CS100J Spring 2001: Project 3 Grading Guide

Notes
- Please carefully review all notes written on your grading form and project.
- Find the codes for these notes below.
- Try to understand why you received the note so that you may avoid it on your next project.
- * means the item is worth twice

Scores
- Bonus may be applied for exemplary work or doing additional tasks
- Let c and s be the number of correctness and style: see table, below
- For each program not included, remove one correctness and style point

category	Points					
	0	**1**	**2**	**3**	**4**	**5**
correctness	nothing turned in	$c \geq 16$	$10 \leq c \leq 15$	$6 \leq c \leq 9$	$5 \leq c \leq 3$	$0 \leq c \leq 2$
style	nothing turned in	$s \geq 16$	$10 \leq s \leq 15$	$6 \leq s \leq 9$	$5 \leq s \leq 3$	$0 \leq s \leq 2$

1. General
(s1a) (not counted this time) grading form provided for each partner, as coversheet
(s1b) (not counted this time) grading form properly filled out and signed
(s1c) title sheet and table of contents provided (OK if unified)
(s1d) pages numbered and properly bound
(s1e) all work typed
(s1f) lines of text/code not chopped off or misaligned

2. Program
(c2a) alternates between player 1 & 2 to get their moves
(c2b) asks again if input location is out of 9 possible positions
(c2c) asks again if there is already a mark at the input location
(c2d) correctly updates the board according to players' inputs
(c2e) prints the current board after each turn
(c2f) * corretly checks if either player won the game
(c2g) ends the game when either player wins
(c2h) * ends the game after 9 moves
(c2i) * enables user to repeat playing game
(c2j) variables properly initialized for the next game
(c2k) nested control structures are correct
(c2l) **break** for each case of **switch** (if any switch is used)
(c2m) **equals()** used for string comparison (not **==**)
(c2n) (output) session where neither player wins
(c2o) (output) session where either player wins
(c2p) (output) shows the user(s) can repeat playing games
(c2q) (output) shows how illegal inputs are handled
(s2a) no arrays used
(s2b) no string methods used except **equals()**
(s2c) no non-static methods or outside methods/classes used
(s2d) * methods used to subdivide "larger" tasks and/or perform redundant tasks
(s2e) * methods defined in reasonable way (not too many methods, methods handle general tasks)
(s2f) meaningful, readable variable/method names used (preferably verbs)
(s2g) * excessive redundancy or needless complication avoided
(s2h) code properly indented
(s2i) title block included at top of the program code
(s2j) * **comments provided above of each method**
(s2k) comments for major varialbes and lines of code
(s2l) clear, readable, and consistent commenting style

3. Miscellaneous
(c3a) miscellaneous
(c3b) miscellaneous