**Name : Indra Bhurtel**

1. False. The statements *Watermelon w1;* and *Apple a1;* will be creating objects of the respective classes and hence are the valid lines of codes.
2. True. Here, the class Watermelon inherits all the properties of Food class publicly so that all the properties of Food class are also in the watermelon class. So, it is possible to refer to the members of the Food class using Watermelon object w1. So, *e1.discount(w1);* is valid line of code.
3. True. The discount function of class Employee takes the argument of Food class so that *e1.discount(f1);* would be a valid line of code.
4. False. The function *set\_seed()* in the Watermelon class is necessary. It sets the seedless to both true and false. If the input argument ‘a’ is less than 3, it is true else will be false.
5. True. The function would be available as protected in the Apple class which inherits the function *give\_info()*, from Food class. Hence, *a1.give\_info(2.99);* would be valid line of code.
6. False. Price is defined as the data member of Food class while number and color are not defined as the data member of Food class. So, *f1.number=3;* and *f1.color="blue";*, both would not be valid line of code.
7. False. The price is defined as protected member so, it is not possible to be accessed outside the class. So, *f1.price=2.99;* and *w1.price=2.99;* would not be a valid line of code.
8. False. The properties of Watermelon cannot be accessed from Food class. So, *e1.check\_watermelon(w1, f1);* would not be a valid line of code.
9. False. The Apple inherits from Food class as a protected. So, it is not possible to call the members of Food class outside the Apple class with the object of Apple class. So, *a1.set\_seed();* would not be a valid line of code.
10. False. Here, the number is the public member of Grape class so that number can be called from anywhere from the program with the object of the Grape class. So, *g1.number=4;* would not be a valid line of code.