CSCI 169

Project Phase 3

Justin Li

1. Yes, Ruby supports OOP. It handles inheritance very similarly to what we learned in both C++ and Java, where you can have subclasses that derive from parent classes (using the symbol ‘<’). The language does support polymorphism, but because variables themselves don’t have individual types, you can simply call the same named function for the objects of both the subclasses and parent class. The language doesn’t support multiple inheritance, but instead, kind of like Java interfaces, it uses “mixin modules” which act just like interfaces in order to be able to implement specific behaviors for different classes without having to inherit from it. Although Ruby includes OOP, unlike Java (and more like in C++), you don’t have to make everything a class, but are still able to include it (an optional feature).
2. In Ruby, memory allocation and deallocation are automatically done, as the language uses garbage collection. Unlike C++, Ruby doesn’t use manual memory management. Memory allocation for variables is very similar to how it is in Python, where the memory associated with a variable has potential to change if you reassign it. However, Ruby has something called “stop-the-world” garbage collection, where it must stop the program when freeing memory so that it doesn’t interfere with memory allocation. Objects in Ruby are true objects, not just references.