Вопросы лабораторной №9 ДЕЛЕГАТЫ

**1. Что такое делегат? Как определить делегат?**

Делегат - это объект, предназначенный для хранения ссылок на методы(указатель на

функцию C++)

функции обратного вызова + без. Типов

**2. Назначение делегатов.**

1) возможности определять вызываемый

метод не при компиляции, а динамически

во время выполнения программы;

2) обеспечения связи между объектами по

типу «источник — наблюдатель»;

3) создания универсальных методов, в

которые можно передавать другие

методы;

4) поддержки механизма обратных

вызовов.

**3. Какие есть способы присваивания делегату адреса метода?**

простой и составной

**4. Поясните назначение метода Invoke.**

Выполняет указанный делегат в том потоке, которому принадлежит базовый дескриптор окна элемента управления.

**5. Что такое групповая адресация делегата?**

Создание списка, или цепочки вызовов для методов, которые вызываются автоматически при обращении к делегату.

**6. Как создать событие?**

Создание события в классе состоит из следующих частей:

* описание делегата, задающего сигнатуру обработчиков событий;
* описание события;
* описание метода (методов), инициирующих событие.

**7. Как события связаны с делегатами? Опишите и поясните схему взаимодействия.**

События построены на основе делегатов: с помощью делегатов вызываются методы-обработчики событий.

**8. Что такое лямбда-выражения? Приведите пример лямбда-выраженияс несколькими параметрами.**

Лямбда-выражения - упрощенная запись анонимных методов

параметр => выражение

(список\_параметров) => выражение

**9. Что такое ковариантность и контравариантность делегатов? Что это дает?**

***Ковариантность*** позволяет присвоить делегату метод, возвращаемым типом которого служит класс, производный от класса, указываемого в возвращаемом типе делегата. А ***контравариантность*** позволяет присвоить делегату метод, типом параметра которого служит класс, являющийся базовым для класса, указываемого в объявлении делегата.

**10. Поясните разницу между встроенными делегатами Action и Func**

Делегат Action является обобщенным, принимает параметры и возвращает значение void. Как правило, этот делегат передается в качестве параметра метода и предусматривает вызов определенных действий в ответ на произошедшие действия.

Func возвращает результат действия и может принимать параметры. Он также имеет различные формы: от Func<out T>(), где T - тип возвращаемого значения, до Func<in T1, in T2,...in T16, out TResult>(), то есть может принимать до 16 параметров.