JOURNEY TO MARS

There and Back Again

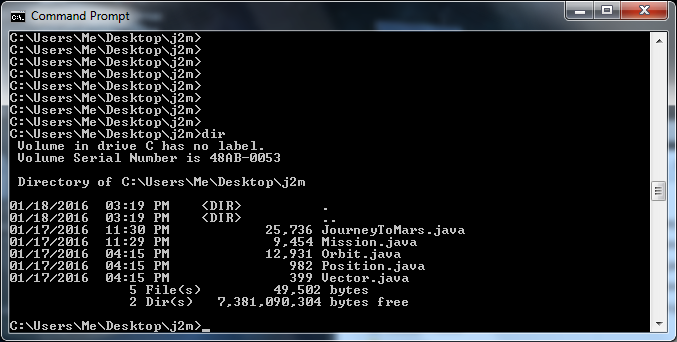
This assignment will introduce you to how software projects are (often) run in the *real* world. Specifically, you will leverage your current programming skills to understand, troubleshoot, and modify an existing piece of software (ah, yes…the joys of working with someone *else’s* code!). Over the course of this assignment, you will be given a series of tasks. Many of these tasks will originate from requests by the *end user*. As you will soon see, the *end user* is mostly concerned with “what”, and less so with “how” (or “why”). Hence, many of these tasks will lack sufficient technical detail, and it will be up to you to fill in the gaps. Take your time, *think* before you code, and don’t be afraid to ask questions. Good luck!

**Level 1**

Congratulations! Fresh out of college, computer science degree in hand, you have landed your dream job at the NASA Jet Propulsion Laboratory (JPL) in sunny Pasadena, CA. Not that you would know much about the sun, though. Since your arrival two weeks ago, you have been working in a dark, window-less, basement-level “cube farm”. Signs of life are scarce so far, but they did place you close enough to the restroom to occasionally hear the toilets flush. You know there must be *someone* out there…at least you hope so!

Your suspicions are confirmed one day when your boss suddenly peers above your cubicle wall, throws over a stack of hand-written notes and a floppy disk (yes, a floppy!), and makes a hasty retreat. You hear him in the distance, like a passing train, bellow, “Looks like we’re sending humans to Mars…again! Review this orbit calculator from 2004 and have an SDD on my desk by COB today!”

SDD? COB? SMH!!!! A quick Google search reveals that an SDD is a “Software Design Document” and COB means “Close of Business”. Uh-oh, there’s not much time, you better get crackin’! Your workstation does not have a floppy drive, but you find an old clunker in the abandoned cube next to yours. You brush the cobwebs aside, boot up the old dinosaur, and *feed* it the floppy. This is what you see:



**Level 1A:** Review the five Java files shown in the previous figure. In the space below, list each associated class (e.g. JourneyToMars, Mission, etc.) and include one sentence that describes what function the class performs (HINT: Don’t waste too much time trying to understand *everything* about a given class. Look at the comments included in each Java file to get a general idea of what the class does.)

**Level 1B**: Draw a simple diagram that illustrates how the five classes are related

**Level 2**

**Level 3**

**Level 4**

**Recursion Practice –** Create a program that produces the following output to the screen. Each item sent to standard output should be a value stored in a variable. Each line should be produced by a separate recursive function:

1, 2, 3, 4, 5, 6, 7, 8, 9, 10,

10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0,

2, 4, 6, 8, 10, 12, 14, 16, 18, 20,

1, 2, 4, 8, 16, 32, 64, 128, 256, 512,

1, 3, 7, 15, 31, 63, 127, 255, 511,

1, 4, 10, 22, 46, 94, 190, 382, 766,

1, 2, 5, 14, 41, 122, 365, 1094,

1, -10, 100, -1000, 10000, -100000,

1, -1, 1, -1, 1, -1, 1, -1, 1, -1,

2, 4, 16, 256, 65536,

**Level 5**

**BabyNamer –** Complete the following methods. You do not need to use recursion to program them.

* eliminateRepetitions () – eliminates all repetitions of names in fullNames  
   so  
   {“Mark Jones”, “Sally Doe”, “Mark Jones”, “Petter Pipper”}  
   becomes  
   {“Mark Jones”, “Sally Doe” , “Petter Pipper”}
* eliminateName(String name) – eliminates all names in fullNames that contain the specified   
   so when eliminateName(“Doe”) is called on the following  
   {“Mark Jones”, “Sally Doe”, “Mark Jones”, “Petter Pipper”}  
   it becomes  
   {“Mark Jones”, “Mark Jones”, “Petter Pipper”}
* eliminateNames(ArrayList<String> names) – eliminates all names in fullNames that contain the specified names  
   so when eliminateNames({“Doe”, “Petter”}) is called on the following  
   {“Mark Jones”, “Sally Doe”, “Mark Jones”, “Petter Pipper”}  
   it becomes  
   {“Mark Jones”, “Mark Jones”}