Computer Science 507
Software Engineering
The College of Saint Rose
Spring 2015

Lab 4: Build Management
Due: 6:00 PM, Monday, February 17, 2015

You may work alone or in groups of size 2 or 3 on this lab. Only one submission per group is needed.

Getting Set Up

Create a document where you will record your answers to the lecture assignment and lab questions. If you use plain text, call it "`lab4.txt`". If it's a Word document, you can call it whatever you'd like, but when you submit, be sure you convert it to a PDF document "`lab4.pdf`" before you submit it.

Guest Speaker Reactions

Note: this portion must be done individually and is graded separately.

Participate in the discussion of this week's speaker, Jorge Diaz, on the speaker blog. To earn full credit, you should make at least 3 comments on this week's post or replies to comments (at least 1 should be a reply, and at least 1 should be posted within the day following the talk). Comments may address any aspect of the speaker's presentation, including how it relates to topics from other parts of the class. You are welcome to start this process during the talk, if you wish. (5 points)

Lecture Assignment Questions

We will usually discuss these questions at the start of class on the lab due date, so no credit can be earned for late submissions of lecture assignment questions.

> **? LA Question 1:**
> Sommerville Exercise 7.3, p. 202-203 (5 points)

> **? LA Question 2:**
> Sommerville Exercise 7.10, p. 203 (5 points)

Using the `make` Utility

Any non-trivial software development involves many iterations of editing, compiling, linking, and running your programs. The code will be spread across multiple files. The most common mechanism for managing this process when programming in C in a Unix environment is the `make`

utility. Its utility is not restricted to C programs. The actions of make are specified by rules in a Makefile.

Copy the following example to your account on mogul.strose.edu:

See Example:
/home/cs507/examples/make-example

You should find a small C program that demonstrates the use of multiple source files and a Makefile. Compile the program with make.

> **? Question 1:**
> What is the output produced when you run make? (2 points)

Now, look at the rules and the description in the Makefile.

> **? Question 2:**
> Briefly describe how make uses the rules in the Makefile to produce the executable main. Be sure to include the series of targets, their dependencies, and the commands used to satisfy those dependencies for each target. (7 points)

Apache Ant for Java Build Management

While make can be used to manage builds of a Java project, many Java projects use a tool called Apache Ant (http://ant.apache.org/. Read the first section on that page to get a brief description of Ant.

Our goal today is only to see the basics of Ant, so we will use an example that can build a "Hello, World" Java program using Ant.

- Start at http://passion4java.blogspot.com/2009/10/hello-world-example-for-a html.

- In your subdirectory for this lab, create another subdirectory named ant. Inside of ant, create subdirectory named src, and in that directory, place the Java program shown in blue near the top of the page as the file AntHelloWorld.java (replacing the printout with a message of your own).

- Then, in your ant subdirectory, create a file build.xml which contains the red text under "Hello world build script".

- Still in your ant directory, issue the command ant.

> **? Question 3:**
> What output is produced? (2 points)

- Run the program with the command

```
java -cp AntHelloWorld.jar com/passion4java/helloworld/ant/AntHelloWorl
```

> ? **Question 4:**
> What output is produced? (2 points)

You have now compiled a Java program using Ant and run that Java program from a jar (Java ARchive) file.

> ? **Question 5:**
> Paste in the output of the command `ls -lR` from inside your `ant` directory. (2 points)

Submitting

Before 6:00 PM, Monday, February 17, 2015, submit your lab for grading. Package up all required files into an appropriate archive format (`.tar.gz`, `.zip`, and `.7z` are acceptable) and upload a copy of the using Submission Box at `http://sb.teresco.org` under assignment "Lab4".

Grading

This assignment is worth 25 points, which are distributed as follows:

Feature	Value	Score
LA 1: (7.3)	5	
LA 2: (7.10)	5	
Question 1: (make output)	2	
Question 2: (make description)	7	
Question 3: (ant output 1)	2	
Question 4: (ant output 2)	2	
Question 5: (ant directory listing)	2	
Total	25	

Note that the 5 points for participating in the speaker discussions is awarded separately.