

COMP 3705 – Winter 2009

HW#1: Chapter 1 and 2 Exercises

Instructor:

Prof. Matt Rutherford – mjr@cs.du.edu

Due by:

5:pm on January 19, 2009

Section 1.1, Exercise 2

The following exercise is intended to encourage you to think of testing in a more rigorous way than you may be used to. The exercise also hints at the strong relationship between specification clarity, faults, and test cases.

1. Write a Java method with the signature: `public static Vector union (Vector a, Vector b)` The method should return a Vector of objects that are in either of the two argument Vectors.
2. Upon reflection, you may discover a variety of defects and ambiguities in the given assignment. In other words, ample opportunities for faults exist. Identify as many possibly faults as you can. (Note: Vector is a Java Collection class. If you are using another language, interpret Vector as a list.)
3. Create a set of test cases that you think would have a reasonable chance of revealing the faults you identified above. Document a rationale for each test in your test set. If possible, characterize all of your rationales in some concise summary. Run your tests against your implementation.
4. Rewrite the method signature to be precise enough to clarify the defects and ambiguities identified earlier. You might wish to illustrate your specification with examples drawn from your test cases.

Section 2.1

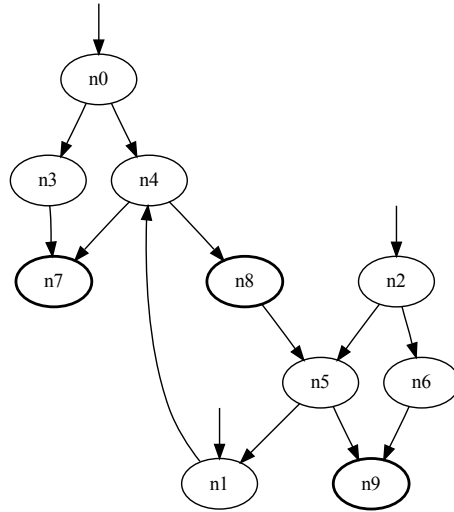


Figure 1: Figure 2.2

Exercise 1

Give the sets N , N_o , N_f , and E for the graph in Figure 2.2.

Exercise 2

Give a path that is not a test path in Figure 2.2.