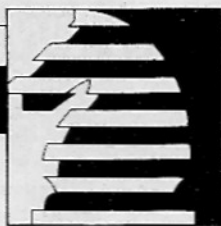


MICRO



CHESS

WINNER TAKES ALL

Richard Lang's chess program has gone from strength to strength since the 1981 PCW Chess Tournament. Tony Harrington reports.

Richard Lang's name has already been mentioned from time to time in these columns. As the designer of the program which won the 1981 PCW Tournament with a crushing score of five out of five, he could hardly have failed to get some attention.

He is one of the new breed of young enthusiasts, who has drifted into the intricacies of computer chess more or less by chance, and who has found in it something seductive and obsessive. Lang had nothing to do with computers in his school days. His school, in the village of Winnersh, near Reading, encouraged studies of the more traditional kind. And the chess that was played there was between schoolboys rather than machines.

'We played a few lightning games and the odd chess tournament,' Lang said. 'My father taught me the moves of chess when I was around eight years old, and I played a bit from time to time. But my interest in computers came when I bought a Video Genie home computer three years ago. I taught myself Basic, but that was irritatingly slow, so after a month I bought a book on machine code programming.'

'The first project I tried was to write a video game. That wasn't a great success. I wanted to do something other than simply produce another version of space invaders, so I decided to have a go at writing a computer chess program.'

Writing the video game in assembler code occupied about the first three months of Lang's time after the arrival of the Genie. Thereafter, he put childish things behind him, and settled down to solve a few of the problems that perplex the AI staff of Britain's university computer departments. (AI, for those new to these pastures, stands for Artificial Intelligence.)

No one just launches into the writing of computer chess programs on instinct and native wit alone. There are certain fundamental things to be learned, as every aspirant computer chess programmer will discover. Lang quickly found that he needed some assistance in getting his program off the ground. Accordingly he went out and scoured the bookshops for likely material.

This was provided in the shape of a book by Dan and Kathy Spracklen (who, as readers of this column might remember, achieved fame as the designers of the Sargon program, and who are still writing

chess programs for Fidelity). The book was *Sargon, the Chess Program*, and it provided, among other things, a full listing of one of the Sargon programs.

'That helped me considerably. I read it and absorbed a lot of what they were doing. It gave me an idea of how chess programs were written,' Lang commented. In case you think that this is all there is to writing tournament winning chess programs, I should point out that within a very short time of reading about that version of the Spracklens' program, Lang fancied that he saw several ways of improving on what they had done.

'I could see far better ways of doing a lot of the things. Much depended on finding ways of improving the speed of their program. I found better algorithms and better ways of obtaining a score for chess positions,' he said modestly.

I asked how it was that he, on his (at that stage) scant acquaintance with chess programs, was able to see further into the intricacies of composing chess algorithms than the Spracklens had produced in that program. (Not that the Spracklens haven't devised other, better algorithms since, because they undoubtedly have.) Lang was puzzled about how to answer my question.

'I simply did,' he said, and seemed to think it perfectly natural. Pressed a little further on the subject, he added: 'Understanding chess algorithms is something you either can do or you can't. It's a bit like a musical gift. I suppose practice helps, but certain people seem to be better at it than others. I found that I was one of the people who was good at it.'

Despite the high graphics content in the video game that was Lang's first venture into programming, his first chess program eschewed graphics altogether. ('To have written graphics for it then would simply have diverted me from the chess program itself.') All it did was to provide an algebraic notation of the moves. And unless the human opponent was an expert at blindfold chess, an ordinary chess board was an essential 'peripheral'.

He started writing the program in January 1981. This was a leisure hours hobby, since his ostensible occupation at that time was as an employee in the research department of British Gas, doing risk analysis. 'I used computer programs in the course of my work, but I didn't have a thing to do with writing them,' he

explained. 'That all happened after the office closed for the day.'

The PCW Tournament for that year was, as always, scheduled for September. Lang knew of the Tournament, and he says that it was a prime incentive in sharpening up the way his program played. 'To start with, I would have written the program anyway, regardless of whether there was a tournament on or not. But once I got started and realised that it was playing reasonably well, I kept the Tournament firmly in mind.'

'It was something of a race to get it into shape in time and I could have done with a few extra months. By the time the Tournament arrived, it was running on a Nascom micro with a 4 MHz clock. It did rather well and won in every round. David Levy and Kevin O'Connell took an interest in it at the Tournament and at the end of the Tournament they offered me two contracts. One for the program, and one to go and work as a programmer for them. This was very pleasing for me,' he said.

Since then, in addition to working on chess programs, Lang has written what might be the world's strongest program for the Oriental game Gomoko Renju — a game which I do not understand, but which I believe is played on a board similar to a Go-board. It involves trying to place stones in a straight or a diagonal line while preventing your opponent from achieving this end.

Lang didn't know how to play Gomoko Renju at the time and still doesn't play it. That wasn't necessary. 'It is sufficient to tell a programmer what the rules of any game are, though it probably helps a bit if you play the game,' he reflected. For the Gomoko Renju game, Lang was able to draw on the advice of a master, and that was sufficient.

Having sold the Cyrus 1 program, Lang set about writing Cyrus 2. He entered this in last year's PCW Show, but it didn't quite come up to his expectations. 'It was written in quite a hurry and the Tournament came in the middle of its development period rather than at the end,' Lang said. There were quite a few new ideas in the program and he didn't have much time to test them before the Tournament.

The new ideas were a combination of getting sections of the program to run faster, and building more chess 'wisdom' into it, by getting it to recognise isolated pawns, doubled pawns and the like.



Richard Lang: one of the new breed

Although it didn't do too well in the Tournament, Lang reckons that the current version of his program is demonstrably better than Fidelity's Sensory Nine.

A version of Cyrus has been bought by Dragon and now runs as a cartridge on the Dragon home computer. That, however, is another tale. Suffice it to say that on several occasions I have played against the Dragon version and have found it a worthy opponent.

Lang intends entering a new version of Cyrus in the 1983 PCW Chess Tournament, and that should answer any remaining questions.

Games section

White: Cyrus; Black: Cray Blitz; Pirc Defence: Notes by David Levy.

This is undoubtedly the most impressive victory scored by Cyrus to date. It was played in the speed tournament at the 1981 North American Computer Championships, and Cyrus' opponent was running on the Cray 1 computer, the world's fastest mainframe. In the previous year's competition, Cray Blitz had won the speed tournament with a score of 15 out of 15.

- | | |
|-----------|--------|
| 1 e2-e4 | d7-d6 |
| 2 d2-d4 | g7-g6 |
| 3 Nb1-c3 | Ng8-f6 |
| 4 Bf1-b5+ | Nb8-d7 |
| 5 Ng1-f3 | c7-c6 |
| 6 Bb5-e2 | Bf8-g7 |
| 7 0-0 | 0-0 |
| 8 Qd1-d3 | e7-e5 |
| 9 Bc1-g5 | h7-h6 |
| 10 Bg5xf6 | Qd8xf6 |
| 11 Rf1-d1 | Rf8-d8 |

(The position is roughly equal, with each player having some chances of creating play. White exerts some pressure along the d-file, while Black has play on the h8-a1 diagonal.)

- | | |
|--|-------|
| 12 a2-a4 | a7-a5 |
| (Otherwise White cramps Black by a4-a5.) | |

- | | |
|-----------|--------|
| 13 Ra1-a3 | Qf6-f4 |
| 14 g2-g3 | Qf4-f6 |
| 15 Ra3-b3 | Ra8-a7 |
| 16 Qd3-c4 | Ra7-a6 |
| 17 d4xe5 | d6xe5 |
| 18 Rd1-d2 | b7-b6 |

(Intending 19... Ra6-a7 and 20... Bc8-a6, trading off the light squared bishops and leaving White with weaknesses on the light squares near his king.)

- | | |
|------------|--|
| 19 Qc4-d3! | |
|------------|--|

(Now White will establish a strongpost on d6.)

- | | |
|-----------|--------|
| 19 ... | Bc8-b7 |
| 20 Qd3-d6 | Qf6xd6 |
| 21 Rd2xd6 | Ra6-a8 |
| 22 Rd6-d2 | Bb7-a6 |
| 23 Be2xa6 | Ra8xa6 |
| 24 Nc3-e2 | c6-c5 |
| 25 Rb3-d3 | |

(Completing White's domination of the d-file.)

- | | |
|------------|--------|
| 25 ... | Ra6-a7 |
| 26 Ne2-c3! | |

(The start of a knight manoeuvre which capitalises on Black's weak Q-side squares.)

- | | |
|------------|--------|
| 26 ... | Ra7-b7 |
| 27 b2-b3 | Kg8-h8 |
| 28 Nc3-b5 | Kh8-g8 |
| 29 Nb5-d6 | Rb7-c7 |
| 30 Nd6-b5 | Rc7-b7 |
| 31 Rd3-d5! | |

(Threatening 32 Nf3xe5 Bg7xe5 33 Rd5xe5 Nd7xe5 34 Rd2xd8+, and thereby forcing Black to weaken its position further.)

- | | |
|------------|--------|
| 31 ... | f7-f6 |
| 32 Nb5-d6 | Rb7-c7 |
| 33 Nd6-b5 | Rc7-b7 |
| 34 c2-c4 | Bg7-f8 |
| 35 Nb5-c3 | g6-g5 |
| 36 Nf3-e1! | |

(Taking advantage of Black's last move by starting the knight on its way to the strong square e3 followed by g4 or f5.)

- | | |
|------------|--------|
| 36 ... | Bf8-g7 |
| 37 Ne1-c2 | Kg8-h8 |
| 38 Nc2-e3 | Kh8-g8 |
| 39 Ne3-f5 | Bg7-f8 |
| 40 Nf5-e3 | Bf8-g7 |
| 41 Rd5-d3 | Kg8-h8 |
| 42 Ne3-f5 | Rb7-a7 |
| 43 Nc3-b5! | |

(The roof caves in on Black's brittle structure. If now 43... Ra7-b7 44 Nb5-d6 Bf8xd6 45 Nf5xd6 Rb7 moves 46 Nd6-f7+, forking Black's king and the rook on d8.)

- | | |
|-----------------------------------|--------|
| 43 ... | Ra7-a6 |
| 44 Nb5-c7 | Ra6-a7 |
| 45 Nc7-e6 | Rd8-e8 |
| If 45... Rd8-g8 46 Ne6xg7 Rg8xg7 | |
| 47 Nf5xg7 Kh8xg7 48 Rd3xd7+ etc.) | |

46 Ne6xg7 Black Resigns

(All this at five seconds per move CPU time!)

END