CSCI 330 Fall 2001

Homework 2

Assigned Monday, 01/9/10. Due Monday, 01/9/17

The questions refer to the following paper: Rajlich, V., N. Wilde, M. Buckellew. and H. Page. "Software Cultures and Evolution." *Computer*, 34, 9 (September 2001), pp.24–28. Please be concise.

1. List and describe briefly the three models of program development explained in class on September 5. Which of the three models do the authors present at the beginning of their paper? **Answer:** The three models are: Waterfall (with strict sequential phases: requirements analysis and specification, software design, implementation (coding), verification and validation (certification), maintenance. RUDE (Run Understand Debug Edit). Spiral: iterative version of waterfall, with prototypes built and critiqued at each iteration.

2. Map the requirement of "effective comprehension" to one or more of the goals and qualities of programming languages that are described in Section 1.5 of your textbook. **Answer:** Important for readability, etc.

3. The textbook mentions several "tools provided by the software development environment." List them. The paper describes briefly a tool for software maintenance. What is it called? What is it used for? **Answer:** Editors, compilers, simulators, interpreters, linkers, and debuggers (p.4-5). Vifor, which is a browser.

4. What is "code decay" and why does it make maintenance of software more difficult? **Answer:** See left column of p.27 of paper.

5. Explain the use of "common blocks" in the Convert program. Common blocks address well a certain quality of software that was of paramount importance when Convert was originally written, but have a substantial negative impact on a different quality of software. Explain this trade-off. **Answer:** Space efficiency. Readability (a component of maintainability).

6. The paper mentions the GOTO statement in Fortran and C. Does Java have a GOTO statement? **Answer:** No.

7. List the five "obsolete program plans" described in the paper. **Answer:** The line printer carriage control character, input/output batching, scratch files, packing data into an integer, binary input/output.

8. What is the solution that the authors propose to solve the code decay problem? In your opinion, is this a practical solution? Explain your answer. **Answer:** Document the prevailing practices considered acceptable

at the time the program was written. (See right column, p.26 of paper.) In my opinion, no, because it is impossible for a programmer to anticipate change in programming environments that would make a program plan obsolete.