### -- Rawlston Morton

### --File: .CH 20 C#

1. What is a delegate? Explain delegates in terms of pointers and reference types.

- A delegate is a reference to a method. You can assign a reference to a method to a delegate in much the same way that you can assign an \*int\* value to an \*int\* variable.

2. How do you declare a delegate? Include a discussion of types, return values, names, and parameters.

- public delegate void PrintPresDel(President p); the common language runtime (CLR) knows that this is a delegate, so it retrieves the method that the delegate references and runs that instead.

3. How do you create instances of delegates and assign values to the instance? What are the values?

- Del del4 = name => { Console.WriteLine($"Notification received for: {name}"); };

4. How do you invoke a method that has been added to a delegate?

- Using the += compound assignment operator.

5. What is an event? Why do we have events?

- An event is a notification sent by an object to signal the occurrence of an action. Events you can use to define and trap significant actions and arrange for a delegate to be called to handle the situation.

6. How do you declare events?

- You declare an event in a class intended to act as an event source. You declare an event similarly to how you declare a field. However, because events are intended to be used with delegates, the type of an event must be a delegate, and you must prefix the declaration with the \*event\* keyword.

7. How do delegates recognize event subscriptions? How do you unsubscribe an event from a delegate?

- You subscribe to an event by using this \*+=\* operator. You use the –= operator to detach a delegate from an event.

8. . How do you raise an event? How do you declare code that raises an event?

- You can raise an event by calling it like a method. When you raise an event, all the attached delegates are called in sequence.

9. Explain with specificity what happens when an event fires and that event has been attached to a delegate.

- In fact, there is no guarantee that the processing performed by the delegated methods has completed by the time the dialog box