовательность выполнения и система оценивания выполнения заданий.

4.4. Типы заданий, сценарии выполнения, критерии оценивания

ТИП ЗАДАНИЯ	ИНСТРУКЦИЯ	СЦЕНАРИИ ВЫПОЛНЕНИЯ	КРИТЕРИИ ОЦЕНИВАНИЯ
Задание закрытого типа с выбором одного правильного ответа из нескольких вариантов предложенных	Прочитайте текст, выберите правильный ответ	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается только один из предложенных вариантов. 2. Внимательно прочитать предложенные вариант-ты ответа. 3. Выбрать один верный ответ. 4. Записать только номер (или букву) выбранного варианта ответа (например, 3 или В).	Ответ считается верным, если правильно указана цифра или буква
Задание закрытого типа на установление соответствия	Прочитайте текст и установите соответствие	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидаются пары элементов. 2. Внимательно прочитать оба списка: список 1 – вопросы, утверждения, факты, понятия и т.д.; список 2 – утверждения, свойства объектов и т.д. 3. Сопоставить элементы списка 1 с элементами списка 2, сформировать пары элементов. 4. Записать попарно буквы и цифры (в зависимости от задания) вариантов ответа (например, А1 или Б4).	Ответ считается верным, если правильно указаны цифры или буквы

Задание закрытого типа с выбором нескольких правильных ответов из нескольких вариантов предложенных	Прочитайте текст, выберите правильные ответы	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается несколько правильных ответов из предложенных вариантов. 2. Внимательно прочитать предложенные вариант-ты ответа. 3. Выбрать несколько правильных ответов. 4. Записать только номера (или буквы) выбранного варианта ответа (например, 1 4 или А Г).	Ответ считается верным, если правильно установлены все соответствия (позиции из одного столбца верно сопоставлены с позициями другого)
Задание закрытого типа на установление последовательнос ти	Прочитайте текст и установите последовательность	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается последовательность элементов. 2. Внимательно прочитать предложенные варианты ответа. 3. Построить верную последовательность из предложенных элементов. 4. Записать буквы/цифры (в зависимости от задания) вариантов ответа в нужной последовательности (например, БВА или 135).	Ответ считается верным, если правильно указана вся последовательность цифр

Задание комбинированного типа с выбором одного правильного ответа из предложенных и обоснованием выбора	Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается только один из предложенных вариантов. 2. Внимательно прочитать предложенные варианты ответа. 3. Выбрать один верный ответ. 4. Записать только номер (или букву) выбранного варианта ответа. 5. Записать аргументы, обосновывающие выбор ответа (например, 4 текст обоснования).	Ответ считается верным, если правильно указана цифра или буква и приведены корректные аргументы, используемые при выборе ответа
Задание открытого типа с развернутым ответом	Прочитайте текст и запишите развернутый обоснованный ответ	1. Внимательно прочитать текст задания и понять суть вопроса. 2. Продумать логику и полноту ответа. 3. Записать ответ, используя четкие компактные формулировки. 4. В случае расчетной задачи, записать решение и ответ	Ответ считается верным: 1. Отсутствие фактических ошибок. 2. Раскрытие объема используемых понятий (полнота ответа). 3. Обоснованность ответа (наличие аргументов). 4. Логическая последовательность излагаемого материала.

4.5. Общая шкала оценивания результатов текущего контроля успеваемости и промежуточной аттестации обучающихся с применением БРС

Оценка по шкале ECTS	**Сумма баллов за все виды учебной деятельности**	**Оценка по государственной шкале**	**Определение**
A	90 – 100	«Отлично»	отличное выполнение с незначительным количеством неточностей
B	80 – 89	«Хорошо»	в целом правильно выполненная работа с незначительным количеством ошибок (до 10%)
C	75 – 79		в целом правильно выполненная работа с незначительным количеством ошибок (до 15%)
D	70 – 74	«Удовлетворительно»	неплохо, но со значительным количеством недостатков
E	60 – 69		выполнение удовлетворяет минимальные критерии
FX	35 – 59	«Не удовлетворительно»	с возможностью повторной сдачи
F	0 – 34		с обязательным повторным изучением дисциплины (выставляется комиссией)

Соотношение баллов за текущий контроль успеваемости и промежуточную аттестацию, а также повторную промежуточную аттестацию:

Максимальная сумма баллов за текущий контроль успеваемости	Максимальная сумма баллов за промежуточную аттестацию	Максимальная итоговая балльная оценка	Максимальная сумма баллов за повторную промежуточную аттестацию
100 баллов	100 баллов	100 баллов	100 баллов

5. Формы аттестации, типовые оценочные материалы для текущего контроля успеваемости обучающихся, критерии и шкалы оценивания по контрольным точкам

5.1. В ходе реализации дисциплины Б1.О.02.07 Разработка программных приложений на C#/C++ используются следующие формы

текущего контроля успеваемости обучающихся (в том числе, задания к контрольным точкам):

Контрольные вопросы для проведения опроса, задания открытого типа на практических занятиях, контрольные задания

Таблица 5.1.

Распределение баллов по видам учебной деятельности (БРС)

Раздел/Темы	Формы текущего контроля		КЗР
	УО	ПЗ	
Р-1. / Т-1	3	7	15
Р-1. / Т-2	3	7	
Р-1. / Т-3	3	7	
Р-1. / Т-4	3	7	15
Р-2. / Т-5	3	7	
Р-2. / Т-6	3	7	
Р-2. / Т-7	3	7	
Итого: 100 б	18	7	30

УО – устный опрос;
ТЗ – тестовое задание;
КЗ – контрольные задания;
ПЗ – практическое занятие;
Д – доклад;
КЗР – контрольные работы по разделу.

Критерии оценивания опроса:

Балы	Описание критерия
3	Обучающийся полно излагает материал (отвечает на вопрос), дает правильное определение основных понятий; обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только из учебника, но и самостоятельно составленные; излагает материал последовательно и правильно с точки зрения норм литературного языка.
2	Обучающийся дает ответ, удовлетворяющий тем же требованиям, что и для оценки «отлично», но допускает 1–2 ошибки, которые сам же исправляет, и 1–2 недочета в последовательности и языковом оформлении излагаемого.
1	Обучающийся обнаруживает знание и понимание основных положений данной темы, но излагает материал неполно и допускает неточности в определении понятий или формулировке правил; не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; излагает материал непоследовательно и допускает ошибки в языковом оформлении излагаемого.
0	Обучающийся обнаруживает незнание вопроса, допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал.

0* - в журнал академической группы не выставляется

Критерии оценивания практических занятий:

Балы	Описание критерия	
3	Свыше 90% правильных ответов.	Обучающийся демонстрирует глубокое познание в освоенном материале.
2	Свыше 70% правильных ответов.	Обучающимся материал освоен полностью, без существенных ошибок.
1	Реализовано более 50% поставленных задач	Обучающимся материал освоен не полностью, имеются значительные пробелы в знаниях.
0	Реализовано менее 30% поставленных задач.	Обучающимся материал не освоен, знания обучающегося ниже базового уровня.

0* - в журнал академической группы не выставляется

Критерии оценивания контрольных заданий:

Балы	Описание критерия
12-15	Обучающимся задание выполнено без ошибок и в полном объеме.
8-11	Обучающимся в целом задание выполнено, имеются отдельные неточности или недостаточно полные ответы, не содержащие ошибок.
5-7	Обучающимся допущены отдельные ошибки при выполнении задания
0-4	У обучающегося отсутствуют ответы на большинство вопросов задачи, задание не выполнено или выполнено не верно.

0* - в журнал академической группы не выставляется

5.2. Типовые оценочные материалы для текущего контроля успеваемости обучающихся (вне контрольных работ):

РАЗДЕЛ 1. РАЗРАБОТКА КЛАССИЧЕСКИХ НАСТОЛЬНЫХ ПРИЛОЖЕНИЙ В СРЕДЕ C# И .NET

Тема 1. Современная среда разработки приложений на C#

Контрольные вопросы:

1. Для чего предназначена среда Microsoft Visual Studio?
2. Какие типы проектов можно создавать в Visual Studio?
3. Что такое платформа .NET?
4. Какие основные элементы включает структура проекта C#?
5. Для чего используется файл решения (*.sln)?
6. Что такое компиляция программы?
7. Какие средства отладки предоставляет Visual Studio?
8. Для чего используется система контроля версий Git?
9. Что такое точка останова?
10. Какие преимущества дает использование IDE при разработке приложений?

Практические занятия:

Создать консольное приложение в среде Microsoft Visual Studio.

1. Создать новый проект на языке C#.
2. Реализовать программу вывода информации о студенте: – фамилия; – имя; – номер группы.
3. Добавить обработку ввода данных с клавиатуры.
4. Реализовать проверку корректности введенных данных.
5. Выполнить отладку программы с использованием точек останова.
6. Сохранить проект в локальном репозитории Git.

Тема 2. Разработка графического интерфейса пользователя

Контрольные вопросы:

1. Что такое графический интерфейс пользователя?
2. В чем различие Windows Forms и WPF?
3. Какие основные элементы управления используются в оконных приложениях?
4. Что такое событие в интерфейсном приложении?
5. Как осуществляется обработка событий?
6. Для чего используются формы и диалоговые окна?
7. Что такое компоновка элементов интерфейса?
8. Какие контейнеры элементов используются в WPF?
9. Для чего применяется меню приложения?
10. Какие требования предъявляются к пользовательскому интерфейсу?

Практические занятия:

Разработать оконное приложение «Калькулятор стоимости товара».

1. Создать форму приложения.
2. Разместить элементы управления: – текстовые поля; – кнопки; – метки.
3. Реализовать ввод цены и количества товара.
4. Организовать обработку нажатия кнопки расчета.
5. Вывести результат вычисления на форму.
6. Реализовать очистку полей формы.

Тема 3. Работа с графикой и анимацией

Контрольные вопросы:

1. Что такое компьютерная графика?
2. Какие графические примитивы используются в C#?
3. Для чего предназначен объект Graphics?
4. Что такое Canvas в WPF?
5. Как выполняется отрисовка фигур?
6. Что такое анимация в приложениях?

7. Какие типы анимации используются в WPF?
8. Как реализуется движение объектов?
9. Для чего применяются таймеры?
10. Где используются графика и анимация в программных приложениях?

Практические занятия:

Разработать приложение для отображения и анимации графических объектов.

1. Создать окно приложения.
2. Реализовать отображение: – круга; – прямоугольника; – линии.
3. Настроить изменение координат графического объекта.
4. Реализовать простую анимацию движения фигуры.
5. Добавить кнопку запуска и остановки анимации.
6. Продемонстрировать работу приложения.

Тема 4. Архитектура приложений и организация взаимодействия компонентов

Контрольные вопросы:

1. Что понимается под архитектурой программного приложения?
2. Какие существуют типы архитектуры приложений?
3. Что такое многослойное приложение?
4. Для чего используется шаблон MVVM?
5. Что такое модель данных?
6. Как организуется взаимодействие компонентов приложения?
7. Для чего применяется обработка исключений?
8. Что такое логирование?
9. Какие преимущества дает разделение логики приложения?
10. Что такое сопровождение программного обеспечения?

Практические занятия:

Разработать приложение «Список задач» с разделением логики и интерфейса.

1. Создать модель данных Task.
2. Реализовать: – добавление задачи; – удаление задачи; – изменение статуса выполнения.
3. Организовать отображение списка задач в интерфейсе.
4. Реализовать обработку ошибок ввода данных.
5. Добавить вывод сообщений об ошибках.
6. Продемонстрировать взаимодействие компонентов приложения.

Тема 5. Работа с базами данных в приложениях C#

Контрольные вопросы:

1. Что такое база данных?
2. Для чего используется SQL?
3. Что такое ADO.NET?
4. Для чего применяется Entity Framework?
5. Какие операции относятся к CRUD?
6. Как осуществляется подключение к базе данных?
7. Что такое таблица и запись?
8. Для чего используется связывание данных?
9. Какие преимущества дает ORM?
10. Какие меры безопасности используются при работе с базами данных?

Практические занятия:

Разработать приложение для учета товаров с подключением к базе данных.

1. Создать таблицы товаров и категорий.
2. Реализовать подключение к базе данных.
3. Организовать: – добавление записи; – редактирование записи; – удаление записи; – поиск записи.
4. Реализовать отображение данных в таблице приложения.
5. Добавить проверку корректности ввода данных.
6. Продемонстрировать работу приложения.

Тема 6. Интеграция приложений через API

Контрольные вопросы:

1. Что такое API?
2. Для чего используются веб-сервисы?
3. Что такое HTTP-запрос?
4. Какие методы HTTP используются при работе с API?
5. Что такое JSON?
6. В чем отличие JSON и XML?
7. Как выполняется отправка запросов в C#?
8. Для чего используется REST API?
9. Какие ошибки могут возникать при работе с API?
10. Где применяется интеграция приложений?

Практические занятия:

Разработать приложение для получения данных из внешнего API.

1. Подключить библиотеку для выполнения HTTP-запросов.
2. Реализовать запрос к открытому API.
3. Получить данные в формате JSON.
4. Выполнить обработку и вывод данных в интерфейсе приложения.
5. Реализовать обработку ошибок подключения.
6. Продемонстрировать работу приложения.

Тема 7. Разработка и сопровождение программных приложений

Контрольные вопросы:

1. Что такое тестирование программного обеспечения?
2. Какие виды тестирования существуют?
3. Для чего используется публикация приложения?
4. Что такое установочный пакет?
5. Какие требования предъявляются к сопровождению программ?
6. Что такое документация программного продукта?
7. Какие средства защиты данных применяются в приложениях?
8. Для чего используется резервное копирование?
9. Какие ошибки относятся к критическим?
10. Какие этапы включает жизненный цикл программного обеспечения?

Практические занятия:

Подготовить программное приложение к публикации.

1. Выполнить тестирование разработанного приложения.
2. Проверить корректность работы основных функций.
3. Подготовить описание программы: – назначение; – основные возможности; – инструкция пользователя.
4. Создать установочный пакет приложения.
5. Выполнить публикацию программы.
6. Продемонстрировать работу готового программного продукта.

5.3. Один или несколько тематических блоков дисциплины завершаются контрольной работой по разделу (далее – КР). Текущий контроль успеваемости по дисциплине предусматривает не менее 2 (двух) и не более 10 (десяти) КР в течение периода освоения дисциплины.

Максимальное количество баллов за любой тип работ в рамках КР составляет 100 (сто) баллов.

Распределение весовых коэффициентов по КР в рамках текущего контроля успеваемости по дисциплине и формулы расчета:

Наименование контрольной работы	Максимальное количество баллов за работу в рамках КР, которое может набрать студент	Коэффициент веса контрольной работы	Результат контрольной работы, участвующий в формировании итоговой балльной оценки по дисциплине
КР 1	100	0,15	15

КР 2	100	0,15	15
Итого:	x	0,30	30

Формула расчета результата контрольной работы:

Результат контрольной работы = Количество баллов за работу в рамках КР Х Коэффициент веса контрольной работы.

5.4. Формы текущего контроля успеваемости обучающихся в рамках КР и типовые оценочные материалы:

КР-1

Раздел 1. Разработка классических настольных приложений в среде C# и .NET

Теоретические вопросы:

1. Для чего используется среда Microsoft Visual Studio?
2. Что такое событие в оконном приложении?
3. В чем различие Windows Forms и WPF?
4. Что такое графический интерфейс пользователя?
5. Для чего применяется шаблон MVVM?

Практическое задание:

Разработать оконное приложение «Учет успеваемости студентов».

1. Создать форму приложения.
2. Разместить элементы управления:

– текстовые поля;
– кнопки;
– таблицу вывода данных.

3. Реализовать ввод информации о студенте:

– фамилия;
– группа;
– средний балл.

4. Организовать:

– добавление записей;
– очистку полей;
– вывод информации в таблицу.

5. Реализовать обработку ошибок ввода данных.
6. Продемонстрировать работу программы.

КР-2

Раздел 2. Интеграция приложений и работа с данными

Теоретические вопросы:

1. Что такое база данных?
2. Для чего используется Entity Framework?
3. Что такое API?
4. В каком формате обычно передаются данные через API?
5. Для чего применяется тестирование программных приложений?

Практическое задание:

Разработать приложение «Справочник товаров».

1. Создать базу данных товаров.
2. Реализовать подключение приложения к базе данных.
3. Организовать:

– добавление товара;
– редактирование товара;
– удаление товара;
– поиск товара по названию.

4. Реализовать получение данных из внешнего API и отображение их в приложении.
5. Добавить обработку ошибок подключения и ввода данных.
6. Продемонстрировать работу приложения.

6. Формы промежуточной аттестации, критерии и шкала оценивания, типовые оценочные материалы по дисциплине

6.1. Промежуточная аттестация по дисциплине «Разработка программных приложений на C#/C++» проводится в форме зачёта с оценкой в третьем семестре в письменной форме. Обучающийся получает три теоретических вопроса и одно практическое задание.

6.2. Типовые оценочные материалы промежуточной аттестации.

Вопросы к зачету с оценкой:

1. Назначение и возможности среды Microsoft Visual Studio.
2. Платформа .NET и ее основные компоненты.
3. Структура проекта на языке C#.
4. Основные этапы разработки программного приложения.
5. Компиляция и отладка программ в Visual Studio.
6. Использование системы контроля версий Git.
7. Понятие графического интерфейса пользователя.
8. Основные элементы управления Windows Forms.
9. Особенности технологии WPF.
10. Обработка событий в оконных приложениях.
11. Компоновка элементов пользовательского интерфейса.
12. Работа с диалоговыми окнами и меню.
13. Основы компьютерной графики в C#.
14. Использование объекта Graphics.
15. Работа с Canvas в WPF.
16. Создание анимации в приложениях.
17. Использование таймеров при разработке приложений.
18. Архитектура программных приложений.
19. Многослойная архитектура приложений.
20. Назначение шаблона MVVM.
21. Модель данных приложения.
22. Организация взаимодействия компонентов приложения.
23. Обработка исключений в C#.
24. Логирование программных приложений.
25. Понятие базы данных.
26. Подключение приложений к базам данных.
27. Технология ADO.NET.
28. Основы Entity Framework.
29. Выполнение CRUD-операций.
30. Связывание данных с пользовательским интерфейсом.
31. Понятие API и веб-сервисов.

32. HTTP-запросы и методы HTTP.
33. Форматы JSON и XML.
34. Использование REST API в приложениях C#.
35. Получение и обработка данных из внешних сервисов.
36. Тестирование программных приложений.
37. Виды тестирования программного обеспечения.
38. Публикация и развертывание приложений.
39. Создание установочных пакетов.
40. Документирование программных продуктов.
41. Основы информационной безопасности приложений.
42. Жизненный цикл программного обеспечения.
43. Сопровождение программных приложений.
44. Основные этапы разработки клиентского приложения.
45. Практическое применение технологий разработки приложений на C#.

Пример практического задания

Разработать программное приложение «Учет личной библиотеки» с графическим интерфейсом и подключением к базе данных.

1. Создать оконное приложение в среде Microsoft Visual Studio на языке C#.
2. Разработать пользовательский интерфейс приложения с использованием Windows Forms или WPF.
3. Реализовать форму ввода информации о книге:

– название книги;
– автор;
– год издания;
– жанр.

4. Создать базу данных для хранения информации о книгах.
5. Реализовать:

– подключение к базе данных;
– добавление записей;
– редактирование записей;
– удаление записей;
– поиск книг по названию или автору.

6. Организовать отображение данных в таблице приложения.
7. Реализовать обработку ошибок ввода данных и ошибок подключения к базе данных.
8. Дополнительно реализовать получение информации о книге из открытого API.
9. Подготовить приложение к публикации и продемонстрировать его работу.

6.3. Критерии и шкала оценивания на основе БРС.

Соответствие государственной шкалы оценивания академической успеваемости и шкалы ECTS при зачёте

Оценка по шкале ECTS	**Сумма баллов за все виды учебной деятельности**	**Оценка по государственной шкале**	**Определение**
A	90 – 100	«Отлично»	отличное выполнение с незначительным количеством неточностей
B	80 – 89	«Хорошо»	в целом правильно выполненная работа с незначительным количеством ошибок (до 10%)
C	75 – 79		в целом правильно выполненная работа с незначительным количеством ошибок (до 15%)
D	70 – 74	«Удовлетворительно»	неплохо, но со значительным количеством недостатков
E	60 – 69		выполнение удовлетворяет минимальные критерии
FX	35 – 59	«Не удовлетворительно»	с возможностью повторной сдачи
F	0 – 34		с обязательным повторным изучением дисциплины (выставляется комиссией)

6.4. Описание дополнительных материалов и оборудования, необходимых для выполнения проверочных заданий

Компьютер с операционной системой Windws с устойчивым Интернет-соединением, программные продукты с открытой лицензией: MS Visual Studio.

7. Методические материалы по освоению дисциплины

Получение углубленных знаний по изучаемой дисциплине достигается за счет дополнительных часов к аудиторной работе самостоятельной работы студентов. Выделяемые часы целесообразно использовать для знакомства с дополнительной научной литературой по проблематике дисциплины, анализа научных концепций и современных подходов к осмыслению рассматриваемых проблем. К самостоятельному виду работы студентов относится работа в библиотеках, в электронных поисковых системах и т.п. по сбору материалов, необходимых для проведения практических занятий или выполнения конкретных заданий преподавателя по изучаемым темам. Студенты могут установить диалог с преподавателем, получать консультации по выполнению заданий. В качестве оценочных средств на протяжении семестра используются практические задания.

Обучение по дисциплине «Разработка программных приложений на C#/C++» предполагает изучение курса на аудиторных занятиях (лекции, практические занятия) и самостоятельную работу студентов. Практические занятия дисциплины предполагают их проведение в различных формах с целью выявления полученных знаний, умений, навыков и компетенций с проведением контрольных мероприятий. С целью обеспечения успешного обучения студент должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

– знакомит с новым учебным материалом;
– разъясняет учебные элементы, трудные для понимания;
– систематизирует учебный материал;
– ориентирует в учебном процессе.

Работа обучающегося на лекции:

Слушание и запись лекций – сложный вид вузовской аудиторной работы. Внимательное слушание и конспектирование лекций предполагает интенсивную умственную деятельность обучающегося. Краткие записи лекций, их конспектирование помогает усвоить учебный материал. Конспект является полезным тогда, когда записано самое существенное, основное и сделано это самим обучающимся.

Подготовка к практическим занятиям:

Подготовку к каждому практическому занятию каждый обучающийся должен начать с ознакомления с планом, который отражает содержание предложенной темы. Тщательное продумывание и изучение вопросов плана основывается на проработке текущего материала лекции, а затем изучения обязательной и дополнительной литературы, рекомендованную к данной теме. Если программой дисциплины предусмотрено выполнение практического задания, то его необходимо выполнить с учетом предложенной инструкции. Все новые понятия по изучаемой теме необходимо внести в глоссарий, который целесообразно вести с самого начала изучения курса. Результат такой работы должен проявиться в способности обучающегося свободно ответить на теоретические вопросы практического занятия, его выступлении и участии в коллективном обсуждении вопросов изучаемой темы, правильном выполнении практических заданий и контрольных работ.

Структура практического занятия:

В зависимости от содержания и количества отведенного времени на изучение каждой темы может практическое занятие состоять из четырех-пяти частей:

1. Устный опрос.

2. Обсуждение теоретических вопросов, определенных программой дисциплины.

3. Выполнение практических заданий с последующим разбором полученных результатов или обсуждение практического задания, выполненного дома.

4. Подведение итогов занятия.

Работа с литературными источниками:

В процессе подготовки к практическим занятиям, обучающимся необходимо обратить особое внимание на самостоятельное изучение рекомендованной учебно-методической (а также научной и популярной) литературы. Самостоятельная работа с учебниками, учебными пособиями, научной, справочной и популярной литературой, материалами периодических изданий и Интернета, статистическими данными является наиболее эффективным методом получения знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у обучающихся свое отношение к конкретной проблеме. Более глубокому раскрытию вопросов способствует знакомство с дополнительной литературой, рекомендованной преподавателем, что позволяет обучающимся проявить свою индивидуальность в рамках выступления на занятиях, выявить широкий спектр мнений по изучаемой проблеме.

8. Учебная литература и ресурсы информационно-телекоммуникационной сети Интернет

8.1. Основная литература

1. 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: 14.05.2026). — Режим доступа: для авториз. пользователей.

2. Залогова, Л. А. Основы объектно-ориентированного программирования на базе языка C# / Л. А. Залогова. — 4-е изд., стер. — Санкт-Петербург : Лань, 2023. — 192 с. — ISBN 978-5-507-48276-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/345992 (дата обращения: 13.05.2026). — Режим доступа: для авториз. пользователей.

8.2. Дополнительная литература

3. Вафин, Р. Р. Разработка программных приложений на C#/C++ на C#.NET : учебно-методическое пособие / Р. Р. Вафин. — Казань : КНИТУ-КАИ, 2020. — 96 с. — ISBN 978-5-7579-2431-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/264920 (дата обращения: 13.05.2026). — Режим доступа: для авториз. пользователей.

8.3. Нормативные правовые документы и иная правовая информация

1. Конституция Российской Федерации. – Текст : электронный // Сайт Президента Российской Федерации. – URL: http://www.kremlin.ru/acts/cnstitutin

8.4 Интернет-ресурсы

1. Информационно-правовой портал ГАРАНТ.РУ. – URL: https://www.garant.ru/

2. Научная электронная библиотека eLIBRARY.RU. – URL: https://elibrary.ru/

3. Научная электронная библиотека «КиберЛенинка». – URL: https://cyberleninka.ru

4. Электронно-библиотечная система «Лань». – URL: http://e.lanbk.cm

5. База знаний по C# – URL: https://metanit.com/sharp/tutorial/

6. Документация по C# – https://learn.microsoft.com/ru-ru/dotnet/csharp/language-reference/

9. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы

Материально-техническое обеспечение дисциплины включает в себя:

- лекционные аудитории, оборудованные видеопроекционным оборудованием для презентаций, средствами звуковоспроизведения, экраном;
- помещения для проведения практических занятий, оборудованные учебной мебелью.

Дисциплина поддержана соответствующими программными продуктами с открытой лицензией: MS Visual Studio.

Вуз обеспечивает каждого обучающегося рабочим местом в компьютерном классе в соответствии с объемом изучаемых дисциплин, обеспечивает выход в сеть Интернет.

Помещения для самостоятельной работы обучающихся включают следующую оснащенность: столы аудиторные, стулья, доски аудиторные, компьютеры с подключением к локальной сети института (для компьютерных аудиторий) и Интернет. Для изучения учебной дисциплины используются автоматизированная библиотечная информационная система и электронные библиотечные системы.