Review: STRUCTURES

1. True/False

(A). Structures cannot have two fields with the same name. T / F
(B). Structures cannot have two fields of the same type. T / F
(C). Structures are derived types (based on existing types) T / F
(D). All fields in a structure must be of the same type. T / F
(E). A field in a structure cannot itself be a structure. T / F
(F). A local variable can have the same name with the name of a field in a structure. T / F

Structures and Functions: NOTE: Questions 2, 3, and 4 are related.

2.

(A). Write the declaration of a **Movie** structure (at least 5 fields of your choice).
(B). Define and initialize the structure variable **m**.

3. The **exchange** function does not work properly. Its task is to exchange (or swap) two **Movie** structure variables. Make all necessary corrections.

```
void exchange ( Movie m1, Movie m2 )
{
    Movie hold;

    hold = m1;
    m1   = m2;
    m2   = m1;
}
```

4. Use the **exchange** function (defined and corrected above) to randomize a sorted list of movies. For each movie in the list generate a random position then exchange it with the movie at that random position.

```
void randomize ( Movie list[], int size )
{
```

Review: STRUCTURES

Nested Structures: NOTE: Questions 5 to 8 are related.

5. First give the graphic representation of the structures describe below. The first structure is named **`Order`** and it has of two fields: the first field is **`qty`** (for quantity, an integer), and the second field is **`book`**, a **`Book`** structure. The **`Book`** structure has four fields: **`title`**, **`author`**, **`price`**, and **`year`**.

6. Declare the structures described above.

7. Define **`ord`**, an **`Order`** structure variable and initialize it. Let's say an instructor wants to order 40 books: Essential C++ by S. Lippman, $44.61, published in 1999.

8. Write a function that receives an **`Order`** structure and prints its fields to the screen.