1. Что такое делегат? Как определить делегат?

*Делегат* – это объект, предназн. для хранения ссылок на методы (указатель на ф-цию С++)

public delegate void D(int i)  
class Class1  
 private static void Hello(int i) { }  
 static void Main()  
 D del //созд. переменную делегата  
 del = Hello //присв. пер-ной адрес метода  
. //del = new D(Hello)  
 del.Invoke(4) //вызываем метод  
 //del(4)

2. Назначение делегатов.

1) возм. опр. вызываемый метод динамически при вып. проги  
2) связь между объектами «источник – наблюдатель»  
3) созд. универс. методов, в кот. мб переданы другие методы  
4) поддерж. механизм обратных вызовов

4. Поясните назначение метода Invoke.

Вызов делегата

5. Что такое групповая адресация делегата?

Создание списка, или цепочки вызовов, для методов, кот. вызываются авто- при обращении к делегату

6. Как создать событие?

**Событие** – элемент класса, позвол. ему посылать др. объектам уведомления об изменении своего состояния.

1) опис. делегат, задающий сигнатуру обработчиков события  
2) опис. событие  
3) опис. методы, инициирующие событие

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

1) обработка собы

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

Анонимные ф-и – безымянный кодовый блок, передаваемый к-ру делегата

delegate int Summator(int b);  
static int result = 0;  
Summator someDelegat = delegate (int number)  
{  
 for (i=0; i<=number; i++)  
 result +=i; //захват переменной  
 return result;  
}

Лямбда-выражения – упрощенная запись анонимных методов (нет return, типа)

(x,y) => x + y;  
i => i \* i

9. Что такое ковариантность и контр-вариантность делегатов? Что это дает?

Ковариантность – позв. методу возвр. объект : тип, возвр. делегатом  
Контрвариантность – позв. методу возвр. объект, более универсальный возвр. типа делегата

10. Поясните разницу между встроенными делегатам Action и Func.

Action – приним. параметры, возвр. void  
Func – назначает значения типа int