# CSCE 330 Fall 2005

## Homework 2

Assigned Monday, 05-08-29. Due Wednesday, 05-09-07

1. The questions below 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. (A local copy of this paper is available on the course web site.) Please be concise.

   (a) List and describe briefly the three models of program development explained in class on August 24. Which of the three models do the authors present at the beginning of their paper?

   **Answer:** Waterfall, Spiral, RUDE model.

   (b) 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:** Readability, Simplicity, Factoring, Locality.

   (c) 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, debuggers. The tool described in the text is the browser VIFOR.

   (d) What is "code decay" and why does it make maintenance of software more difficult?

   **Answer:** Accumulation of factors that make software changes difficult; it drives software from the evolution phase to the servicing phase. It makes maintenance difficult because the programmers that are going to perform the maintenance will probably be from a different culture, and have difficulty understanding and interpreting the code.

   (e) 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:** Old computers had a very small amount of memory. In order to overcome this space limitation, Convert holds most of the data in the large global blocks that many subroutines share. This improves space efficiency but make programs hard to maintain.

   (f) The paper mentions the GOTO statement in Fortran and C. Does Java have a GOTO statement?

   **Answer:** No (although `goto` is a reserved word in Java).

(g) List the five "obsolete program plans" described in the paper.

**Answer:** input/output batching, scratch files, packing data into an integer, binary input/output, using the first column of output as a carriage control character for output to line printers.

(h) 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:** The solutions that the authors propose is the systematic documentation of the programming practices of the old culture. This will help programmers, but it will be very difficult to document every single detail of how and why a program was implemented in a certain way.

2. Short answer questions on history.

(a) Which major language is based on the principle of orthogonality? **Answer:** Algol 68

(b) Which major language was designed by John McCarthy? **Answer:** LISP

(c) Name two languages designed by John Backus. **Answer:** FORTRAN and FP. (Note: ALGOL 60 was designed by a committee, of which John Backus was a member.)

(d) What has been the dominant programming language for business applications from 1960 to 1990 (at least)? **Answer:** COBOL.

(e) Name two logic languages. **Answer:** Prolog and Mercury (or Goedel or CLP or Eclipse or some others).

(f) Which two of the six languages you listed above were designed by committee? **Answer:** Algol 68 and COBOL.