

# CS100 J & M Spring 2001

## Project 1: A Little Bit Of Everything

### Due Tuesday 1/30

## 1. Objective

---

---

Completing all tasks in this assignment will help you:

- find information and understand policies in both CS100J and CS100M
- learn the rudiments of editing files for programs
- run your first Java and MATLAB programs

First skim, and then, carefully read the entire assignment before starting any tasks!

## 2. Information and Policies

---

---

**Topics:** course policies and procedures, accessing CS100 webpages and newsgroups, editing text files

**Tasks:**

- Access either the CS100J or CS100M newsgroup.
- Save the article with the subject line **PROJECT 1 QUESTIONS** to a file.
- Edit the file with any text editor or word processor.
- Write the following information at the top of the file:  

<b>Name:</b> <i>your name</i>	<b>Partner:</b> <i>partner's name</i>
<b>ID:</b> <i>your ID</i>	<b>ID:</b> <i>partner's ID</i>
<b>Section:</b> <i>your section's instructor,day,time</i>	<b>Section:</b> <i>partner's section's instructor,day,time</i>
<b>Date:</b> <i>due date</i>	
<b>Project:</b> <i>number and title</i>	
- Skip a line, write **Project 1: Part A**, and skip another line.
- Answer the questions inside this file by writing your responses after each question.
- Be neat! Format the file following the course guidelines for e-mail and newsgroup posting.
- Save your work as ASCII text if you used a word processor.
- Print your file.

## 3. MATLAB

---

---

**Topics:** setting your path in the MATLAB command window, using MATLAB's help function, generating a plot of data

**Tasks:**

Assume that you are conducting an experiment that involves measuring how far a spring will stretch (deflection, centimeters) for various weights (Newtons). Suppose you have collected the following data:

Test	1	2	3	4	5	6	7	8	9	10	11
Weight (N)	0	10	20	30	40	50	60	70	80	90	100
Deflection (cm)	0	0.7	2.2	2.9	4.1	4.8	5.9	7.0	8.3	9.1	9.9

Write a MATLAB script called **p1PartB.m** that plots this data. You will fill in missing portions of the code, including a prediction of a reasonable deflection for a weight of 75N. Print out the completed code and figure.

## 4. Java

---

---

**Topics:** downloading files, running code, editing program, experimenting, debugging, printing a window

**Tasks:**

From either CS100 website, download the file called `p1PartC.java` from the [Projects](#) page. Run the code. You will see a window with three “snowpersons” in the order *red, green, blue*. Also, the middle snowperson is missing arms. Rewrite the program such that the snowpersons will appear in the order *blue, red, green* and so that the middle snowperson has arms. Print out the revised code and the window (or screen snapshot).

## 5. Algorithms

---

---

**Topics:** problem solving, algorithms, pseudocode

**Background:**

Programming is sometimes referred to as *automating problem solving*. A program “tells” a computer the steps to follow in a language that the computer “understands.” But, the programmer must first devise those steps in a *algorithm*, which is a “recipe” for the problem in question. An algorithm is composed of a sequence of steps that someone (or something) would follow. For each task, below:

- Create a new text file with the following information at the top of the file:  

<b>Name:</b> <i>your name</i>	<b>Partner:</b> <i>partner’s name</i>
<b>ID:</b> <i>your ID</i>	<b>ID:</b> <i>partner’s ID</i>
<b>Section:</b> <i>your section’s instructor,day,time</i>	<b>Section:</b> <i>partner’s section’s instructor,day,time</i>
<b>Date:</b> <i>due date</i>	
<b>Project:</b> <i>number and title</i>	
- Skip a line, write **Project 1: Part D, Task** (either 1 or 2), and skip another line.
- Do the task and print out the file.

**Task 1:**

Using 1-2 pages of single-spaced, 12-point fixed-width font, write an algorithm that a robot would follow to tie your shoe laces. Complete steps should read as a sentence and end with punctuation. To make your algorithm clear, indent steps that are substeps of other instructions. When you need to make choices or repeat tasks, indent those steps. For instance,

**Find matching shoes.**

**While matching shoes not found**

**Search closet for 1st shoe.**

**Search closet for 2nd shoe.**

**While 1st shoe not equal 2nd shoe and shoe-pile not exhausted**

**Pick another 2nd shoe.**

**Bring shoes to human owner.**

...

**Task 2:**

Using at most 1 page of single-spaced, 12-point fixed-width font, write an algorithm that the course instructor could use in lecture to find the most common birthday (e.g. “50 students share February 28 as their birthday”). Although now the algorithm is for a human, not a computer, you should still strive for clarity and precision. Also, try to make your algorithm efficient by reducing unnecessary or “expensive” computations.

## 6. Submitting Your Work

---

---

Follow the submission guidelines stated on the [Projects](#) page for either CS100J or CS100M.