CS 162 Recitation 7 Worksheet

Midterm Reflections
1. A lot of the concepts in the class are abstract. Instead of going further without fully understand those concepts, we want you to provide us with terms or concepts you don’t think you understand yet and provide your current definition or understanding of the term or concept to clear any misconceptions or affirm your knowledge. Your answers can be either specific vocabulary terms or more general ideas. We’ll try to address some of these concepts and/or terms in future lectures/recitations.

2. What did you struggle with the most on Exam I?

Common mistake 1: Redeclaring variables in the constructor.

```cpp
class Coffee {
private:
   string name;
   float small_cost;
   float medium_cost;
   float large_cost;
public:
   Coffee();
   ...
};

Coffee::Coffee() { // constructor
   string name = “Latte”;
   float small_cost = 1.5;
   float medium_cost = 2.5;
   float large_cost = 3.5;
}
...
```

1. Analyze the code above. Would the member variables of a Coffee object be initialized after calling the constructor? Why or why not? How would you fix the code?
Common mistake 2: Creating extra object when working with class composition.

class Shop {
private:
    Menu m;
    ... 
public:
    void load_menu(); //populate coffee(s) detail
    void print_menu();
    ...
};

class Menu {
private:
    Coffee* coffee_arr;
    int num_coffee;
public:
    void load_all_coffee();
    void print_all_coffee();
    ...
};

int main () {
    Shop s;
    Menu m;
    m.load_all_coffee();
    m.print_all_coffee();
    s.print_menu();
    ...
}

2. Assuming all member functions are correctly implemented. Is the menu object m within the Shop s loaded/populated? Why or why not? How would you fix the code?
Common mistake 3: A chain of accessor calls.

```cpp
void Shop::print_menu() {
    for (int i = 0; i < m.get_num_coffee(); i++) {
        cout << "Name: " << m.get_coffee_array()[i].get_name() << endl;
        cout << "Small cost: " << m.get_coffee_arr()[i].get_small() << endl;
        cout << "Medium cost: " << m.get_coffee_arr()[i].get_medium() << endl;
        cout << "Large cost: " << m.get_coffee_arr()[i].get_large() << endl;
    }
}
```

Shop s;
s.print_menu();

3. What is the issue with the `print_menu()` function above? Why is it a bad idea to use a chain of accessors to get the internals of Coffee class from the Shop? How would you fix the code?

4. Read assignment 4 document. Discuss with your classmates and draw the hierarchy of all classes. What is the parent class? What are the child classes?