

1.

В результате выполнения фрагмента программы:

```
using System;
```

```
class Program {  
    delegate void Del();  
    public static event Del MyDel;  
    static void Main() {  
        int x = 3;  
        MyDel = delegate {  
            for (int i = 0; i < 5; i++) {  
                x *= i + i;  
                Console.Write(x * i);  
            }  
        };  
        MyDel();  
    }  
}
```

на экран будет выведено:

Примечание:

*Если возникнет ошибка компиляции, введите: ****

Если ошибок и исключений нет, но на экран не выведется ничего, введите: ---

Если возникнет ошибка исполнения или исключение, введите: +++

2.

Допустимы следующие модификаторы для событий (укажите все верные ответы):

- 1) extern.
- 2) static.
- 3) abstract.
- 4) virtual.
- 5) override.

3.

В результате выполнения фрагмента программы:

```
using System;

delegate void Del(ref int d);
class Program {
    static event Del onRun;
    static void Main() {
        onRun += new Del(One);
        onRun += new Del(Two);
        int z = 5;
        while (++z < 7) {
            onRun(ref z);
            Console.Write(z);
        }
    }
    static void One(ref int x) {
        Console.Write(x += 3);
    }
    static void Two(ref int y) {
        Console.Write(y += 5);
    }
}
```

на экран будет выведено:

4.

Стандартный шаблон событий может принимать следующие аргументы
(укажите все верные ответы):

- 1) (5, null);
- 2) (null, null);
- 3) (new Delegate(), new EventArgs());
- 4) ("args");
- 5) (new Int32(), new Double());

5.

Выберите верные утверждения (укажите все верные ответы):

- 1) Событие – инкапсулированный делегат.
- 2) Стандартный шаблон событий имеет следующие входные аргументы: object sender и EventArgs args.
- 3) Событие может быть аргументом метода.
- 4) На событие можно подписать как методы экземпляра, так и статические методы.
- 5) Событие можно запустить только в классе, где это событие объявлено.

6.

Событие содержит следующие аксессоры (укажите все верные ответы):

- 1) get.
- 2) set.
- 3) add.
- 4) remove.
- 5) addRange.

7.

В результате выполнения фрагмента программы:

```
using System;
```

```
public delegate int[] MyDel(int x);
```

```
public class A {
```

```
    public event MyDel MyDelEventHandler;
```

```
    public static int[] Meth1(int x) {
```

```
        int[] args = new int[10];
```

```
        for (int i = 1; i < args.Length; i++) {
```

```
            args[i] = args[i - 1] + x * 2;
```

```
        }
```

```
        return args;
```

```
    }
```

```
    public int[] Meth2(int x) {
```

```
        int[] args = new int[5];
```

```
        for (int i = 0; i < args.Length; i++) {
```

```
            args[i] = x + i * (int)Math.Truncate(10.31 % 3);
```

```
        }
```

```
        return args;
```

```
    }
```

```
    public void Event() {
```

```
        int[] args = MyDelEventHandler?.Invoke(5);
```

```
        foreach (var item in args) {
```

```
            Console.Write(item);
```

```
        }
```

```
    }
```

```
}
```

```
class Program {
```

```
    static void Main() {
```

```
        A a = new A();
```

```
        a.MyDelEventHandler += A.Meth1;
```

```
        a.MyDelEventHandler += a.Meth2;
```

```
        a.Event();
```

```
    }
```

```
}
```

на экран будет выведено:

Примечание:

*Если возникнет ошибка компиляции, введите: ****

Если ошибок и исключений нет, но на экран не выведется ничего, введите: ---

Если возникнет ошибка исполнения или исключение, введите: +++

8.

Расположите по порядку жизненного цикла события Windows Forms (укажите все верные ответы):

- 1) Closed.
- 2) Activated.
- 3) Closing.
- 4) Load.
- 5) Deactivate.

9.

В результате выполнения фрагмента программы:

```
using System;

delegate int MyDel(int x);

class A {
    public event MyDel MyDelEventHandler;

    public static int Meth1(int x) {
        return x % 2;
    }

    public static int Meth2(int x) {
        return x % 3;
    }

    public static int Meth3(int x) {
        return x % 4;
    }

    public void Event() {
        Console.WriteLine(MyDelEventHandler(15));
    }
}

class Program {
    static void Main() {
        A a = new A();
        a.MyDelEventHandler += A.Meth1;
        a.MyDelEventHandler += A.Meth2;
        a.MyDelEventHandler += A.Meth3;
        a.Event();
    }
}
```

на экран будет выведено:

10.

**Выберите элементы управления, которые содержат в себе редактируемый текст
(укажите все верные ответы):**

- 1) TextBox.
- 2) CheckBox.
- 3) PictureBox.
- 4) Label.
- 5) ComboBox.

1	***
2	12345
3	91414
4	123
5	12345
6	34
7	56789
8	41315
9	3
10	145