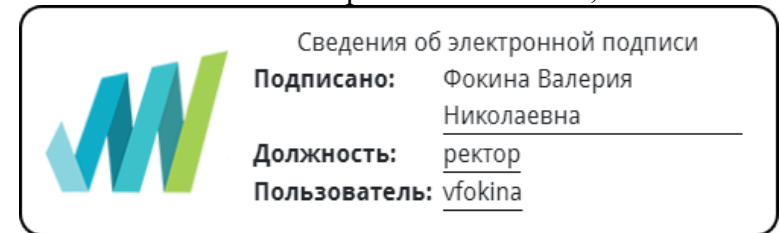


**Автономная некоммерческая организация высшего образования
«Открытый университет экономики, управления и права»
(АНО ВО ОУЭП)**

УТВЕРЖДАЮ:

Ректор АНО ВО ОУЭП, Фокина В.Н.



19 апреля 2023 г.

Решение Ученого совета АНО ВО ОУЭП,
Протокол N 9 от 19.04.2023 г.

09.03.01 «Информатика и вычислительная техника»

Направленность (профиль): Информатика и вычислительная техника

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ (МАТЕРИАЛОВ)

Приложение 1
по компетенциям

Оценочные материалы для проверки сформированности компетенции

ПК-3. Способен участвовать в тестировании информационных системы, применять современные методики тестирования разрабатываемых приложений, фиксировать выявленные ошибки кодирования в разрабатываемых модулях информационной системы, использовать критерии оценки качества и надежности функционирования информационных систем

Москва 2023

Оценочные материалы для проверки сформированности компетенции

ПК-3. Способен участвовать в тестировании информационных системы, применять современные методики тестирования разрабатываемых приложений, фиксировать выявленные ошибки кодирования в разрабатываемых модулях информационной системы, использовать критерии оценки качества и надежности функционирования информационных систем

ПК-3.1. Участвует в тестировании информационных систем, применяет современные методики тестирования разрабатываемых приложений и фиксирует выявленные ошибки кодирования в разрабатываемых модулях информационных систем

ПК-3.2. Демонстрирует навык использования критериев оценки качества и надежности функционирования информационных систем

Компетенция формируется дисциплинами:

Программирование (продвинутый уровень)	6 семестр
Разработка профессиональных приложений	7 семестр
Разработка приложений на базе объектно-ориентированного программирования	6 семестр
Разработка приложений на базе интегрированных сред	6 семестр

Вопросы и задания для проверки сформированности компетенции

Дисциплина «Разработка приложений на базе объектно-ориентированного программирования»

Разъясните основные понятия:

№	Понятие	Определение
1.	Класс	Основная концепция объектно-ориентированного программирования (ООП), представляющая шаблон или форму для создания объектов.
2.	Объект	Экземпляр класса, который содержит состояние (поля) и функциональность (методы), определенные в классе.
3.	Инкапсуляция	Принцип ООП, который связывает данные и методы, работающие с этими данными, внутри класса и скрывает их от внешнего мира.
4.	Наследование	Процесс, позволяющий создавать новые классы на основе существующих.
5.	Полиморфизм	Возможность объектов разных классов обладать одним и тем же интерфейсом и по-разному реализовывать свои методы.
6.	Интерфейс	Абстрактный тип данных, определяющий набор методов без их реализации.
7.	Конструктор	Специальный метод класса, вызываемый при создании нового объекта. Конструктор инициализирует поля объекта и выполняет другие необходимые операции.
8.	Деструктор	Специальный метод класса, вызываемый перед уничтожением объекта. Деструктор выполняет дополнительные операции, такие как освобождение ресурсов, занимаемых объектом.
9.	Множественное наследование	Возможность класса наследовать свойства и методы от нескольких родительских классов.

10.	Абстрактный класс	Класс, который имеет хотя бы один абстрактный метод. Абстрактные классы используются для создания общего интерфейса и не могут быть инстанцированы.
-----	-------------------	---

Вопросы открытого типа:

№	Вопрос	Ответ
1.	<p>Что произойдет после выполнения данного кода программы</p> <pre>Procedure TForm1.Button1Click(Sender: TObject); begin Form1.Caption:='Новая форма'; end;</pre>	При нажатии на кнопку Button1 название формы Form1 изменится на Новая форма.
2.	<p>Что произойдет после выполнения данного кода программы</p> <pre>Procedure TForm1.Button1Click(Sender: TObject); begin Form1.Width:=1200; end;</pre>	При нажатии на кнопку Button1 ширина формы станет равной 1200 пикселей.
3.	<p>Что произойдет после выполнения данного кода программы</p> <pre>Procedure TForm1.Button1Click(Sender: TObject); begin Edit1.Text:=Button1.Caption; end;</pre>	При нажатии на кнопку Button1 в текстовом окне Edit1 отобразится надпись с кнопки Button1.

4.	<p>Что произойдет после выполнения данного кода программы</p> <pre> Procedure TForm1.Button1Click(Sender: TObject); begin Edit1.Text:= Edit1.Text +Button1.Caption; end; </pre>	<p>При каждом нажатии на кнопку Button1 в текстовом окне Edit1 к предыдущему тексту будет добавляться надпись кнопки Button1.</p>
5.	<p>Что произойдет после выполнения данного кода программы</p> <pre> Procedure TForm1.Shape1MouseDown(Sender: TObject; Button: TMouseButton; Shift: TShiftState; X, Y: Integer); begin Shape1.Left:=Shape1.Left+50; Shape1.Top:=Shape1.Top-50; end; </pre>	<p>При каждом нажатии левой кнопки мыши на фигуре Shape1 она будет перемещаться по форме на 50 пикселей вправо и 50 пикселей вверх.</p>
6.	<p>Что произойдет после выполнения данного кода программы</p> <pre> Procedure TForm1.Button1Click(Sender: TObject); begin Timer1.Enabled:=True; end; </pre>	<p>При каждом нажатии на кнопку Button1 будет активирован Timer1.</p>
7.	<p>Что произойдет после выполнения данного кода программы</p> <pre> Procedure TForm1.Timer1Timer(Sender: TObject); begin Shape1.Left:=Shape1.Left+10; Shape1.Top:=Shape1.Top+10; end; </pre>	<p>При активации таймера Timer1 объект Shape1 (фигура) начнет перемещаться по диагонали сверху вниз на 10 пикселей через интервал времени, заданный в настройках таймера.</p>

8.	<p>Что произойдет после выполнения данного кода программы</p> <pre> Procedure TForm1.Button1Click(Sender: TObject); begin Shape1.Visible:=False; end; </pre>	<p>При каждом нажатии на кнопку Button1 объект Shape1 (фигура) станет невидимым.</p>
9.	<p>Что произойдет после выполнения данного кода программы</p> <pre> Procedure TForm1.Button1Click(Sender: TObject); begin n:=n+1; if n>2 then n:=1; if n=1 then Button1.Caption:='Пуск'; if n=2 then Button1.Caption:='Стоп'; end; </pre>	<p>При каждом нажатии на кнопку Button1 на ней будет меняться надпись с Пуск на Стоп и наоборот.</p>

Тестовые задания:

1	<p>В Delphi компонент представляет собой:</p> <ul style="list-style-type: none"> a) функциональный элемент интерфейса, обладающий определенными свойствами; b) набор свойств, событий и методов для решения какой-то задачи программирования; c) аллюр для отображения информации о процессах, идущих в ходе выполнения программы.
2	<p>Свойство Name в большинстве компонентов означает:</p> <ul style="list-style-type: none"> a) начало программы; b) тело программы; c) имя компонента.
3	<p>Изменить размеры формы можно в инспекторе объектов с помощью свойств:</p>

	<p>a) Height и Width; b) Длина и Ширина; c) Left и Wright.</p>
4	<p>Изменить форму фигуры компонента Shape можно с помощью свойства:</p> <p>a) color; b) shape; c) length.</p>
5	<p>Изменить цвет компонента Shape можно с помощью свойства:</p> <p>a) Properties – Events; b) Enabled - True; c) Brush – Color.</p>
6	<p>Изменить место расположения компонента на форме можно с помощью свойств:</p> <p>a) Shape и Height; b) Left и Top; c) Align - None.</p>
7	<p>Чтобы объект стал невидимый на форме нужно:</p> <p>a) свойство Visible сделать false; b) свойство True сделать Unvisible; c) удалить свойство Visible.</p>
8	<p>Компонент Timer предназначен для:</p> <p>a) установки времени создания программы; b) выполнения последовательности действий через заданный промежуток времени; c) ускорение выполнения программы.</p>

9	<p>Компонент Button – это:</p> <ul style="list-style-type: none"> a) установочный ключ загрузки системы; b) устройство прерывания; c) кнопка.
10	<p>Чтобы ввести текст в компонент Edit1 нужно выполнить:</p> <ul style="list-style-type: none"> a) Edit1.Text:=’Текст’; b) Edit1.Text==’Текст’; c) Text.Edit1.Pole=Текст.

Ключ к тестовым заданиям

1	2	3	4	5
a	c	a	b	c
6	7	8	9	10
b	a	b	c	a

Дисциплина «Разработка приложений на базе интегрированных сред»

Разъясните основные понятия:

№	Понятие	Определение
1.	Назначение технологии COM	Технология COM (Component Object Model) предоставляет механизм для разработки компонентного программного обеспечения в Windows. Она позволяет создавать независимые компоненты, которые могут быть использованы в разных приложениях.
2.	Назначение технологии ActiveX	Технология ActiveX представляет собой набор технологий и инструментов, которые позволяют создавать интерактивные и мультимедийные веб-компоненты.
3.	Назначение технологии ASP	Это технология, предназначенная для разработки динамических веб-страниц и приложений на основе сервера. ASP позволяет встраивать код программирования (например, на языке VBScript или JavaScript) непосредственно в HTML-страницы, что позволяет создавать динамический контент, взаимодействовать с базами данных и обрабатывать пользовательские запросы на сервере.
4.	Назначение элемента управления Кнопка (CommandButton) в Visual Basic	Элемент управления Кнопка представляет собой виджет, который обычно используется для активации определенного действия или события в приложении.
5.	Назначение элемента управления Надпись (Label) в Visual Basic	Элемент управления Надпись предназначен для отображения текстовой информации пользователю. Он может использоваться для вывода текстовых сообщений, меток или просто для отображения информации на форме.
6.	Назначение элемента управления Текстовое поле (TextBox) в Visual Basic	Элемент управления Текстовое поле предоставляет возможность пользователю вводить текстовую информацию. Оно может быть использовано для ввода единичного значения или для многострочного ввода текста.

7.	Назначение элемента управления Флажок (CheckBox) в Visual Basic	Элемент управления Флажок позволяет пользователю выбрать одно или несколько опций из предоставленного списка. Флажок может быть установлен или снят, в зависимости от требований приложения.
8.	Назначение элемента управления Переключатель (OptionButton) в Visual Basic	Элемент управления Переключатель предназначен для выбора одного элемента из группы взаимосвязанных опций. Пользователь может выбрать только один переключатель из группы, что облегчает выбор и управление опциями.
9.	Назначение элемента управления Список (ListBox) в Visual Basic	Элемент управления Список позволяет пользователю выбирать один или несколько элементов из предоставленного списка. Это часто используется для предоставления пользователю множества вариантов выбора.
10.	Назначение элемента управления Поле со списком (ComboBox) в Visual Basic.	Элемент управления Поле со списком комбинирует функциональность текстового поля и списка. Он позволяет пользователю выбрать значение из предоставленного списка или ввести свое собственное значение текста.

Вопросы открытого типа:

№	Вопрос	Ответ
1.	Что такое интегрированная среда разработки (IDE) и какие основные функции она выполняет?	Это программное обеспечение, объединяющее в себе различные инструменты и функции для разработки приложений. Она предоставляет редактор кода, компилятор, отладчик, инструменты автоматизации сборки и тестирования, а также другие инструменты для повышения производительности и удобства разработки.

2.	Какие преимущества предоставляет использование интегрированной среды разработки по сравнению с использованием отдельных инструментов разработки?	Интегрированная среда разработки предоставляет общую среду работы для разработчиков, объединяя различные инструменты в одном месте.
3.	Какие языки программирования поддерживаются популярными интегрированными средами разработки, такими как Visual Studio, IntelliJ IDEA и Xcode?	Популярные интегрированные среды разработки, такие как Visual Studio, IntelliJ IDEA и Xcode, поддерживают множество языков программирования. Например, Visual Studio поддерживает языки C#, Visual Basic, C++ и F#. IntelliJ IDEA поддерживает Java, Kotlin, Scala и другие языки.
4.	Какие функции отладки предоставляют интегрированные среды разработки?	Интегрированные среды разработки предоставляют инструменты отладки, такие как установка точек останова, пошаговое выполнение кода, просмотр значений переменных и стека вызовов.
5.	Какую роль играет система контроля версий (СКВ) в интегрированных средах разработки и какие преимущества она предоставляет?	Система контроля версий (СКВ) играет важную роль в интегрированных средах разработки, обеспечивая отслеживание и управление изменениями в программном коде. Она позволяет разработчикам работать над проектом одновременно, управлять версиями кода, откатывать изменения, решать конфликты слияния и обеспечивать безопасное хранение и доступ к исходному коду.
6.	Что такое автодополнение кода, и какая польза от этой функции в интегрированных средах разработки?	Автодополнение кода - это функция, позволяющая автоматически предлагать варианты завершения кода в процессе его набора. Это упрощает навигацию по коду, увеличивает производительность разработчика и помогает избежать опечаток и ошибок в синтаксисе.

7.	Какие инструменты и функции используются в интегрированных средах разработки для обеспечения качества кода и безопасности приложений?	Интегрированные среды разработки предоставляют инструменты для статического анализа кода, автоматической проверки соблюдения стандартов кодирования, выявления уязвимостей безопасности и контроля качества кода. Эти инструменты позволяют обнаружить потенциальные проблемы в коде и обеспечить безопасность приложений.
8.	Какие основные компоненты включает интегрированная среда разработки (IDE)?	Интегрированная среда разработки (IDE) включает редактор кода, компилятор, отладчик, средства автоматизации сборки и тестирования, инструменты анализа кода, систему контроля версий и другие инструменты, необходимые для разработки приложений.

Тестовые задания:

1.	Основным элементом электронной почты является
а)	почтовый ящик
б)	сообщение
в)	список рассылки
г)	линия связи

2.	База данных - это
а)	совокупность данных, организованных по определенным правилам
б)	совокупность программ для хранения и обработки больших массивов информации
в)	интерфейс, поддерживающий наполнение и манипулирование данными
г)	операционная система

3.	По типу управления локальные вычислительные сети разделяются на сети:
а)	с выделенным сервером
б)	одноранговые
в)	многоранговые

г)	многофункциональные
----	----------------------------

4.	Набор подпрограмм, используемых приложениями для совместного выполнения задач, представляет собой
а)	интерфейс программирования приложений
б)	библиотеку динамической компоновки
в)	объектно-ориентированное программирование
г)	абстрактное представление данных

5.	Для перемещения текста и графики из одного приложения в другое через буфер обмена применяются операции:
а)	копирования
б)	вырезания
в)	вставки
г)	очистки

6.	В технологии OLE объектами могут являться:
а)	фрагменты графики
б)	электронные таблицы
в)	файлы
г)	магнитные диски

7.	Табличная организация данных называется
а)	реляционной
б)	базой данных
в)	интегрированной
г)	структурной

8.	Существуют следующие типы баз данных:
а)	реляционные
б)	сетевые

в)	иерархические
г)	звездообразные

9.	_____ модель данных - это модель, в которой общая структура данных представлена в виде таблицы, где каждая строка соответствует логической записи, а заголовки столбцов являются названиями полей
Реляционная	

10.	Для обеспечения безопасности баз данных в локальной сети данные об идентификации пользователя хранятся на
а)	сервере
б)	рабочей станции пользователя
в)	всех компьютерах сети
г)	магнитной дискете у администратора

Ключ к тестовым заданиям

1	2	3	4	5
а	а	а,б,в,г	а	а,б,в,г
6	7	8	9	10
а,б,в	а	а,б,в,г	реляционная	сервере

Дисциплина «Разработка профессиональных приложений»

Разъясните основные понятия:

№	Понятие	Определение
1.	Профессиональное приложение	Программное приложение, разработанное с использованием современных инструментов и методологий, соответствующих требованиям профессиональной разработки, обладающие высоким уровнем качества, надежности, масштабируемости и производительности.
2.	Разработка программного обеспечения	Процесс создания программного продукта от начала до конца, включая анализ требований, проектирование, реализацию, тестирование и развертывание.
3.	Архитектура приложения	Структура и организация компонентов приложения, включая взаимодействие между ними.
4.	Язык программирования	Формальный набор правил и синтаксиса, используемых для написания программного кода.
5.	Интегрированная среда разработки	Программа, которая облегчает процесс разработки программного обеспечения, предоставляя средства для написания, отладки и тестирования кода.
6.	Тестирование и отладка	Тестирование является неотъемлемой частью разработки профессиональных приложений. Оно включает в себя создание тестовых сценариев, запуск тестов и анализ их результатов. Отладка используется для поиска и исправления ошибок в коде приложения.
7.	Управление версиями	Управление версиями относится к системе контроля версий, которая позволяет разработчикам отслеживать изменения в исходном коде и управлять ими.
8.	Документирование	Создание и поддержка документации, которая описывает различные аспекты проекта, такие как требования, архитектура, руководства пользователя и документация API.
9.	Облачные вычисления	Модель предоставления компьютерных ресурсов, таких как вычислительная мощность, хранение данных, сетевые ресурсы и приложения, через Интернет.

10.	Базы данных	Структурированное собрание данных, организованных, хранящихся и управляемых в цифровом формате. В базе данных информация хранится с использованием определенных моделей, схем и языков запросов.
-----	-------------	--

Вопросы открытого типа:

№	Вопрос	Ответ
1.	Перечислите правила работы со структурами, их полями и методами	<ol style="list-style-type: none"> 1. Инкапсуляция. 2. Объектно-ориентированные принципы. 3. Использование адекватных имен переменных, пролей и методов структур. 4. Ограниченный доступ. 5. Осознанное использование методов. 6. Проверка входных данных. 7. Документирование.
2.	Опишите преимущества использования модулей при программировании сложных задач	<ol style="list-style-type: none"> 1. Организация кода. 2. Повторное использование кода. 3. Улучшение читаемости. 4. Разделение ответственности. 5. Тестирование и отладка.
3.	Что включает в себя интегрированная среда разработки (IDE)?	<p>Интегрированная среда разработки включает в себя:</p> <ul style="list-style-type: none"> - текстовый редактор, - транслятор (компилятор и/или интерпретатор), - средства автоматизации сборки.
4.	Опишите процедуру создания файла структур	<ol style="list-style-type: none"> 1. Определение структуры данных: поля или атрибуты. 2. Создание нового файла в среде программирования или текстовом редакторе. 3. Объявление структуры.

		<ul style="list-style-type: none"> 4. Определение поля. 5. При необходимости объявление нескольких структур. 6. Сохранение файла.
5.	Как происходит поиск самой короткой и самой длинной строк файла?	<ul style="list-style-type: none"> 1. Открыть файл для чтения. 2. Инициализировать переменные для самой короткой и самой длинной строки. 3. Считать строку из файла. 4. Если считанная строка короче текущей самой короткой строки, обновить переменную самой короткой строки. 5. Если считанная строка длиннее текущей самой длинной строки, обновить переменную самой длинной строки. 6. Повторить шаги 3-5 для всех оставшихся строк в файле. 7. Закрыть файл. 8. Вывести самую короткую и самую длинную строку.
6.	Опишите процедуру создания рисунков в выбранной среде программирования	<ul style="list-style-type: none"> 1. Импортировать необходимые библиотеки 2. Создать холст или область рисования 3. Рисовать фигуры и объекты 4. Настроить внешний вид рисунка 5. Отобразить рисунок на экране или сохранить его в файл.
7.	Опишите процедуру преобразования и анимации изображений в выбранной среде программирования	<ul style="list-style-type: none"> 1. Импортировать необходимые библиотеки 2. Загрузить изображения 3. Преобразовать изображения. 4. Создать анимацию путем последовательного отображения изображений с разными кадрами. 5. Отобразить анимацию.
8.	Опишите процедуру преобразования и	<ul style="list-style-type: none"> 1. Импортировать необходимые библиотеки 2. Загрузить изображения

	анимации изображений в выбранной среде программирования	3. Преобразовать изображения: изменить размер, применить фильтры или изменить цветовую схему 4. Создать анимацию. 5. Отобразить анимацию.
9.	Какой алгоритм, отображающий рисунок на форме и подписывающее этот рисунок?	1. Импортировать необходимые библиотеки. 2. Создать форму или окно: размер и расположение формы на экране. 3. Загрузить рисунок. 4. Отобразить рисунок на форме. 5. Добавить подпись рисунка. 6. Отображение формы: запустить цикл отображения формы, чтобы увидеть рисунок и подпись на экране.
10.	Какой алгоритм, осуществляющий удаление из файла пустых строк?	1. Открыть файл для чтения. 2. Прочитать файл построчно. 3. Проверить является ли строка пустой 4. Сохранить непустые строки в новый файл (опционально) 5. Закрыть файлы (старый и созданный).
11.	Какой алгоритм, осуществляющий добавление строк в существующий файл?	1. Открыть файл для записи. 2. Определить строки, которые нужно добавить. 3. Записать строки в файл. 4. Закрыть файл для сохранения внесенных изменений.

Тестовые задания:

1.	По способу подключения к основной программе библиотеки можно разделить на типы:
А)	динамические
Б)	статические
В)	компиляционные
Г)	стандартные

2.	Сущность в объектно-ориентированном языке, которой можно посылать сообщения, и которая может на них реагировать, используя свои данные, - это _____
объект	

3.	Структура данных, которая может содержать в своем составе переменные, функции и процедуры – это _____
класс	

4.	Взаимодействие между объектами в объектно-ориентированном языке осуществляется с помощью _____
сообщений	

5.	Использование библиотек с динамической компоновкой (DLL) способствует _____ .
A)	разбиению кода на модули
Б)	эффективному использованию памяти
В)	сокращению дискового пространства
Г)	повышению защищенности информации

6.	Основные виды трансляторов: _____
A)	Компиляторы
Б)	Интерпретаторы
В)	Ассемблеры
Г)	Модуляторы

7.	К основным функциям редактора кода интегрированной среды разработки можно отнести: _____
А)	подсветка синтаксиса
Б)	Автодополнение
В)	Отладка
Г)	автоисправление ошибок кода

8.	Элементом интегрированной среды разработки, который позволяет изменять характеристики интерфейсных элементов, является _____
А)	окно проводника проекта
Б)	окно макета формы
В)	окно свойств
Г)	окно дизайнера форм

9.	Набор файлов, используемых для построения приложения, - это _____
Проект	

10.	Форма - контейнер, на котором располагаются элементы _____
Управления	

Ключ к тестовым заданиям

1	2	3	4	5
а,б,в	объект	класс	сообщений	а,б,в
6	7	8	9	10
а,б,в	а,б,в	а	проект	управление

Дисциплина «Программирование (продвинутый уровень)»

Разъясните основные понятия:

№	Понятие	Определение
11.	Визуальное программирование	Способ создания компьютерной программы путём манипулирования графическими объектами вместо написания её текста.
12.	Визуальная среда программирования	Современная среда программирования, основанная на парадигме визуального программирования или по-другому визуального конструирования программ. Наиболее популярными средами визуального программирования являются Visual Studio, Delphi, C++ Builder.
13.	Программное приложение	Отлаженная программа и комплекс программ ориентированных на решение конкретных задач и рассчитанная на взаимодействие с пользователем.
14.	Отладка программы	Поиск (локализация), анализ и устранение ошибок в программном приложении, которые были найдены во время тестирования.
15.	Анимация	Придание изображению подвижности, мультяшно-двигательных функций.
16.	Конструктор форм	Инструмент создания или редактирования формы путем перетаскивания на форму элементов из набора элементов и изменения их свойств.
17.	Инспектор свойств	Окно, расположенное обычно слева от окна редактора и используемое для установки свойств выделенного на форме компонента.
18.	Обработчик событий	Программа, которая выполняется в случае наступления определенного события (нажатия на кнопку, изменения содержимого текстового поля, щелчка мышью элемента и т. д.).
19.	Как добавить компонент на форму?	Добавление компонентов на форму осуществляется перетаскиванием мышью компонента из палитры компонентов на форму либо щелчком мыши на компоненте, затем в том месте формы, где требуется его разместить.

20.	Что такое таймер в Visual Studio?	Это элемент, который позволяет задать периодическое выполнение определенного кода программы через заданный интервал времени.
-----	-----------------------------------	--

Вопросы открытого типа:

№	Вопрос	Ответ
10.	<p>Что произойдет в ходе выполнения программы?</p> <pre>int x, y; Random rnd = new Random(); Random rnd1 = new Random();</pre>	Будут инициализированы два генератора случайных чисел rnd и rnd1.
11.	<p>Что произойдет в ходе выполнения программы?</p> <p>ссылка: 1</p> <pre>private void button1_Click(object sender, EventArgs e) { button1.Left = button1.Left + 150; button1.Top = button1.Top - 150; }</pre>	При нажатии на кнопку button1 она переместится на 150 пикселей вверх и 150 пикселей вправо.
12.	<p>Что произойдет в ходе выполнения программы?</p>	При нажатии на кнопку button1 она изменит свою позицию случайным образом.

	<pre>public partial class Form1 : Form { int x, y; Random rnd = new Random(); Random rnd1 = new Random(); ссылка: 1 public Form1() { InitializeComponent(); } ссылка: 1 private void button1_Click(object sender, EventArgs e) { x = rnd.Next(10, Width - 100); y = rnd1.Next(10, Height - 100); button1.Left = x; button1.Top = y; } }</pre>	
13.	Что произойдет в ходе выполнения программы?	При наведении указателя мыши на кнопку button1 она изменит свою позицию случайным образом.

```

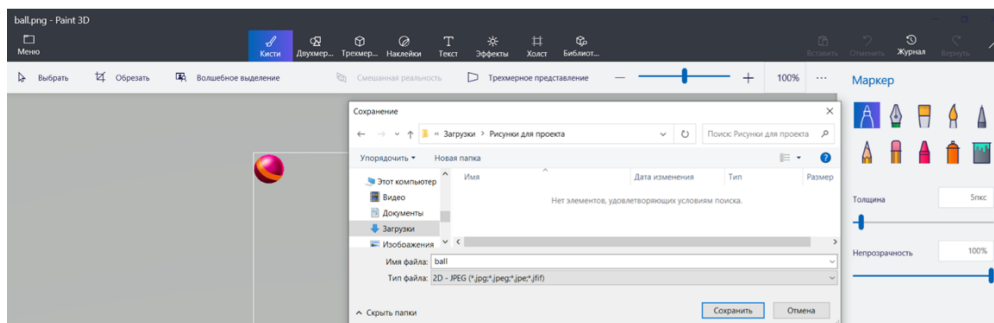
public partial class Form1 : Form
{
    int x, y;
    Random rnd = new Random();
    Random rnd1 = new Random();
    ссылка: 1
    public Form1()
    {
        InitializeComponent();
    }

    ссылка: 1
    private void button1_MouseMove(object sender, MouseEventArgs e)
    {
        x = rnd.Next(10, Width - 100);
        y = rnd1.Next(10, Height - 100);
        button1.Left = x;
        button1.Top = y;
    }
}

```

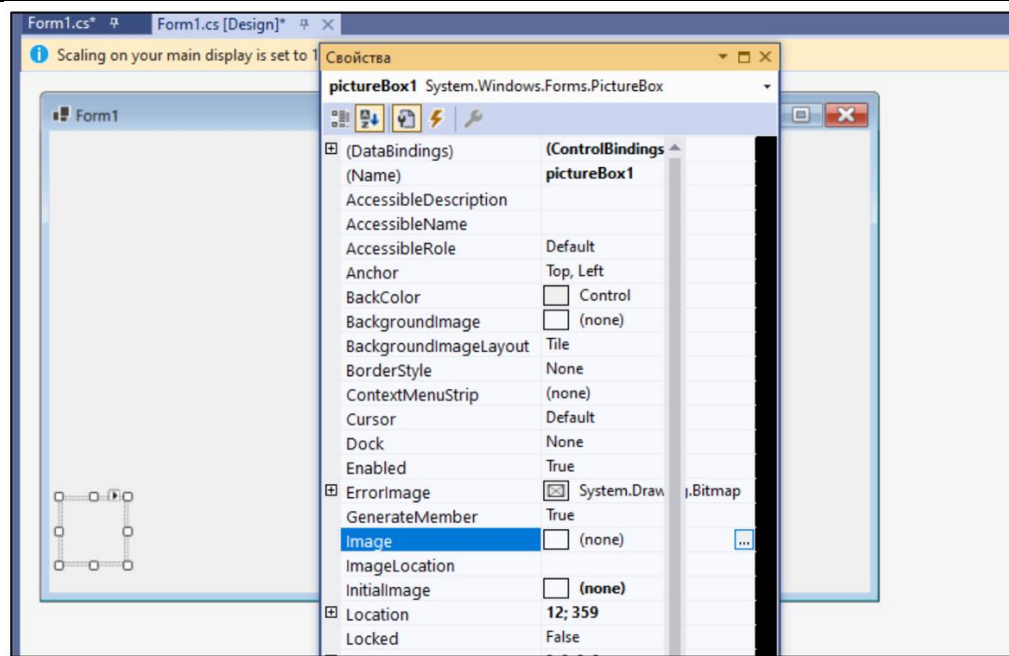
14. Какое действие осуществляет пользователь?

Редактирует изображение в Paint 3D.



15. Какое действие осуществляет пользователь?

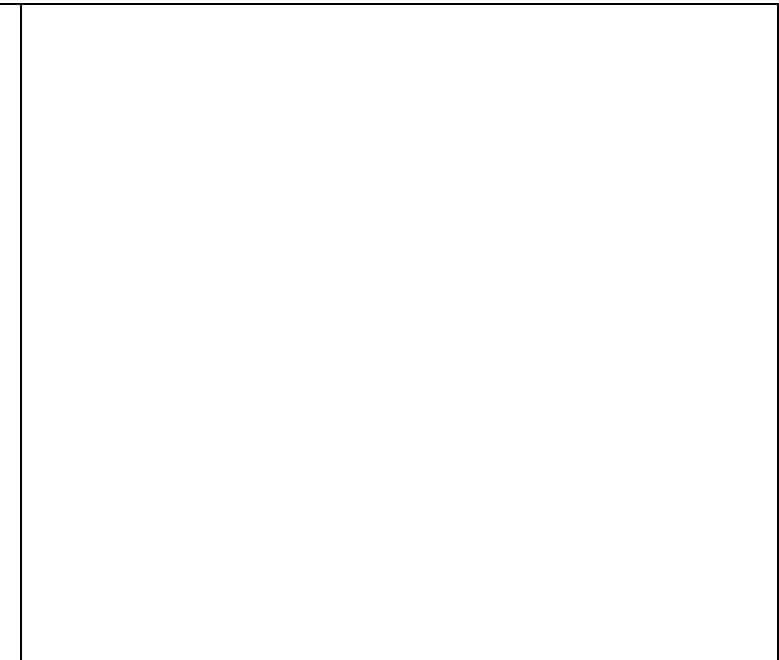
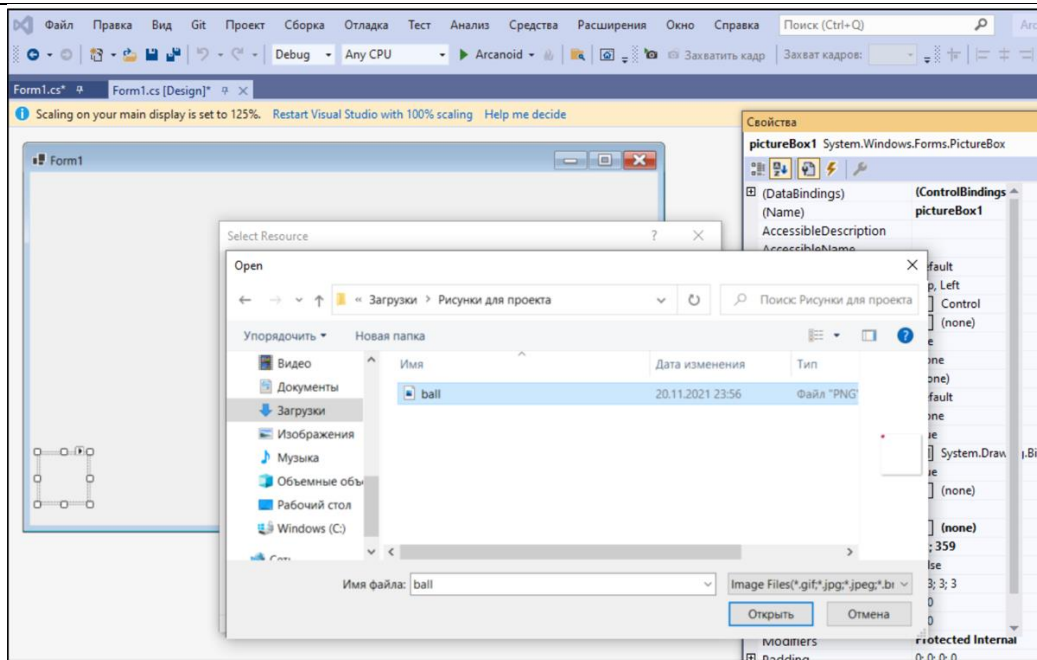
Выбор свойства Image элемента pictureBox1.



16.

Какое действие осуществляет пользователь?

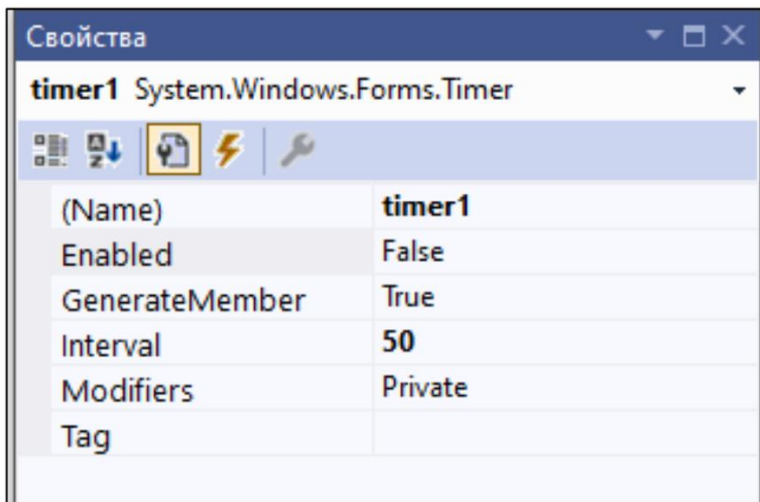
Загрузка рисунка в Select Resource.



17.

Какое действие осуществляет пользователь?

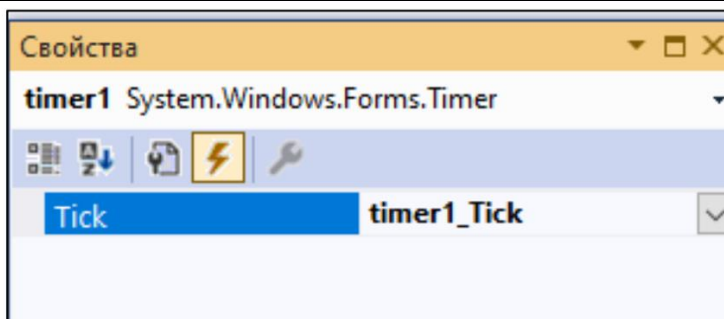
Установку свойств элемента timer1.



18.

Какое действие осуществляет пользователь?

Выбор события Tick таймера timer1.



19. Что произойдет в ходе выполнения программы?

```
public partial class Form1 : Form
{
    ссылка: 1
    public Form1()
    {
        InitializeComponent();
    }

    ссылка: 1
    private void button1_Click(object sender, EventArgs e)
    {
        timer1.Enabled = true;
    }

    ссылка: 1
    private void timer1_Tick(object sender, EventArgs e)
    {
        pictureBox1.Left = pictureBox1.Left + 10;
        pictureBox1.Top = pictureBox1.Top - 10;
    }
}
```

При нажатии на кнопку button1 контейнер с выбранным рисунком будет перемещаться влево и вверх на 10 пикселей через интервал времени заданный в timer1.

Тестовые задания:

1	<p>Какое действие выполнит данный фрагмент кода программы</p> <pre>ch = e.KeyChar; if (ch == 'd') pictureBox1.Left + 10;</pre> <p>a) Удалит рисунок из pictureBox1; b) При нажатии на клавишу d переместит картинку на 10 пикселей вправо; c) Заблокирует клавишу d.</p>
2	<p>Какое действие выполнит данный фрагмент кода программы?</p> <pre>Timer1.Enabled = true;</pre> <p>a) Включит Timer1; b) Отключит Timer1; c) Иницирует генератор случайных чисел.</p>
3	<p>Какое событие таймера Timer1 задает последовательность команд, которые он будет выполнять?</p> <p>a) Do; b) Tick; c) Work.</p>
4	<p>Какое действие выполнит данный фрагмент кода программы?</p> <pre>pictureBox1.Visible = False;</pre> <p>a) Запустит генератор случайных чисел; b) Удалит рисунок из инспектора объектов; c) Сделает рисунок невидимым.</p>
5	<p>Какое действие выполнит данный фрагмент кода программы?</p> <pre>pictureBox1.Image = Properties.Resources.juk;</pre> <p>a) Загрузит рисунок в контейнер из файла Juk; b) Изменит свойство Image контейнера рисунков; c) Удалит контейнер рисунков из инспектора ресурсов.</p>

6	В Visual Studio контейнером рисунков является элемент: PictureBox
7	Позицию элемента на форме по вертикали определяет свойство a) left b) top c) width
8	Позицию элемента на форме по горизонтали определяет свойство a) left b) top c) width
9	Событие кнопки, которое происходит в момент отпущения клавиши – это a) KeyPress b) KeyDown c) KeyUp
10	Событие кнопки, которое происходит в момент нажатия кнопки мыши, когда указатель мыши находится над компонентом – это a) MouseUp b) MouseDown c) MouseMove

Ключ к тестовым заданиям

1	2	3	4	5
b	a	Tick	c	a
6	7	8	9	10
PictureBox	b	a	c	b