Continuo is a game for 2-5 players in which they take turns to place coloured cards on the table and score points according to how well they do it. The aim of the programming contest was to implement play of a single-player variation of the game, in Common Lisp.

Details of the contest were posted on February 3rd. The first entry was received the following day. A total of eight entries were received, half of them in the last week before the closing date of March 3rd. Many thanks to everyone who took the time to have a go at this.

The judges have awarded prizes as follows. In some cases we have made an "honourable mention" of someone who did almost as well, or who is not eligible for a prize because they are not attending ILC 2007.

First half-decent entry received: **Frank Buss**.

Fastest half-decent entry: **Vladimir Sedach**, with honourable mentions to **Peter Salvi** (non-attendee) and **Guido Witmond** (whose entry played very fast but not so well). The fastest entries could play several games per second; the slowest took two minutes per game. No entries used any particular speed tricks. Some entries used an array to represent the playing board, and some used a list of played card positions. Interestingly this distinction was not reflected in the speed of play. Similarly there were entries using CLOS for cards, positions, etc, and those using old-school lists.

Guido Witmond receives an award for entering his first Lisp program ever.

The judges were not minded to award any prizes for elegance. The Lisp submitted was of a generally high quality, but none of it was startlingly cool or hacky.

For relying on the non-nil return value of (digit-char-p), we feel that **Christophe Rhodes** deserves a prize for obscure use of Lisp. An honourable mention in this category goes to **"larryl"** for an entry which included a one-megabyte file of floating-point numbers, but which had by far the lowest score at actually playing the game.

The judges are pleased to note that nobody used non-determinism in their entry this year.

Facilities: **Christophe Rhodes**, for including a McCLIM interface. Honourable mentions to **Peter Salvi** and **Frank Buss**, who provided output in graphical file formats.

Added Value: an honourable mention to **Peter Salvi**, for providing head-to-head play between four separate algorithms.

The judges used particular diligence in assessing scores. A test harness was used to check the self-scoring of the entries, and hand-checking to test the test harness. Four of the entries were found to be scoring their play incorrectly (not all overstating their scores). To measure the highest scoring entries, the scores from the test harness were used.

Highest score: **Vladimir Sedach**, with an honourable mention to **Peter Salvi**, who might have won this prize if he had registered for the conference, and also to **Christophe Rhodes** and **Frank Buss**. These 4 entries all played a reasonable game (scoring consistently over 800).

Scores and timings were assessed by generating ten random orderings of the cards. Each entry played each card ordering, and the average score and time for that entry was calculated. The four highest-scoring entries were then provided with a hundred new random orderings, to calculate final score rankings.

The score rankings between entries varied between card orderings, and it would be interesting to investigate the sources of the variation. The judges would be delighted with a graphical harness allowing the different playing styles of the entries to be compared, or for entries to play against each other.

The test platform was a 2.4 GHz P4 with 512MB memory, running LispWorks 5.0.1 on Windows XP Service Pack 2.

	average time(s)	10-game average score	100-game average score	incorrect scoring
Buss	1.28	818.2	804.52	
LarryL	10.30	122.1	-	
Ohler	116.00	780.7	-	x
Rhodes	16.03	810.5	819.79	x
Salvi	0.69	829.0	827.49	
Sedach	0.15	835.4	829.87	
Senator	3.80	685.9	-	x
Witmond	0.20	720.2	-	x