CS 161 Winter 2020 Practice Final Exam Questions: Structs and Recursion

Review the questions on Midterms 1 and 2 (and the associated review questions) for content through week 7.

Additional questions on structs and recursion (weeks 8 and 9) follow.

Use this struct definition for questions 1 through 5:

```c
struct animal {
    int n_eyes;
    float weight;
    string name;
};
```

1. Given the following declaration:
   ```c
   animal* deer = new animal;
   ```
   Choose the statement that will set the deer's number of eyes to 1.
   A. `deer.n_eyes = 1;`
   B. `animal.n_eyes = 1;`
   C. `deer->n_eyes = 1;`
   D. `(&deer).n_eyes = 1;`

2. Given the following declaration:
   ```c
   animal bear = { 2, 17.5, "bear" };
   ```
   Choose the statement that will print the bear's weight.
   A. `cout << bear.weight << endl;`
   B. `cout << animal.weight << endl;`
   C. `cout << bear->weight << endl;`
   D. `cout << (&bear).weight << endl;`

3. Given the following declaration:
   ```c
   animal* farm = new animal[20];
   ```
   Choose the statement that will set the name of the animal at index 2 to "pig".
   A. `&farm[2].name = "pig";`
   B. `farm[2].name = "pig";`
   C. `animal[2].name = "pig";`
   D. `farm[2]->name = "pig";`

4. Given two `animal` variables named `fox` and `badger`, choose the expression that evaluates to true if the badger weighs more than the fox.
   A. `fox > badger`
   B. `badger > fox`
   C. `fox.weight > badger.weight`
   D. `badger.weight > fox.weight`

5. (A: True, B: False) A recursive function cannot use a `for` loop inside its definition.
6. (A: True, B: False) The **base case** describes the condition in which the recursion stops.

7. Assume that you want to write a recursive function that prints every letter from character c down to 'a':

   ```cpp
   void print_letters(char c) {
     if (_________①_________) { // What would be an appropriate base case to go in the blank marked ①?
       cout << c << endl;
     } else {
       cout << c << " ";
       __________②____________; // Select the best recursive call to go in the blank marked ②.
     }
   }
   ```

   A. c == 0
   B. c != 0
   C. c == 'a'
   D. c != 'a'

8. Given the same task as in question 7, select the best recursive call to go in the blank marked ②.

   A. `print_letters(c);`
   B. `print_letters(c-1);`
   C. `print_letters('c');`
   D. `print_letters();`

9. Given this definition of function **play()**:

   ```cpp
   float play(int x) {
     if (x == -1) {
       return 1.5;
     } else {
       return 3.6 + play(x-1); // What will play(0) return?
    }
   }
   ```

   A. 3.6
   B. 5
   C. 5.1
   D. 6.6

10. Given the definition of function **play()** in question 9, how many times will the function **play()** be called, if we start by calling **play(2)**?

    A. 1
    B. 2
    C. 3
    D. 4