**1. What is a delegate in C#?**

a) A reference type variable that holds a reference to a method  
b) A value type variable that holds an integer value  
c) A keyword used for memory allocation  
d) A special type of class

**Answer:** a) A reference type variable that holds a reference to a method

**2. Which of the following correctly declares a delegate in C#?**

a) delegate void MyDelegate();  
b) public delegate int MyDelegate(string s);  
c) delegate int MyDelegate(string s);  
d) All of the above

**Answer:** d) All of the above

**3. What is the purpose of the delegate keyword in C#?**

a) To declare a pointer to a variable  
b) To define an event handler  
c) To define a method that can reference other methods with the same signature  
d) To create a new instance of a class

**Answer:** c) To define a method that can reference other methods with the same signature

**4. Which of the following correctly instantiates a delegate?**

a) MyDelegate d = new MyDelegate(MyMethod);  
b) MyDelegate d = MyMethod;  
c) delegate MyDelegate = new MyMethod();  
d) int d = delegate(MyMethod);

**Answer:** a) MyDelegate d = new MyDelegate(MyMethod);

**5. What happens if a delegate is used with a method that does not match its signature?**

a) Compilation error  
b) The method executes with a warning  
c) The method is automatically converted to match the signature  
d) The delegate ignores the method

**Answer:** a) Compilation error

**6. What is multicasting in delegates?**

a) Using multiple delegates with different signatures  
b) Assigning multiple methods to a single delegate  
c) Using delegates across multiple namespaces  
d) Calling a delegate multiple times in a loop

**Answer:** b) Assigning multiple methods to a single delegate

**7. How do you add multiple methods to a delegate in C#?**

a) Using the += operator  
b) Using the = operator  
c) Using the \* operator  
d) Using the Append method

**Answer:** a) Using the += operator

**8. How do you remove a method from a multicast delegate?**

a) delegate.Remove(MethodName);  
b) delegate -= MethodName;  
c) delegate.Delete(MethodName);  
d) delegate.RemoveHandler(MethodName);

**Answer:** b) delegate -= MethodName;

**9. What is the base class of all delegates in C#?**

a) System.Object  
b) System.MulticastDelegate  
c) System.Delegate  
d) System.Reflection

**Answer:** c) System.Delegate

**10. Which of the following statements about delegates is incorrect?**

a) Delegates can be used to define callback methods  
b) Delegates cannot be used to implement events  
c) Delegates can hold multiple method references  
d) Delegates are type-safe function pointers

**Answer:** b) Delegates cannot be used to implement events