

CS 350: COMPUTER SYSTEM CONCEPTS

PROGRAMMING ASSIGNMENT 1

DATE ANNOUNCED: FEB. 4, 2010

DUE : FEB. 18, 2010, 11:59PM (PROGRAMS & DOCUMENTATION – SUBMIT COMMAND)
: FEB. 16, 2010, 5:00PM, (HARDCOPY OF PROGRAMS & DOCUMENTATION – IN CLASS)

MAXIMUM MARK: 100

Problems are taken from the recommended book for the C Language Part:

H.M. Deitel and P.J. Deitel, *C: How to Program*, Prentice Hall, Upper Saddle River, NJ, 1994, 2001.

This assignment description and copies from relevant parts of the book are available on the table outside my office.

Question 1 (Pythagorean Triples) 15 marks
Exercise 4.27, pp. 139

Question 2 (Perfect Numbers) 15 marks
Exercise 5.26, pp. 188

Question 3 (GCD – greatest common divisor) 28 marks
3a. Exercise 5.42, pp. 193 (GCD with recursion)
3b. Exercise 5.42, pp. 193 (GCD without recursion, see also Ex. 5.29, pp 192)

Question 4 (The Knight's Tour Problem – using Accessibility Heuristic) 32 marks
Exercise 6.24, pp. 243

You are required to do the following:

- (1) Print out an empty chessboard at the beginning
- (2) Display the Table of Accessibility Numbers at the beginning
- (3) After each successful placement, show the status of the chessboard. That is, indicate the sequence of steps that was used to visit all the positions on the board. For each board position, display the step at which the position was visited (i.e. 1, 2, 3, ..., 64).

Your program can stop after finding and displaying one valid sequence of moves.

[**Hint :** You might think of using different functions. For instance, we could use a different function for each of the following: compute the Table of Accessibility Numbers, print (display) the **chessboard**, check for valid placement, etc.]

You can represent the 8x8 chessboard with a two-dimensional array.

Question 4b: Bonus Mark (The Knight's Tour Problem – using Brute Force) 5 marks
Exercise 6.25, pp. 245

Programming Style 10 marks

Marks will be given for good programming style, exception handling, and special considerations for improving the program beyond the stated requirement. The maximum here will be **10 marks**.

Algorithm Design Phase. For each problem above, you are required to include a documentation of the algorithm you used to solve the problem, before you started to code. The algorithm will carry about **30% of the mark**, before we start to consider the programs. This is different from the 10 marks for programming style. Submit algorithm design phase as a text file, or a file in .pdf or Word format. Also, you are required to submit hard copies of your assignment in class as indicated above.

Please submit your assignment using the usual *submit* command:

<http://www.csee.wvu.edu/~adjeroh/classes/cs350/assessment/submit.html>