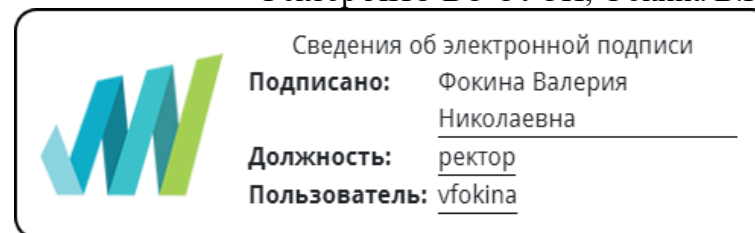


**Автономная некоммерческая организация высшего образования
«Открытый университет экономики, управления и права»
(АНО ВО ОУЭП)**

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

ОПК-8. Способен разрабатывать алгоритмы и программы, пригодные для практического применения

Оценочные материалы для проверки сформированности компетенции

ОПК-8. Способен разрабатывать алгоритмы и программы, пригодные для практического применения

ОПК-8.1. Знает: основы программирования, способы описания и методики разработки алгоритмов, современные структурные и объектно-ориентированные языки программирования

ОПК-8.2. Умеет: проводить проектирование программ с использованием современных инструментальных средств, разрабатывать алгоритмы решения профессиональных задач, разрабатывать эффективные программы, пригодные для практического применения в профессиональной деятельности

ОПК-8.3. Владеет: навыками выбора и обоснования выбора средств программирования, навыками разработки алгоритмов и программ, пригодных для практического применения

Компетенция формируется дисциплинами:

Основы алгоритмизации и программирования	1 семестр
Разработка профессиональных приложений	2 семестр
Программирование (продвинутый уровень)	3 семестр

Вопросы и задания для проверки сформированности компетенции

Дисциплина «Основы алгоритмизации и программирования»

Разъясните основные понятия:

№	Понятие	Определение
---	---------	-------------

1.	Программирование	Процесс создания компьютерных программ на одном из языков программирования.
2.	Программное приложение	Отлаженная программа или комплекс программ, ориентированных на решение конкретных задач и рассчитанных на взаимодействие с пользователем.
3.	Технология программирования	Совокупность методов и средств, используемых в процессе разработки программного обеспечения.
4.	Типы данных	Фундаментальное понятие языка программирования, которое определяет, что именно представляют собой данные, как они хранятся в памяти компьютера, как осуществляется доступ к ним, какие действия с ними можно осуществлять и в какой последовательности.
5.	Язык программирования	Формальный язык, для записи компьютерных программ. Язык программирования определяет набор лексических, синтаксических, семантических правил, определяющих внешний вид программы и действия, которые выполнит компьютер под её управлением.
6.	Оператор ввода-вывода данных	Оператор, который позволяет ввести в программу данные во время выполнения программы и осуществить вывод рассчитанных данных в понятном человеку виде.
7.	Отладка программы	Поиск (локализация), анализ и устранение ошибок в программном приложении, которые были найдены во время тестирования.
8.	Парадигма программирования	Совокупность идей и понятий, определяющих стиль разработки компьютерных программ и реализованный на языке программирования.
9.	Переменные	Поименованная либо адресуемая иным способом область памяти, адрес которой можно использовать для осуществления доступа к данным.
10.	Потоки ввода-вывода данных	Упорядоченные последовательности данных, которым соответствует

	определенный источник для потоков ввода или получатель для потоков вывода.
--	--

Вопросы открытого типа:

№	Вопрос	Ответ
1.	<p>Что выведет на экран данный фрагмент кода программы?</p> <pre>int a=5; int b; b=++a; cout << a << b << endl;</pre>	66
2.	<p>Что выведет на экран данный фрагмент кода программы?</p> <pre>int a=5; int b=6; int c=a + 5 * --b; cout << c << endl;</pre>	30
3.	<p>Что выведет на экран данный фрагмент кода программы?</p> <pre>int a=9; int b=6; int c=4; if ((a>b) && (c>a-b)) cout << a << b << c << endl; else cout << c << b << a << endl;</pre>	964
4.	<p>Что выведет на экран данный фрагмент кода программы?</p> <pre>int i, a=5, c=0; for (i=1; i<=10; i++) c=c+a; cout << c << endl;</pre>	50

5.	<p>Что выведет на экран данный фрагмент кода программы?</p> <pre>float x, x1, dx; x = 3; x1 = 4; dx = 0.2; while (x <= x1) { cout << x << " "; x = x + dx; }</pre>	3 3.2 3.4 3.6 3.8 4
6.	<p>Что выведет на экран данный фрагмент кода программы?</p> <pre>float x, x1, dx, sum; x = 3; x1 = 5; dx = 0.5; sum = 0; do {sum = sum + x; x = x + dx;} while (x <= x1); cout << sum;</pre>	20
7.	<p>Что выведет на экран данный фрагмент кода программы?</p> <pre>char ch; for (ch ='a'; ch<=d; ch++) { cout << ch << " "; }</pre>	a b c d
8.	<p>Что выведет на экран данный фрагмент кода программы?</p> <pre>string st, st1="Здравствуй, товарищи!"; for (int i =0; i <= 11; i++) st = st + st1[i];</pre>	Здравствуй

	cout << st << endl;	
9.	<p>Что выведет на экран данный фрагмент кода программы?</p> <pre>int x, y; int mass [9][9]; for (y = 1; y <= 8; y++) for (x = 1; x <= 8; x++) mass[y][x] = 0; for (y = 1; y <= 8; y++) { for (x = 1; x <= 8; x++) cout << mass[y][x] << " "; cout << endl;}</pre>	Квадрат 8 на 8, состоящий из нулей;

Тестовые задания:

1	<p>Как подключить стандартную библиотеку iostream в C++?</p> <p>a) #include <iostream>; b) #include <iostream.h>; c) #include "iostream"; d) #include 'iostream.h'.</p>
2	<p>Как правильно подключить русский язык в C++?</p> <p>a) # Setlocale<LC_All, "Russian">; b) # Setlocale<"Russian">; c) Setlocale(LC_All,"Russian"); d) Setlocale("Russian").</p>
3	<p>Где правильно инициализирована переменная целого типа в C++?</p> <p>a) int a4</p>

	<p>b) <code>int a=5;</code> c) <code>float a;</code> d) <code>char a=3.</code></p>
4	<p>Что выведет на экран данный фрагмент кода программы в C ++? <code>const int x=22;</code> <code>x++;</code> <code>cout <<x<< endl;</code></p>
5	<p>Где правильно указан комментарий в C ++? a) <code>#</code> здесь комментарий; b) <code>/*</code> здесь комментарий/<code>*/</code>; c) <code>/#</code> здесь комментарий; d) <code>//</code> здесь комментарий.</p>
6	<p>Укажите оператор выбора в C ++? <code>switch ... case ...;</code></p>
7	<p>Что выведет на экран данный фрагмент кода программы? <code>for (int y =1; y <= 10; y++) {</code> <code>for (int x =1; x <= 10; x++)</code> <code>cout << 0 << ““;</code> <code>cout << endl; }</code></p> <p>a) Сообщение об ошибке; b) Квадрат 10 на 10 состоящий из нулей; c) Квадрат 9 на 9 состоящий из нулей; d) Строку из 100 нулей.</p>

8	<p>Дан массив <code>int array[5] = { 3, 10, 7, 9, 2}</code>. Как обратиться к числу 7?</p> <p>a) <code>array[7]</code>; b) <code>array[2]</code>; c) <code>array[3]</code>; d) <code>array[2+]</code>.</p>
9	<p>Что делает фрагмент кода программы?</p> <pre>int b=0; for (int i = 0; i < 10; i++) b+=array[i];</pre> <p>a) Определяет индекс максимального элемента массива <code>array</code>; b) Подсчитывает количество элементов массива <code>array</code>; c) Вычисляет сумму индексов массива <code>array</code>; d) Посчитывает сумму первых 10 элементов массива <code>array</code>.</p>

Ключ к тестовым заданиям

1	2	3	4	5
a	b	c	Ошибка компиляции	d
6	7	8	9	10
switch ... case ...;	b	b	d	

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	<ol style="list-style-type: none"> 1. Определение структуры данных: поля или атрибуты.

	создания файла структур	<ol style="list-style-type: none"> 2. Создание нового файла в среде программирования или текстовом редакторе. 3. Объявление структуры. 4. Определение поля. 5. При необходимости объявление нескольких структур. 6. Сохранение файла.
5.	Как происходит поиск самой короткой и самой длинной строк файла?	<ol 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	<ol 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	<ol 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 анимации изображений в выбранной среде программирования	<ol style="list-style-type: none"> 1. Импортировать необходимые библиотеки 2. Загрузить изображения 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?	<ol style="list-style-type: none"> 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?	<ol style="list-style-type: none"> 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?	<ol style="list-style-type: none"> 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 _____ .
А)	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: _____
A)	подсветка синтаксиса
Б)	Автодополнение
В)	Отладка
Г)	автоисправление ошибок кода

8.	Элементом интегрированной среды разработки, который позволяет изменять характеристики интерфейсных элементов, является _____
A)	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
1.	Визуальное программирование	Способ создания компьютерной программы путём манипулирования графическими объектами вместо написания её текста.
2.	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.
3.	Программное приложение	Отлаженная программа и комплекс программ ориентированных на решение конкретных задач и рассчитанная на взаимодействие с пользователем.
4.	Отладка программы	Поиск (локализация), анализ и устранение ошибок в программном приложении, которые были найдены во время тестирования.

5.	Анимация	Придание изображению подвижности, мультяшно-двигательных функций.
6.	Конструктор форм	Инструмент создания или редактирования формы путем перетаскивания на форму элементов из набора элементов и изменения их свойств.
7.	Инспектор свойств	Окно, расположенное обычно слева от окна редактора и используемое для установки свойств выделенного на форме компонента.
8.	Обработчик событий	Программа, которая выполняется в случае наступления определенного события (нажатия на кнопку, изменения содержимого текстового поля, щелчка мышью элемента и т. д.).
9.	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.
10.	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
1.	<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.
2.	<p>Что произойдет в ходе выполнения программы?</p>	При нажатии на кнопку button1 она переместится на 150 пикселей вверх и 150 пикселей вправо.

	<pre> ссылка: 1 private void button1_Click(object sender, EventArgs e) { button1.Left = button1.Left + 150; button1.Top = button1.Top - 150; } </pre>	
3.	<p>Что произойдет в ходе выполнения программы?</p> <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>	<p>При нажатии на кнопку button1 она изменит свою позицию случайным образом.</p>
4.	<p>Что произойдет в ходе выполнения программы?</p>	<p>При наведении указателя мыши на кнопку</p>

```

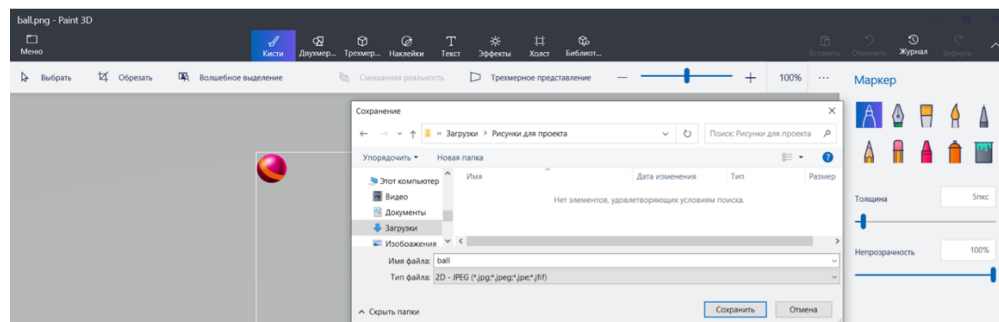
public partial class Form1 : Form
{
    int x, y;
    Random rnd = new Random();
    Random rnd1 = new Random();
    ссылка: 1
    public Form1()
    {
        InitializeComponent();
    }

    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;
    }
}

```

button1 она изменит свою позицию случайным образом.

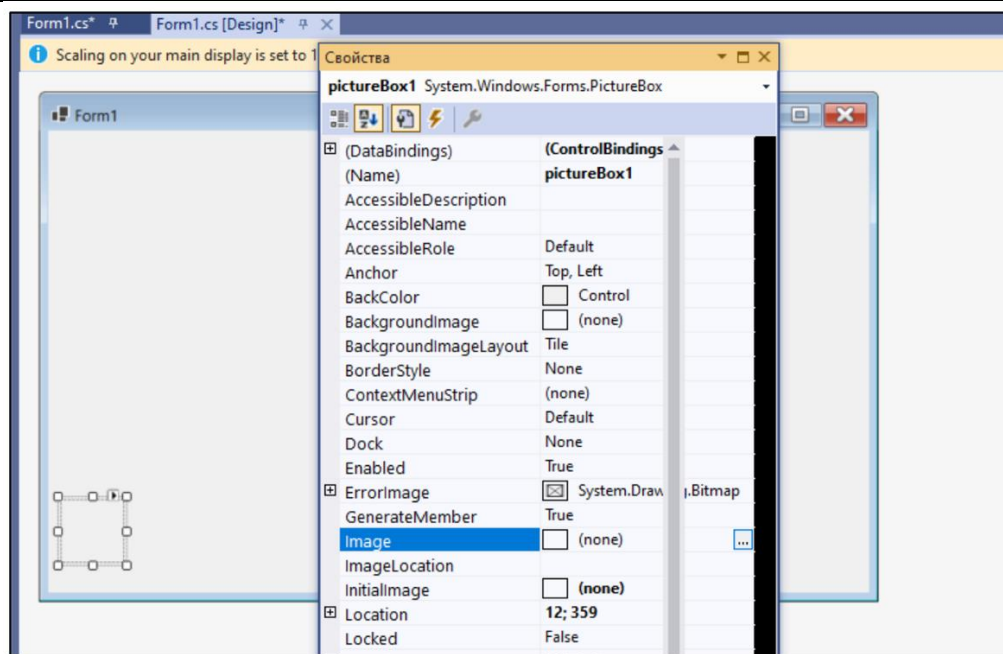
5. Какое действие осуществляет пользователь?



Редактирует изображение в Paint 3D.

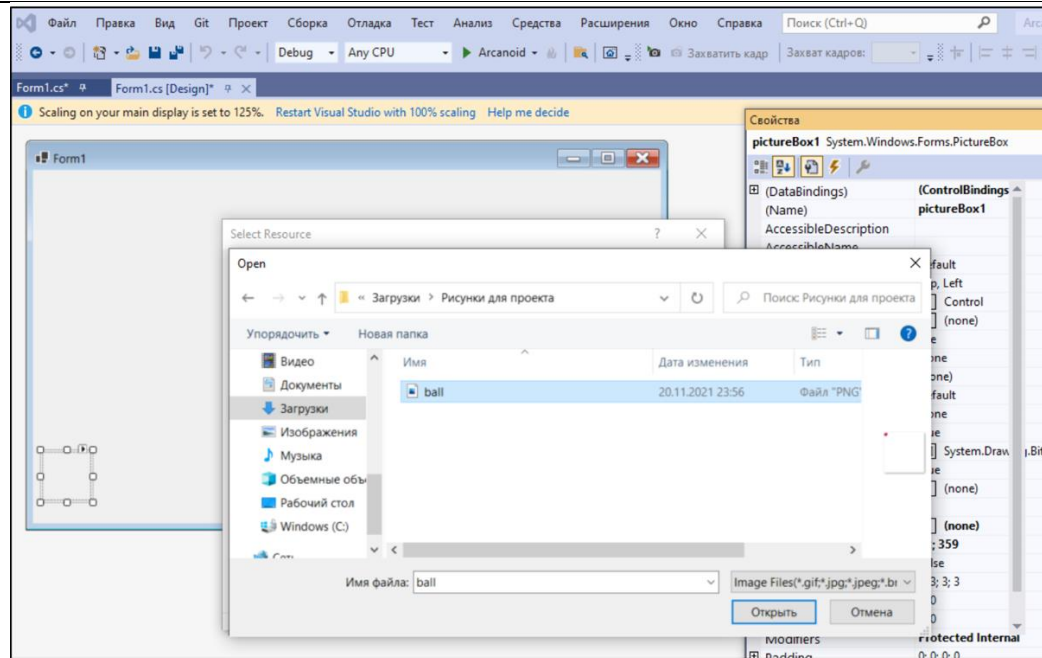
6. Какое действие осуществляет пользователь?

Выбор свойства Image элемента pictureBox1.

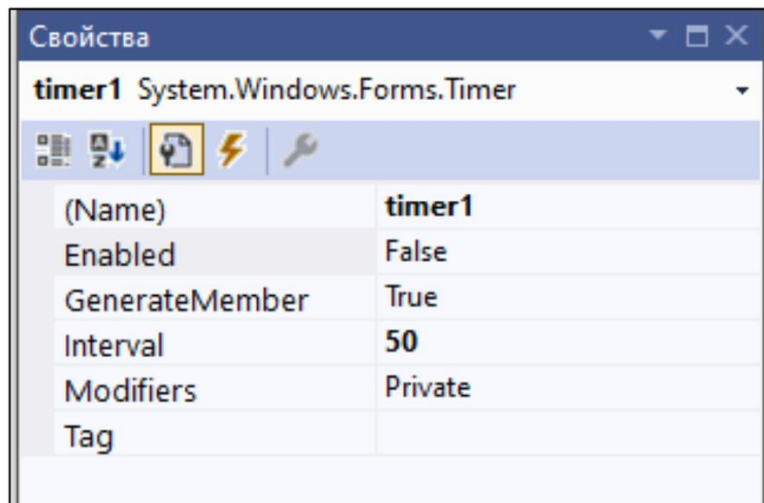


7. Какое действие осуществляет пользователь?

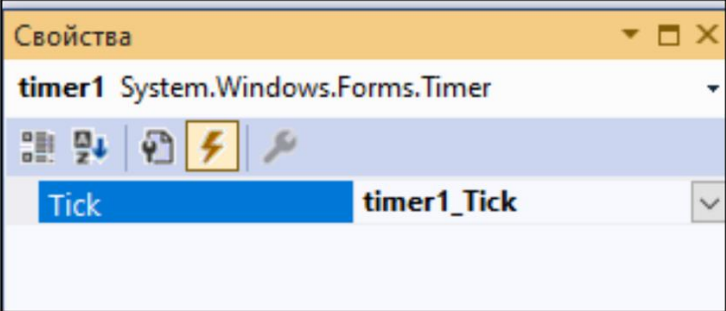
Загрузка рисунка в Select Resource.



8. Какое действие осуществляет пользователь?



Установку свойств элемента timer1.

9.	<p>Какое действие осуществляет пользователь?</p> 	<p>Выбор события Tick таймера timer1.</p>
10.	<p>Что произойдет в ходе выполнения программы?</p>	<p>При нажатии на кнопку button1 контейнер с выбранным рисунком будет перемещаться влево и вверх на 10 пикселей через интервал времени заданный в timer1.</p>

```

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;
    }
}

```

Тестовые задания:

1

Какое действие выполнит данный фрагмент кода программы

```
ch = e.KeyChar;
```

```
if (ch == 'd') pictureBox1.Left + 10;
```

а) Удалит рисунок из pictureBox1;

б) При нажатии на клавишу d переместит картинку на 10 пикселей вправо;

	<p>c) Заблокирует клавишу d.</p>
2	<p>Какое действие выполнит данный фрагмент кода программы? Timer1.Enabled = true;</p> <p>a) Включит Timer1; b) Отключит Timer1; c) Инициализирует 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? pictureBox1.Visible = False;</p> <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? pictureBox1.Image = Properties.Resources.juk;</p> <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 отпускания 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

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