

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Порохня Андрей Александрович

Должность: И.о. директора института перспективной инженерии

Дата подписания: 19.06.2026 16:38:42

Уникальный программный ключ:

990bea3aecc48c23bd8699e48288f8d1960c600

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ**

**Федеральное государственное автономное образовательное учреждение
высшего образования**

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ

И.о. директора института пер-
спективной инженерии

_____ А.А.Порохня

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

для проведения текущего контроля успеваемости и промежуточной аттестации по
дисциплине

Технологии разработки программного обеспечения

Направление подготовки	<u>09.03.04 Программная инженерия</u>
Направленность (профиль)	<u>Инженерия сложных программных систем</u>
Форма обучения	очная
Год начала обучения	2026
Реализуется в	4 семестре

Введение

1. Назначение: Оценочные материалы (оценочные средства) предназначены для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся, и представляют собой совокупность контрольно-измерительных материалов и методов их использования для измерения уровня достижения обучающимися установленных результатов обучения.

2. ФОС является приложением к программе дисциплины «Технологии разработки программного обеспечения».

3. Разработчик Новикова Е.Н., заведующая межинститутской базовой кафедры департамента цифровых, робототехнических систем и электроники

4. Проведена экспертиза ФОС.

Члены экспертной группы:

Председатель: Новикова Елена Николаевна, зав. межинститутской базовой кафедрой

(ФИО, должность)

Члены комиссии:

Николаев Евгений Иванович, доцент департаменты цифровых и робототехнических систем и электроники.

(ФИО, должность)

Винокурский Д. Л., доцент департамента цифровых, робототехнических систем и электроники

(ФИО, должность)

Представитель организации-работодателя: А. В. Новиков, начальник отдела АСУТП ООО «ЕНДС-Ставрополь»

(ФИО, должность)

Экспертное заключение: Оценочные материалы (оценочные средства) по дисциплине «Технологии разработки программного обеспечения» составлены в соответствии с требованиями самостоятельно устанавливаемого Федерального государственного образовательного стандарта высшего образования по направлению 09.03.04 «Программная инженерия» и позволяют оценить уровень сформированности компетенции ПК-3.

5. Срок действия ФОС определяется сроком реализации образовательной программы.

1. Описание показателей и критериев оценивания на различных этапах их формирования, описание шкал оценивания

Уровни сформированности компетенци(ий), индикатора (ов)	Дескрипторы			
	Минимальный уровень не достигнут (Неудовлетворительно) 2 балла	Минимальный уровень (удовлетворительно) 3 балла	Средний уровень (хорошо) 4 балла	Высокий уровень (отлично) 5 баллов
Компетенция: ПК-3 – Владеет навыками разработки программного обеспечения: способен реализовывать алгоритмы, писать код на современных языках программирования, создавать веб- и desktop-приложения, работать с базами данных				
Результаты обучения по дисциплине (модулю): <i>Индикатор:</i> ПК-3ид-1 – Разрабатывает алгоритмы и реализует их на современных языках программирования (C++, C#, Python, Java, JavaScript/TypeScript, Go) с использованием эффективных структур данных и стандартных библиотек	Не знает базового синтаксиса C#. Не может назвать основные типы данных (int, string, bool). Не понимает принципов ООП (наследование, инкапсуляция, полиморфизм).	Пишет простейшие программы на C# с использованием переменных, условных операторов (if/else), циклов (for, while). Создает классы с полями и методами, но допускает ошибки в области видимости (public/private). Понимает наследование на уровне «дочерний класс получает методы родителя».	Уверенно использует все основные конструкции C#: перечисления (enum), коллекции (List, Dictionary), исключения (try-catch-finally). Применяет принципы ООП: наследование с переопределением (virtual/override), интерфейсы, полиморфизм через вызов методов базового класса. Использует стандартные библиотеки (LINQ, работа со строками, DateTime).	Глубоко владеет семантикой C#: значимые и ссылочные типы, boxing/unboxing, паттерны (using для IDisposable, async/await, record, switch expression). Проектирует иерархии классов с учетом SOLID. Эффективно выбирает структуры данных (HashSet для уникальности, Stack/Queue для LIFO/FIFO). Пишет переиспользуемый код с обобщениями (generics) и расширяющими методами.
Результаты обучения по дисциплине (модулю): <i>Индикатор:</i> ПК-3ид-2 – Создает клиент-серверные, веб- (frontend + backend) и desktop-приложения с графическим интерфейсом, применяя современные	Не работал с Windows Forms или WPF. Не может разместить элементы управления на форме. Не знает, что такое событийная модель.	Создает простую форму с кнопками, текстовыми полями, метками. Назначает обработчики событий (Click, TextChanged) через конструктор или код. Выводит графические примитивы (линии, прямоугольники) с помощью GDI+	Разрабатывает полноценное desktop-приложение на WPF с использованием разметки XAML. Применяет стандартные контролы (ListBox, DataGrid, ComboBox), обрабатывает	Создает профессиональное WPF-приложение с каскадными стилями, шаблонами, триггерами. Использует асинхронную обработку событий (async/await) для длительных операций (файловый ввод-вывод, сетевые вызовы) без

фреймворки (React, Spring, Django, WPF) и протоколы (REST, gRPC, WebSocket)		в обработке Paint.	сложные события (MouseMove, KeyDown). Рисует динамическую графику (например, график функции) с перерисовкой по таймеру. Использует двумерную графику GDI+ (пера, кисти, пути).	блокировки UI. Реализует сложную 2D-графику с трансформациями (масштабирование, поворот), буферизацией (DoubleBuffering) и перехватом событий мыши для интерактивного рисования (например, редактор векторных фигур).
Результаты обучения по дисциплине (модулю): <i>Индикатор:</i> ПК-3 _{ид-3} – Проектирует, мигрирует и оптимизирует запросы к реляционным и нереляционным БД с использованием SQL и ORM (Hibernate, Entity Framework, Django ORM)	Не умеет читать/писать файлы в C#. Не знаком с ADO.NET. Не может установить соединение с базой данных.	Выполняет чтение текстового файла (StreamReader) и запись (StreamWriter). Открывает соединение с SQL Server через SqlConnection, выполняет простой SELECT с помощью SqlCommand и читает результат через SqlDataReader. Выполняет INSERT с параметрами (без защиты от SQL-инъекций).	Реализует полный CRUD через ADO.NET: использует SqlParameter для безопасных запросов, DataSet и DataAdapter для работы с набором данных. Применяет транзакции (SqlTransaction) для согласованности операций. Пишет простые оптимизированные запросы (индексы, WHERE по ключу). Обрабатывает исключения (try-catch для SQLException).	Проектирует эффективный файловый ввод-вывод: асинхронные методы (ReadAsync, WriteAsync), буферизация, работа с бинарными файлами. В ADO.NET использует пул соединений, асинхронные команды (ExecuteNonQueryAsync), возвращает данные через DbDataReader с помощью yield return. Оптимизирует CRUD: массовая вставка через SqlBulkCopy, пакетные операции, stored procedures. Обеспечивает устойчивость при сбоях (политики повторных попыток, transient fault handling).
Результаты обучения по дисциплине (модулю): <i>Индикатор:</i> ПК-3 _{ид-6} – Реализует аутентификацию и авторизацию с	Не знает JWT, OAuth2, bcrypt. Не работал с паттернами WPF (привязка данных, стили).	Понимает концепцию хеширования паролей (bcrypt), может использовать готовую библиотеку. Знает общее устройство	Реализует систему аутентификации с JWT (генерация access/refresh токенов, проверка подписи)	Разрабатывает полноценный механизм безопасности: OAuth2 (авторизационный код, implicit flow) с интеграцией через

использованием JWT (access/refresh токены), OAuth2, Basic Auth, сессий, хеширование паролей (bcrypt, Argon2) и ролевую модель доступа (RBAC)		JWT (заголовок, полезная нагрузка, подпись). В WPF использует простую привязку данных (DataContext = объект, {Binding Path}) и встроенные стили для кнопок.	и ролевую модель RBAC (проверка прав доступа на основе Claim). Применяет bcrypt для хеширования паролей с солью. В WPF использует MVVM-паттерн с INotifyPropertyChanged, конвертеры (IValueConverter), динамические ресурсы (DynamicResource) и триггеры свойств.	IdentityServer или аналоги, ротация refresh-токенов, защита от CSRF. Использует Argon2 как более современный алгоритм хеширования. В WPF внедряет сложные паттерны: привязка к коллекциям (ObservableCollection), вложенные стили (BasedOn), ResourceDictionary с темами, DataTemplateSelector для динамической смены шаблонов, анимацию через Storyboard. Демонстрирует, как UI-паттерны помогают отображать состояние аутентификации (например, видимость элементов в зависимости от роли).
--	--	---	---	---

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕРКИ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
1.	2	Где в управляемой среде .NET CLR хранятся значимые типы (int, double, struct)? 1) В куче (heap) 2) В стеке (stack) 3) В сегменте кода 4) В глобальной куче	ПК-3
2. 2	1,3	Какие утверждения о сборщике мусора (GC) в .NET верны? (Выберите все верные) 1) GC освобождает память от недостижимых объектов в куче 2) GC автоматически освобождает неуправляемые ресурсы (файлы, соединения с БД) 3) GC использует поколения (0, 1, 2) для оптимизации производительности 4) Вызов GC.Collect() всегда рекомендуется для повышения производительности	ПК-2
3. 3	3	Какой принцип ООП нарушается, если класс Square наследуется от Rectangle и переопределяет свойства Width/Height так, что ширина и высота остаются равными? 1) Принцип единственной ответственности (SRP) 2) Принцип разделения интерфейсов (ISP) 3) Принцип подстановки Барбары Лисков (LSP) 4) Принцип открытости/закрытости (OCP)	ПК-2
4. 4	2	Что из перечисленного является примером композиции (composition) вместо наследования? 1) class Car : Vehicle 2) class Car { private Engine engine; } 3) class Car : IEngine 4) class Car where T : Engine	ПК-2
5. 5	4	Какой механизм в .NET позволяет передать методу ссылку на другой метод как параметр? 1) Событие (event) 2) Атрибут (attribute)	ПК-3

		3) Исключение (exception) 4) Делегат (delegate)	
6. 6	1	Для чего в асинхронном методе используется ConfigureAwait(false)? 1) Чтобы избежать возврата в исходный SynchronizationContext (например, в UI поток) и предотвратить deadlock 2) Чтобы отключить проверку await 3) Чтобы метод выполнялся синхронно 4) Чтобы гарантировать выполнение в потоке пула независимо от контекста	ПК-3
7. 7	3	Что выполняет метод Application.DoEvents() в Windows Forms? 1) Завершает приложение 2) Останавливает цикл сообщений 3) Принудительно обрабатывает ожидающие сообщения Windows (позволяет UI не зависать при длительных операциях) 4) Перезапускает главную форму	ПК-3
8. 8	2	Как правильно выполнить потокобезопасное обновление элемента управления TextBox из фонового потока в Windows Forms? 1) textBox1.Text = "new value"; 2) textBox1.Invoke((Action)(() => textBox1.Text = "new value")); 3) Application.Run(() => textBox1.Text = "new value"); 4) Control.CheckForIllegalCrossThreadCalls = false;	ПК-3
9. 9	1,4	Какие утверждения о паттерне MVVM (Model-View-ViewModel) верны? (Выберите все верные) 1) ViewModel не имеет прямой ссылки на View 2) View управляет данными через наследование от Model 3) Привязка данных (Data Binding) не используется в MVVM 4) ViewModel реализует интерфейс INotifyPropertyChanged для уведомления View об изменениях	ПК-2
10. 10	2	Какой метод в Windows Forms вызывается автоматически при необходимости перерисовки формы? 1) Draw() 2) OnPaint(PaintEventArgs e) 3) Refresh() 4) Invalidate()	ПК-3

11. 11	3	Что представляет собой свойство зависимости (DependencyProperty) в WPF? 1) Обычное свойство C# с get/set 2) Поле, хранящееся в стеке 3) Система свойств, поддерживающая привязку данных, анимацию, наследование значений и валидацию 4) Аналог статического поля	ПК-2
12. 12	4	Какой интерфейс должен реализовать класс, чтобы уведомлять WPF об изменении свойств для обновления привязки? 1) IDisposable 2) IComparable 3) IObservable 4) INotifyPropertyChanged	ПК-3
13. 13	2	Какой подход к работе с БД в ADO.NET использует SqlDataReader? 1) Отключённый (Disconnected) 2) Подключённый (Connected) – только чтение вперёд, постоянное открытое соединение 3) ORM 4) Только для хранимых процедур	ПК-3
14. 14	1,3	Какие преимущества даёт использование Entity Framework Core по сравнению с прямыми SQL-запросами через ADO.NET? (Выберите все верные) 1) Автоматическая генерация кода для CRUD операций (LINQ) 2) Гарантированно более высокая производительность 3) Упрощение смены СУБД (абстракция от диалекта SQL) 4) Полное отсутствие необходимости знать SQL	ПК-3
15. 15	3	Какой метод используется для применения миграций в Entity Framework Core при Code First подходе? 1) Update-Database 2) Add-Migration 3) dotnet ef database update 4) Scaffold-DbContext	ПК-3
16. 16	4	Что такое DbContext в Entity Framework Core? 1) Класс, представляющий строку подключения 2) Только конфигурация маппинга	ПК-2

		3) Контейнер для подключений ADO.NET 4) Основной класс, объединяющий сессию работы с БД, отслеживание изменений, Unit of Work и репозиторий	
17. 17	2	Какой жизненный цикл (время жизни) сервиса при регистрации в DI-контейнере Microsoft.Extensions.DependencyInjection означает создание одного экземпляра на всё приложение? 1) Transient 2) Singleton 3) Scoped 4) PerThread	ПК-2
18. 18	1	Какой паттерн проектирования напрямую реализует принцип инверсии зависимостей (DIP) через передачу зависимостей извне? 1) Dependency Injection (внедрение зависимостей) 2) Factory Method 3) Singleton 4) Observer	ПК-2
19. 19	3	Какая коллекция в WPF автоматически уведомляет об изменениях при добавлении/удалении элементов для обновления привязки ItemsControl? 1) List<T> 2) IEnumerable 3) ObservableCollection<T> 4) HashSet<T>	ПК-3
20. 20	2	Что из перечисленного является признаком утечки памяти в .NET-приложении? 1) Регулярный вызов GC.Collect() 2) Объект остаётся достижимым (есть ссылка), но больше не нужен (например, неподписанное событие) 3) Значимый тип хранится в стеке 4) Использование using для IDisposable	ПК-2
21. 21	4	Как правильно освободить неуправляемые ресурсы (например, хэндл файла) в .NET? 1) Переопределить метод ToString() 2) Положить объект в стек 3) Вызвать GC.Collect()	ПК-3

		4) Реализовать интерфейс IDisposable и вызвать Dispose() (или использовать using)	
22. 22	1	Для чего используется ключевое слово base в конструкторе производного класса? 1) Для вызова конструктора базового класса 2) Для вызова статического метода 3) Для обращения к закрытым полям базового класса 4) Для предотвращения наследования	ПК-3
23. 23	3	Какое утверждение верно для интерфейсов в C#? 1) Интерфейс может содержать реализацию методов по умолчанию (до C# 8.0) 2) Класс может наследовать не более одного интерфейса 3) Интерфейс не может содержать поля экземпляра 4) Интерфейсы не могут содержать события	ПК-3
24. 24	2	В чём разница между IComparable и IComparer? 1) IComparable используется для сортировки по внешнему критерию, IComparer – внутри самого класса 2) IComparable определяет естественный порядок сравнения внутри класса, IComparer – отдельный класс-компаратор для альтернативного порядка 3) Нет разницы, это синонимы 4) IComparer работает только с числами	ПК-2
25. 25	1,4	Какие модификаторы доступа в C# делают член доступным только внутри текущего класса и производных классов в той же сборке? (Выберите все верные) 1) protected internal 2) private protected 3) internal 4) protected	ПК-3
26. 26	3	Как в Windows Forms принудительно вызвать перерисовку элемента управления? 1) Draw() 2) Refresh() 3) Invalidate() – помечает область для перерисовки, а фактическая перерисовка происходит при следующем цикле сообщений 4) Update()	ПК-3
27. 27	2	Для чего в WPF используется IValueConverter? 1) Для преобразования типов в XAML	ПК-3

		2) Для преобразования значения источника привязки перед отображением в цели (например, bool в Visibility) 3) Для валидации данных 4) Для анимации свойств	
28. 28	4	Какая проблема решается использованием паттерна Repository поверх Entity Framework? 1) Ускорение запросов к БД 2) Отказ от LINQ 3) Полное сокрытие EF Core от бизнес-логики, упрощение тестирования 4) Верно всё, кроме 1	ПК-2
29. 29	2	Какой тип делегата в .NET предопределён для методов, возвращающих значение и принимающих один параметр? 1) Action<T> 2) Func<T, TResult> 3) Predicate<T> 4) EventHandler	ПК-3
30. 30	1	Что произойдёт, если в async-методе не использовать await? 1) Метод выполнится синхронно, компилятор выдаст предупреждение 2) Метод выполнится асинхронно, но без возврата управления 3) Возникнет исключение во время выполнения 4) Метод не скомпилируется	ПК-3
31. 31	3	Что такое boxing (упаковка) в .NET? 1) Преобразование ссылочного типа в значимый 2) Создание объекта в стеке 3) Преобразование значимого типа в ссылочный (упаковка в объект в куче) 4) Вызов виртуального метода	ПК-2
32. 32	1,4	Какие из перечисленных операций приводят к упаковке (boxing)? (Выберите все верные) 1) int i = 10; object o = i; 2) string s = "123"; int j = int.Parse(s); 3) int a = 5; int b = a; 4) ArrayList.Add(42); (ArrayList хранит object)	ПК-3

33. 33	2	Какое ключевое слово в C# используется для освобождения неуправляемых ресурсов в блоке, который выполнится даже при исключении? 1) try-catch 2) using 3) lock 4) unsafe	ПК-3
34. 34	4	Какой способ лучше всего подходит для решения проблемы «хрупкого базового класса» (fragile base class)? 1) Использовать больше уровней наследования 2) Сделать все методы виртуальными 3) Сделать все методы sealed 4) Отдавать предпочтение композиции перед наследованием (composition over inheritance)	ПК-2
35. 35	3	В чём суть принципа инверсии зависимостей (DIP)? 1) Класс должен зависеть от конкретных реализаций, а не от абстракций 2) Абстракции должны зависеть от деталей 3) Модули верхних уровней не должны зависеть от модулей нижних уровней; оба должны зависеть от абстракций 4) Каждый класс должен иметь только одну причину для изменения	ПК-2
36. 36	1	Какое из утверждений о делегатах в C# верно? 1) Делегаты являются ссылочными типами, производными от System.Delegate 2) Делегаты могут указывать только на статические методы 3) Делегаты не поддерживают multicast (вызов нескольких методов) 4) Делегаты не могут быть переданы как параметры методов	ПК-3
37. 37	2	Что произойдёт при попытке вызвать Control.InvokeRequired из фонового потока, если элемент управления был создан в другом потоке? 1) Вернёт false 2) Вернёт true, так как требуется маршалинг в UI поток 3) Выбросит исключение 4) Заблокирует текущий поток навсегда	ПК-3
38. 38	3	Какой класс в Windows Forms отвечает за отображение графических примитивов (линии, прямоугольники, текст) в событии Paint? 1) Pen	ПК-3

		2) Brush 3) Graphics 4) Bitmap	
39. 39	4	Что из перечисленного является отличительной особенностью WPF от Windows Forms? 1) WPF использует растровую графику с ручной перерисовкой 2) WPF не поддерживает стили и триггеры 3) WPF не имеет механизма привязки данных 4) WPF использует retained mode (декларативный рендеринг с визуальным деревом)	ПК-2
40. 40	2	Как называется способ привязки в WPF, при котором изменения в цели обновляют источник, но не наоборот? 1) OneWay 2) OneWayToSource 3) TwoWay 4) OneTime	ПК-3
41. 41	1,3	Какие методы используются для чтения данных в подключённом режиме ADO.NET (SqlDataReader)? (Выберите все верные) 1) reader.Read() – переход к следующей записи 2) reader.Fill() 3) reader.GetString(0), reader.GetInt32(1) и т.д. 4) reader.Update()	ПК-3
42. 42	2	Что такое SqlDataAdapter в ADO.NET? 1) Объект для выполнения хранимых процедур 2) Мост между DataSet и базой данных для заполнения DataSet и сохранения изменений обратно 3) Защищённое соединение с БД 4) Асинхронный вариант SqlCommand	ПК-3
43. 43	4	Какая команда используется в Entity Framework Core для создания новой миграции после изменения модели данных? 1) Update-Database 2) Database.EnsureCreated() 3) dotnet ef migrations list	ПК-3

		4) dotnet ef migrations add ИмяМиграции	
44. 44	1	Какой метод LINQ используется для фильтрации последовательности? 1) Where 2) Select 3) Join 4) OrderBy	ПК-3
45. 45	2	Что из перечисленного является преимуществом использования IEnumerable<T> перед List<T> при возврате из метода? 1) IEnumerable<T> всегда выполняется быстрее 2) IEnumerable<T> позволяет использовать отложенное выполнение (deferred execution) и не требует полной материализации коллекции в памяти 3) IEnumerable<T> поддерживает изменение элементов 4) IEnumerable<T> имеет свойство Count	ПК-2
46. 46	3	Какой тип исключения возникает при попытке доступа к элементу массива по индексу, выходящему за пределы? 1) NullReferenceException 2) ArgumentException 3) IndexOutOfRangeException 4) InvalidOperationException	ПК-3
47. 47	1,4	Какие ключевые слова используются для обработки исключений в C#? (Выберите все верные) 1) try 2) using 3) lock 4) finally	ПК-3
48. 48	2	Что произойдёт, если в блоке catch повторно выбросить исключение с помощью throw ex; вместо throw;? 1) Ничего, результат одинаков 2) Сбросится стек вызовов (stack trace), и потеряется информация о первоначальном месте ошибки 3) Исключение будет проигнорировано 4) Программа завершится без сообщения	ПК-3

49. 49	3	Как объявить метод, который может принимать любой тип данных и при этом сохранять типобезопасность (без использования object)? 1) Использовать ключевое слово dynamic 2) Использовать перегрузку для всех типов 3) Использовать Generics (например, void MyMethod<T>(T param)) 4) Использовать тип var	ПК-3
50. 50	2	Что такое ограничение (constraint) where T : new() в обобщённом методе? 1) T должен быть значимым типом 2) T должен иметь конструктор без параметров 3) T должен быть ссылочным типом 4) T должен реализовывать интерфейс IDisposable	ПК-3
51. 51	1,4	Какие из перечисленных паттернов проектирования относятся к порождающим (creational)? (Выберите все верные) 1) Singleton 2) Adapter 3) Decorator 4) Factory Method	ПК-2
52. 52	4	Для чего используется паттерн Unit of Work в сочетании с Repository? 1) Для ускорения запросов к БД 2) Для полного отказа от ORM 3) Для обеспечения единственного экземпляра репозитория 4) Для группировки нескольких операций в одну транзакцию (все изменения сохраняются вместе)	ПК-2
53. 53	2	Как в Windows Forms открыть диалог выбора файла и получить имя выбранного файла? 1) OpenFileDialog.Open() 2) Создать экземпляр OpenFileDialog, вызвать ShowDialog() и проверить DialogResult, затем прочитать свойство FileName 3) File.OpenDialog() 4) FolderBrowserDialog	ПК-3
54. 54	3	Какой элемент управления в Windows Forms используется для отображения и редактирования форматированного текста (с возможностью изменения шрифта, цвета)?	ПК-3

		1) TextBox 2) Label 3) RichTextBox 4) MaskedTextBox	
55. 55	1	Что делает метод Console.ReadKey()? 1) Ожидает нажатие любой клавиши и возвращает информацию о нажатой клавише; не требует нажатия Enter 2) Считывает строку до нажатия Enter 3) Очищает буфер клавиатуры 4) Завершает приложение	ПК-3
56. 56	2	Какая конструкция языка C# используется для проверки типа объекта во время выполнения и условного приведения? 1) as 2) is 3) typeof 4) sizeof	ПК-3
57. 57	3	Что вернёт выражение myObject as string, если myObject не является строкой? 1) Исключение InvalidCastException 2) Пустую строку "" 3) null 4) false	ПК-3
58. 58	2	Как создать массив целых чисел из 10 элементов, заполненный нулями? 1) int[] arr = new int[] {0,0,...0}; 2) int[] arr = new int[10]; 3) int[] arr = Array.Empty<int>(); 4) int arr[10];	ПК-3
59. 59	3	Как объявить двумерный массив (матрицу) размером 3x4 в C#? 1) int[3,4] matrix; 2) int matrix[3][4]; 3) int[,] matrix = new int[3,4]; 4) int[][] matrix = new int[3][4];	ПК-3
60. 60	4	Что такое ReadOnlyCollection<T> в .NET? 1) Коллекция, которая не может быть прочитана	ПК-3

		2) Коллекция, элементы которой нельзя изменить, но можно добавлять 3) Синоним для IEnumerable<T> 4) Обёртка только для чтения над существующей коллекцией, не позволяющая изменять элементы или структуру	
61. 61	1	Какой интерфейс нужно реализовать, чтобы класс можно было перебирать с помощью foreach? 1) IEnumerable (или IEnumerable<T>) 2) ICollection 3) IList 4) IDisposable	ПК-3
62. 62	2	В чём разница между IEnumerable и IEnumerator? 1) IEnumerable отвечает за перебор элементов, а IEnumerator – за получение переключателя 2) IEnumerable предоставляет метод GetEnumerator(), возвращающий IEnumerator, который и осуществляет перебор 3) Нет разницы, это одинаковые интерфейсы 4) IEnumerator используется только для словарей	ПК-2
63. 63	1,4	Какие из утверждений о структурах (struct) в C# верны? (Выберите все верные) 1) Структуры являются значимыми типами 2) Структуры могут наследоваться от других структур 3) Структуры могут иметь конструктор без параметров (до C# 10 – запрещён, в новых версиях разрешён) 4) Структуры могут реализовывать интерфейсы	ПК-3
64. 64	2	Что такое partial class в C#? 1) Класс, который нельзя наследовать 2) Класс, определение которого может быть разбито на несколько файлов (частичные классы) 3) Класс, содержащий только абстрактные методы 4) Класс, который не может создавать экземпляры	ПК-3
65. 65	3	Какой модификатор доступа делает член доступным только внутри текущей сборки (assembly)? 1) private 2) protected	ПК-3

		3) internal 4) protected internal	
66. 66	4	Какой метод у класса File позволяет прочитать все строки текстового файла в массив строк? 1) File.OpenText() 2) File.ReadAllText() 3) File.WriteAllLines() 4) File.ReadAllLines()	ПК-3
67. 67	2	Какой класс в .NET используется для записи текста в файл с возможностью указания кодировки? 1) FileStream 2) StreamWriter 3) BinaryWriter 4) StringWriter	ПК-3
68. 68	1	Что произойдёт при попытке открыть несуществующий файл с помощью File.OpenRead("test.txt")? 1) Будет выброшено исключение FileNotFoundException 2) Файл будет создан автоматически 3) Вернётся null 4) Программа зависнет	ПК-3
69. 69	3	Как в Windows Forms создать контекстное меню (по правой кнопке мыши) для элемента управления? 1) Добавить MainMenu 2) Добавить MenuStrip 3) Добавить ContextMenuStrip и присвоить свойству ContextMenuStrip элемента управления 4) Обработать событие MouseClick и вручную показать форму	ПК-3
70. 70	2	Какой элемент WPF позволяет размещать дочерние элементы в виде сетки с ячейками (строками и столбцами)? 1) StackPanel 2) Grid 3) WrapPanel 4) Canvas	ПК-3

71. 71	4	Что такое DependencyObject в WPF? 1) Базовый класс для всех моделей ViewModel 2) Базовый класс для всех стилей 3) Базовый класс для всех триггеров 4) Базовый класс для всех объектов, которые могут участвовать в системе свойств зависимостей (DependencyProperty)	ПК-2
72. 72	1	Как в WPF реализовать валидацию вводимых данных с отображением ошибки в интерфейсе? 1) Реализовать интерфейс INotifyDataErrorInfo (или IDataErrorInfo) и настроить ValidatesOnDataErrors в привязке 2) Использовать try-catch в коде формы 3) Отключить привязку 4) Использовать событие TextChanged	ПК-3
73. 73	2	Какое из утверждений о SqlCommand верно? 1) SqlCommand не поддерживает параметры 2) Параметры SqlParameter помогают защититься от SQL-инъекций и улучшить производительность 3) SqlCommand может выполнять только SELECT запросы 4) SqlCommand автоматически закрывает соединение	ПК-3
74. 74	1,3	Какие способы загрузки данных в DataSet существуют? (Выберите все верные) 1) SqlDataAdapter.Fill(DataSet) 2) SqlCommand.ExecuteNonQuery() 3) SqlDataAdapter.Fill(DataTable) 4) DataSet.LoadFromDatabase()	ПК-3
75. 75	4	Как в Entity Framework Core выполнить явную загрузку связанных данных (например, все заказы клиента) после того, как основная сущность уже загружена? 1) Использовать Include в первоначальном запросе 2) Использовать ThenInclude 3) Использовать Load в цикле 4) Использовать метод Collection(x => x.Orders).Load() или Reference(x => x.Customer).Load()	ПК-3
76. 76	2	Что такое AsNoTracking() в Entity Framework Core?	ПК-3

		1) Запрещает выполнение запроса к БД 2) Отключает отслеживание изменений для повышения производительности при чтении данных только для просмотра 3) Включает отложенную загрузку (lazy loading) 4) Отключает миграции	
77. 77	4	Какой метод у DbContext используется для сохранения всех изменений в базе данных? 1) Update() 2) Commit() 3) Flush() 4) SaveChanges()	ПК-3
78. 78	1	Какая архитектура приложения предполагает разделение на три слоя: Presentation, Business Logic, Data Access? 1) Трёхуровневая (трёхслойная) архитектура 2) Clean Architecture (Hexagonal) 3) Client-Server 4) Peer-to-Peer	ПК-2
79. 79	3	В чём преимущество использования DI-контейнера вместо ручного внедрения зависимостей через конструктор? 1) DI-контейнер ускоряет выполнение кода 2) Ручное внедрение невозможно для сложных графов объектов 3) DI-контейнер автоматически управляет временем жизни зависимостей и разрешает глубокие графы объектов (не нужно вручную создавать все зависимости) 4) Ручное внедрение не поддерживает интерфейсы	ПК-2
80. 80	2	Что такое Scoped время жизни сервиса в DI-контейнере ASP.NET Core (и в Microsoft.Extensions.DependencyInjection)? 1) Один экземпляр на всё приложение 2) Один экземпляр на один HTTP-запрос (или на один Scope) 3) Новый экземпляр при каждом запросе сервиса 4) Один экземпляр на поток	ПК-2
81. 81	1	Какой метод массива в C# сортирует элементы на месте? 1) Array.Sort(arr) 2) arr.Sort() (метод экземпляра)	ПК-3

		3) Enumerable.OrderBy(arr) 4) List.Sort(arr)	
82. 82	3	Каким образом можно задать начальное значение генератора случайных чисел (Random) для получения воспроизводимой последовательности? 1) Использовать Random.Create() 2) Это невозможно, Random всегда даёт разные последовательности 3) Передать фиксированное seed (например, new Random(42))	ПК-3
83. 83	2	Что из перечисленного является примером полиморфизма в ООП? 1) Перегрузка методов (compile-time polymorphism) 2) Использование виртуальных методов и переопределение (runtime polymorphism) 3) Инкапсуляция полей 4) Наследование от интерфейса	ПК-2
84. 84	4	В чём отличие abstract class от interface в C#? 1) Абстрактный класс не может содержать реализацию методов, а интерфейс может (начиная с C# 8.0) 2) Класс может реализовать несколько абстрактных классов, но только один интерфейс 3) Интерфейс может содержать поля, а абстрактный класс – нет 4) Абстрактный класс может содержать состояние (поля), конструкторы и реализацию методов, а интерфейс (до C# 8) – только сигнатуры	ПК-2
85. 85	3	Как объявить метод, который не возвращает значение и принимает два целых числа? 1) public int MyMethod(int a, int b) 2) public void MyMethod(int a, int b, out int result) 3) public void MyMethod(int a, int b) 4) public static MyMethod(int a, int b)	ПК-3
86. 86	2	Что такое рекурсия в программировании? 1) Вызов методом самого себя с другим набором параметров 2) Вызов методом самого себя (прямо или косвенно) 3) Вызов методом другого метода того же класса 4) Циклический вызов методов	ПК-3
87. 87	1	Для чего используется ключевое слово params в C#?	ПК-3

		1) Позволяет методу принимать переменное количество аргументов одного типа 2) Позволяет методу возвращать несколько значений 3) Указывает, что параметр является выходным 4) Указывает, что метод является обобщённым	
88. 88	4	Как правильно закрыть соединение с базой данных в ADO.NET, чтобы гарантировать его возврат в пул даже при исключении? 1) connection.Close() в конце метода 2) connection.Dispose() 3) Полагаться на финализатор 4) Использовать конструкцию using (SqlConnection conn = new SqlConnection(...)) { ... }	ПК-3
89. 89	2	Что такое Deadlock в асинхронном программировании на C# при использовании async/await? 1) Ситуация, когда два потока взаимно блокируют друг друга 2) Ситуация, когда await пытается вернуться в исходный SynchronizationContext (например, UI поток), который заблокирован вызовом Task.Result или Wait() – что приводит к взаимной блокировке 3) Бесконечный цикл 4) Утечка памяти	ПК-2
90. 90	3	Что из перечисленного является верным утверждением о методе ConfigureAwait(false)? 1) Он отключает асинхронность 2) Он гарантирует, что продолжение выполнится в том же потоке, что и предыдущий код 3) Он указывает, что после await не требуется возвращаться в исходный контекст синхронизации (например, можно выполниться в любом потоке пула), что помогает избежать deadlock и повысить производительность 4) Он обязателен для всех асинхронных методов	ПК-3

2. Описание шкалы оценивания

Оценка успеваемости студентов по дисциплине осуществляется в ходе текущего контроля и промежуточной аттестации. Для проверки уровня сформированности компетенций используется совокупность оценочных средств, позволяющих оценить знания, умения и навыки, приобретённые студентами в процессе изучения дисциплины.

Основными формами текущего контроля являются:

- выполнение и защита лабораторных работ;
- устные опросы (собеседования) по темам занятий;
- проверка индивидуальных заданий;
- тестирование по отдельным разделам дисциплины.

Промежуточная аттестация проводится в форме экзамена и включает выполнение практического задания и/или ответы на теоретические вопросы.

3. Критерии оценивания компетенций заданий открытого типа

Оценка «отлично» выставляется студенту, если в ходе выполнения лабораторных работ и итогового проекта продемонстрировано умение самостоятельно планировать этапы разработки desktop-приложения, декомпозировать задачи на подзадачи (например, проектирование иерархии классов с учётом SOLID, реализация слоёв доступа к данным через Repository + Unit of Work, настройка Dependency Injection вручную или через контейнер, создание WPF-интерфейса с использованием MVVM и асинхронных операций) и оценивать трудозатраты. Студент использовал систему контроля версий (Git) с осмысленной историей коммитов, ветвлением (feature branches, pull requests) и код-ревью. Применял инструменты управления задачами (Jira, Trello или GitHub Projects) для отслеживания прогресса, назначения приоритетов и фиксации технического долга. В отчёте представлены диаграммы классов, последовательностей или планы спринтов, обоснование выбора архитектурных решений (например, использование WPF вместо Windows Forms, выбор между подключённым и отключённым режимом ADO.NET, применение Entity Framework Core или прямых SQL-запросов). Компетенция освоена на высоком уровне.

Оценка «хорошо» выставляется студенту, если студент демонстрирует умение планировать основные этапы работы, использовать систему контроля версий (регулярные коммиты, понятные сообщения), но история коммитов может быть не полностью структурированной (например, отсутствуют отдельные ветки под задачи). Инструменты управления задачами применялись, но не в полной мере (например, только список задач без приоритетов и сроков). В отчёте присутствует общий план работ, однако детальная декомпозиция (например, разделение на UI, бизнес-логику, репозиторий) и оценка ресурсов выполнены не для всех этапов. При разработке используются базовые паттерны (например, наследование и интерфейсы), но могут отсутствовать продвинутые механизмы (async/await, Dependency Injection, привязка данных в WPF с INotifyPropertyChanged). Компетенция освоена на среднем уровне.

Оценка «удовлетворительно» выставляется студенту, если студент демонстрирует базовое умение использовать систему контроля версий (коммиты есть, но сообщения неинформативны, ветвление не применялось), планирование работ ограничивается перечнем лабораторных заданий без разбивки на задачи. Инструменты управления задачами не использовались либо использовались формально. В отчёте отсутствуют планы сроков и ресурсов, не показано распределение нагрузки между участниками (при групповой работе). Допущены ошибки в оценке сложности задач (например, завышение или занижение времени на реализацию класса с наследованием или настройку подключения к БД). Приложение работает, но содержит смешение логики в коде формы («умная форма»), отсутствует разделение на слои, файловый ввод-вывод или работа с БД выполнены с грубыми нарушениями (например, прямой SQL без параметров, игнорирование исключений). Компетенция освоена на минимальном уровне.

Оценка «неудовлетворительно» выставляется студенту, если студент не использовал систему контроля версий либо использовал её нерегулярно (один коммит на всю работу), не умеет планировать задачи, не проводил декомпозицию и не оценивал ресурсы. Инструменты управления задачами не применялись. В отчёте отсутствует какая-либо информация об управлении проектом, сроках или распределении работ. Программный код отсутствует или не компилируется, студент не может создать простейшее консольное приложение или форму Windows, не понимает базовых конструкций C#. Компетенция не сформирована.

4. Критерии оценивания компетенций заданий закрытого типа

Оценка «отлично» выставляется студенту, если он правильно ответил на 88-100% от общего количества заданий. Компетенция ПК-3 освоена на высоком уровне.

Оценка «хорошо» выставляется студенту, если он правильно ответил на 72-87% от общего количества заданий. Компетенция ПК-3 освоена на среднем уровне.

Оценка «удовлетворительно» выставляется студенту, если он правильно ответил на 53-71% от общего количества заданий. Компетенция ПК-3 освоена на минимальном уровне.

Оценка «неудовлетворительно» выставляется студенту, выставляется студенту, если он правильно ответил на менее 53% от общего количества заданий. Компетенция ПК-3 не сформирована.