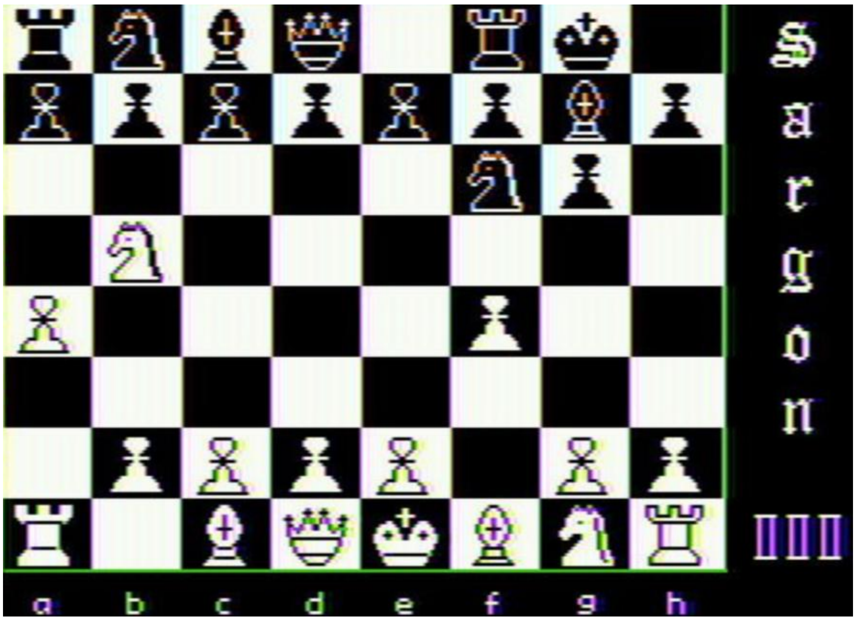
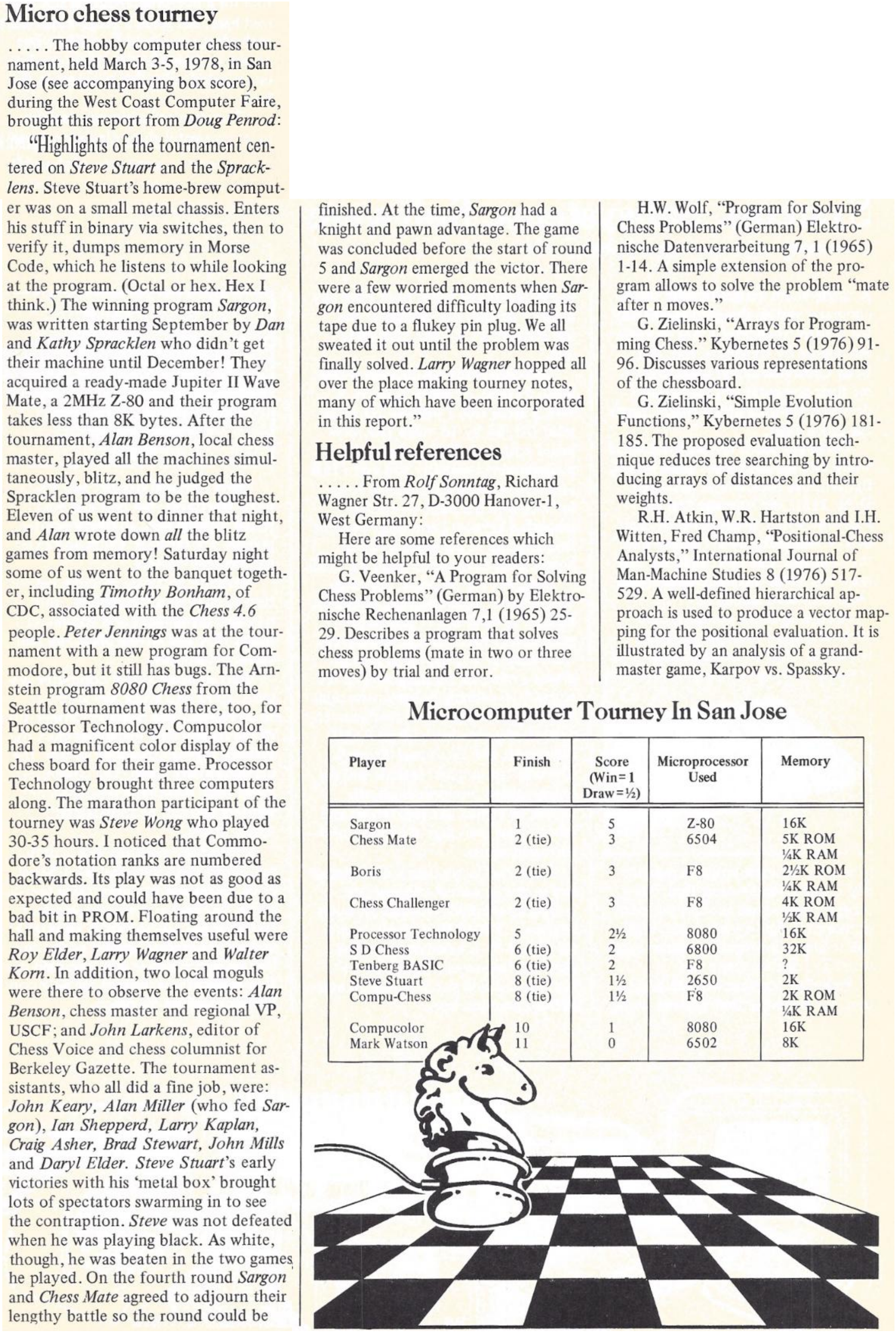
Exactly one year after the introduction of the Fidelity Chess Challenger (1), the world's first Microcomputer Chess Tournament was held. This tournament took place from March 3-5, 1978, during the Second West Coast Computer Faire in San Jose, California. Participants were limited to computers based on 8-bit microprocessor chips, with less than 32K bytes of memory. It was the first computer chess competition without terminals connected by telephone to multi-million dollar machines far from the playing site. Participants ranged from an $85 "homemade metal box" to a device costing around $6,000. Three of the computers, including Chess Challenger and Boris, were commercial consumer products. The winning program, called Sargon, scored 5-0, two points ahead of the contestant field. Sargon was programmed by married couple Dan and Kathe Spracklen for a 16 Kbyte Z-80 computer. Boris and Chess Challenger competed for second place with Commodore Chessmate, the prototype of a consumer product expected on the market soon. They each finished with three points. In this article I want to give you the interesting background stories as recorded in 1978 and 1979.





Micro chess tourney

. . . . . The hobby computer chess tournament, held March 3-5, 1978, in San Jose (see accompanying box score), during the West Coast Computer Faire, brought this report from *Doug Penrod*:

"Highlights of the tournament centered on *Steve Stuart* and the *Spracklens*. Steve Stuart's home-brew computer was on a small metal chassis. Enters his stuff in binary via switches, then to verify it, dumps memory in Morse Code, which he listens to while looking at the program. (Octal or hex. Hex I think.) The winning program *Sargon*, was written starting September by *Dan* and *Kathy Spracklen* who didn't get their machine until December! They acquired a ready-made Jupiter II Wave Mate, a 2MHz Z-80 and their program takes less than 8K bytes. After the tournament, *Alan Benson*, local chess master, played all the machines simultaneously, blitz, and he judged the Spracklen program to be the toughest. Eleven of us went to dinner that night, and *Alan* wrote down *all* the blitz games from memory! Saturday night some of us went to the banquet together, including *Timothy Bonham*, of CDC, associated with the *Chess 4.6* people. *Peter Jennings* was at the tournament with a new program for Commodore, but it still has bugs. The Arnstein program *8080 Chess* from the Seattle tournament was there, too, for Processor Technology. Compucolor had a magnificent color display of the chess board for their game. Processor Technology brought three computers along. The marathon participant of the tourney was *Steve Wong* who played 30-35 hours. I noticed that Commodore's notation ranks are numbered backwards. Its play was not as good as expected and could have been due to a bad bit in PROM. Floating around the hall and making themselves useful were *Roy Elder*, *Larry Wagner* and *Walter Korn*. In addition, two local moguls were there to observe the events: *Alan Benson*, chess master and regional VP, USCF; and *John Larkens*, editor of Chess Voice and chess columnist for Berkeley Gazette. The tournament assistants, who all did a fine job, were: *John Keary, Alan Miller* (who fed *Sargon*), *Ian Shepperd, Larry Kaplan, Craig Asher, Brad Stewart, John Mills* and *Daryl Elder. Steve Stuart*'s early victories with his 'metal box' brought lots of spectators swarming in to see the contraption. *Steve* was not defeated when he was playing black. As white, though, he was beaten in the two games he played. On the fourth round *Sargon* and *Chess Mate* agreed to adjourn their lengthy battle so the round could be finished. At the time, *Sargon* had a knight and pawn advantage. The game was concluded before the start of round 5 and *Sargon* emerged the victor. There were a few worried moments when *Sargon* encountered difficulty loading its tape due to a flukey pin plug. We all sweated it out until the problem was finally solved. *Larry Wagner* hopped all over the place making tourney notes, many of which have been incorporated in this report."

Helpful references

. . . . . From *Rolf Sonntag*, Richard Wagner Str. 27, D-3000 Hanover-1, West Germany:

Here are some references which might be helpful to your readers:

G. Veenker, "A Program for Solving Chess Problems" (German) by Elektronische Rechenanlagen 7,1 (1965) 25-29. Describes a program that solves chess problems (mate in two or three moves) by trial and error.

H.W. Wolf, "Program for Solving Chess Problems" (German) Elektronische Datenverarbeitung 7, 1 (1965) 1-14. A simple extension of the program allows to solve the problem "mate after n moves."

G. Zielinski, "Arrays for Programming Chess." Kybernetes 5 (1976) 91-96. Discusses various representations of the chessboard.

G. Zielinski, "Simple Evolution Functions," Kybernetes 5 (1976) 181-185. The proposed evaluation technique reduces tree searching by introducing arrays of distances and their weights.

R.H. Atkin, W.R. Hartston and I.H. Witten, Fred Champ, "Positional-Chess Analysts," International Journal of Man-Machine Studies 8 (1976) 517-529. A well-defined hierarchical approach is used to produce a vector mapping for the positional evaluation. It is illustrated by an analysis of a grandmaster game, Karpov vs. Spassky.

Microcomputer Tourney In San Jose

Player	Finish	Score (Win=1 Draw=½)	Microprocessor Used	Memory
Sargon	1	5	Z-80	16K
Chess Mate	2 (tie)	3	6504	5K ROM ¼K RAM
Boris	2 (tie)	3	F8	2½K ROM ¼K RAM
Chess Challenger	2 (tie)	3	F8	4K ROM ½K RAM
Processor Technology	5	2½	8080	16K
S D Chess	6 (tie)	2	6800	32K
Tenberg BASIC	6 (tie)	2	F8	?
Steve Stuart	8 (tie)	1½	2650	2K
Compu-Chess	8 (tie)	1½	F8	2K ROM ¼K RAM
Compucolor	10	1	8080	16K
Mark Watson	11	0	6502	8K

Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Personal Computing – May 1978

Software Dynamics Chess

. . . . . The following pleasant monologue was received from *Ira Baxter,* of Software Dynamics, author of *SD Chess:* "One might ask, 'Why would someone bother to write a chess program in something as arcane as BASIC?' The answer – how could I resist? Nobody believes it to be possible! *SD Chess* is a program written in SD BASIC, a compiler version of BASIC. The program can be instructed to play at different levels of skill, but has only two practical modes of play – blitz and dumb, due to time constraints.

"*SD Chess* was entered in the West Coast Computing Faire's Chess Tournament. It beat Mark Watson and Tenberg BASIC but lost to *Chess Challenger* and *CompuChess.*

"This note describes a little about the operation of *SD Chess,* and is mostly oriented towards the programming tricks and heuristics installed in the program. In the following discussion, I assume the readers are familiar with the fundamental operation of lookahead logic on game trees that most chess-playing programs use. There is an excellent book on the subject, 'Chess and Computers' by David Levy, that is definitely worth reading.

"*SD Chess* operates by generating all the possible moves for White for a particular (parent) board position, making each move on the parent board (creating a daughter board) in turn, changing all White pieces to Black, and recursively applying the move generation logic again. Each of these 'make a move and switch piece colors' I call a single-'ply' lookahead. The program can look ahead up to 5 ply (the actual depth of lookahead is established by a conversation with the human player at the beginning of the game).

"The program stops looking ahead when it reached the maximum lookahead level, and then applies a board evaluation function to determine the score for ('how good') the resulting position. The actual evaluation function at this point in time is simply the sum of the values of White pieces minus the sum of the values of the Black pieces, with a free point thrown to the appropriate side if that side has castled.

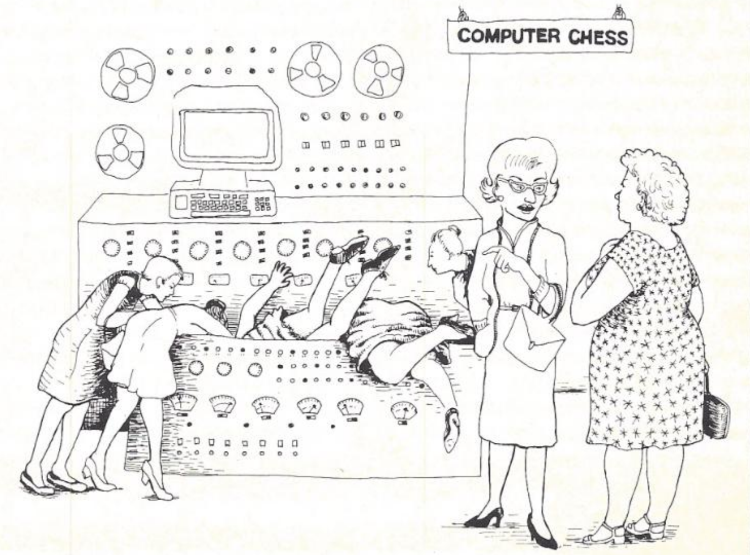
"The scores of the board positions resulting from moving pieces from a parent board B are then compared to obtain the maximum score (the heuristically 'best' move). The maximum score is then taken to be the score of board position 8.

"When passing scores up to a parent board, the negative of the score for the daughter board is used. This is because the daughter board actually was scored with White pieces swapped for Black (I also swap back).

"By passing these negated scores up the game tree and always applying a MAX function, I effectively cause a mini-max evaluation of the game tree. This process also eliminates the need for any move generating logic for the Black pieces.

"The lookahead is limited by a minimum and maximum value. The minimum lookahead guarantees that *SD Chess* will examine all possible legal moves to a certain depth. *SD Chess* will then continue looking deeper if the board position obtained at the minimum lookahead was arrived at by a capture move. This heuristic is used to make *SD Chess* examine capture sequences out to the bitter (or Max depth) end, so that it can see that taking your protected pawn with its King is a good way to lose, even if the pawn capture occurs at the minimum lookahead depth.

"In an attempt to minimize the number of moves processed by *SD Chess,* the move generating logic retains only the highest-scoring capture moves when it is operating at the maximum depth; generally, a high-scoring capture



"They say there are midgets in there doing all the work!"

Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Personal Computing – June 1978

indicates a good refutation of a previous move, and by trimming the move list at the deepest level, we save the program an enormous amount of processing time, since the number of moves to process at depth N is roughly 30†N (2700 for N=3).

"An Alpha-Beta pruning algorithm is used to prevent further search of sub-trees which are obviously fruitless. Since I have to evaluate only White board positions, the -Beta part of the pruning algorithm is not needed. Furthermore, the move generating logic sorts generated moves by descending value of score, to maximize the probability of the Alpha-prune occurring.

"The first version of the program had the sorting test backwards, which minimized the probability of pruning. This 'bug' was discovered by sheer introspection because all it did was slow down the program by a factor of 3, and I didn't know how fast the program would run!

"The most annoying effect I have yet encountered is what is called the 'horizon' effect, where the program fails to see that it is going to be in hot water. A typical problem this causes is the following: The program discovers the opponent can castle, no matter what the program does. So the opponent will gain 1 point. Now the program chooses a move which gives a pawn to the opponent for free (after all, if you're going to lose a point, it hardly matters which one you lose, does it?), because the lookahead maximum prevents the program from discovering that after it has lost the pawn the opponent can *still* castle (i.e., the program has parlayed a one point loss into a two point loss). Yuk! I don't know a general cure for this problem other than extending the horizon. I did cure this particular aspect by scoring castling slightly less than a pawn.

At the top level of move evaluation (i.e., ply 0), *SD Chess* selects moves which move toward the enemy king if all other things are equal. This provides *SD Chess* with the long-range goal of 'get near the opponent's king'. It also provides P - K4 as a standard opening for free.

"*SD Chess* also attempts to minimize the opponent's mobility (all other things being equal). This is supposed to help trap the enemy king in the late endgame, but I've never seen any evidence of it helping. The only other interesting characteristic of the implementation of *SD Chess* is a programming trick. *SD Chess* stores the chess board (8 by 8) in the middle of a 12 by 12 field, with the borders filled with White pawns:

```
P P P P P P P P P P P P
P P P P P P P P P P P P
P P                 P P
P P                 P P
P P                 P P
P P                 P P
P P                 P P
P P                 P P
P P                 P P
P P                 P P
P P P P P P P P P P P P
P P P P P P P P P P P P
```

"This trick considerably simplifies the move logic since any generated move of a piece lands somewhere inside the 12 by 12 board. Since the move generator must already check to make sure that a piece does not move onto a square occupied by a piece of the same color, it will reject any attempt to move off the edge of the 8 by 8 playing area because the moved piece would then land on a White pawn. This means the 'did the piece move off the edge of the board?' check is done for free, and results in a considerable savings. The program has no opening move sequences stored, so it does absolutely no 'back' play.

"Last but not least is, how well does the program play? At blitz level, it plays at a level comparable to the Randy Miller chess program (written in Altair [TM] BASIC) using about 1 second of CPU per move while the Miller program takes some 3-4 minutes to do a poorer job (Miller's program goaded me into writing *SD Chess* because it was impossible to improve his).

"At the level which I play it (1-5 minutes a move, because I haven't the patience to wait longer), it is short enough to discover sequences in which it can safely make off with your piece, to discover it has been checkmated, and to prevent a threatened mate with a simple counter-move. During one freak game I played with it, it used a tolerably good version of the Ruy Lopez opening. It seems to be clever enough so you can't beat it with one arm tied behind your back, and sometimes wins because its exhaustive search never overlooks a combination you didn't notice. This is the level I expect it to play in tournaments.

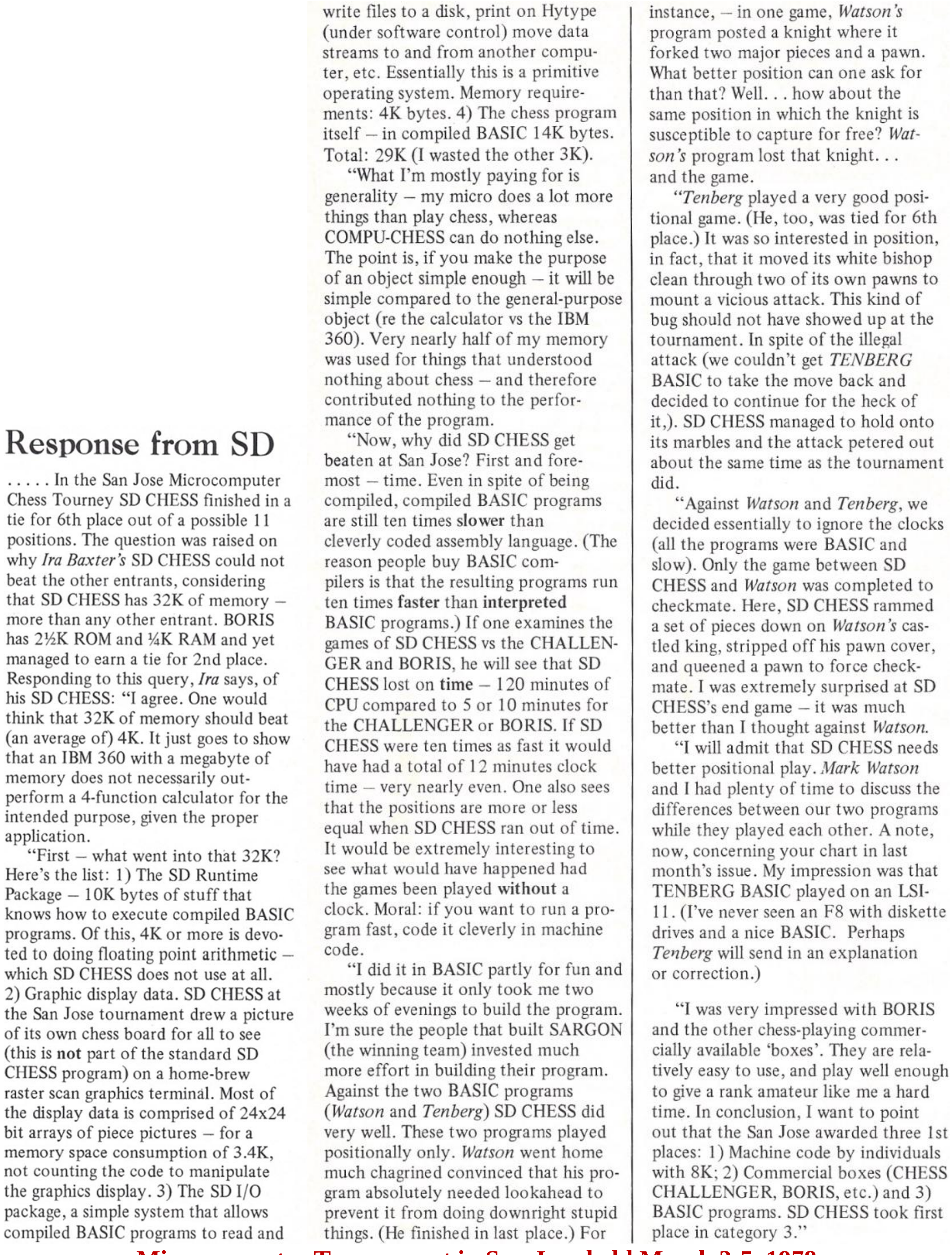
"The biggest lesson learned from this program is that one apparently needs enormous amounts of processing power to play chess well this way. No wonder *Chess 4.6* does well, it has a CDC 176 behind it! I fully expect *SD Chess* to be beaten by assembly language programs simply because of the extra analysis an assembly program can invest in the same amount of time *SD Chess* has spent analyzing. (I note here that *SD Chess* is compiled to 'pop code', which runs 10-60 times faster than conventional BASIC interpreters and some 10 times slower than assembly language programs doing all 16-bit arithmetic.) My next optimization is to really compile the beast to machine code. But a compiler that can do it isn't yet available.

"One more comment. It seems that since microprocessors lack computing power so miserably, perhaps we should run microprocessor Chess tournaments by mail, with an allowed overnight turnaround to make a single move. I'd bet this would considerably improve the games these beasties play.

"*SD Chess* can be obtained from Software Dynamics, 17914 S. Laurelbrook Pl., Cerritos, CA 90701. It requires the SD RUNTIME Package, and runs on 6800 CPUs with at least 32K bytes of memory. SDRUN + SDCHESS together cost some $125. Most people buy the SD BASIC Compiler system and get *SD Chess* as a demo program."

Why would someone bother to write a chess program in something as arcane as BASIC?

Microcomputer Tournament in San Jose held March 3-5, 1978
Software Dynamics Chess (SD Chess)
Source: Personal Computing – June 1978



Response from SD

. . . . . In the San Jose Microcomputer Chess Tourney SD CHESS finished in a tie for 6th place out of a possible 11 positions. The question was raised on why *Ira Baxter's* SD CHESS could not beat the other entrants, considering that SD CHESS has 32K of memory – more than any other entrant. BORIS has 2½K ROM and ¼K RAM and yet managed to earn a tie for 2nd place. Responding to this query, *Ira* says, of his SD CHESS: "I agree. One would think that 32K of memory should beat (an average of) 4K. It just goes to show that an IBM 360 with a megabyte of memory does not necessarily outperform a 4-function calculator for the intended purpose, given the proper application.

"First – what went into that 32K? Here's the list: 1) The SD Runtime Package – 10K bytes of stuff that knows how to execute compiled BASIC programs. Of this, 4K or more is devoted to doing floating point arithmetic – which SD CHESS does not use at all. 2) Graphic display data. SD CHESS at the San Jose tournament drew a picture of its own chess board for all to see (this is **not** part of the standard SD CHESS program) on a home-brew raster scan graphics terminal. Most of the display data is comprised of 24x24 bit arrays of piece pictures – for a memory space consumption of 3.4K, not counting the code to manipulate the graphics display. 3) The SD I/O package, a simple system that allows compiled BASIC programs to read and write files to a disk, print on Hytype (under software control) move data streams to and from another computer, etc. Essentially this is a primitive operating system. Memory requirements: 4K bytes. 4) The chess program itself – in compiled BASIC 14K bytes. Total: 29K (I wasted the other 3K).

"What I'm mostly paying for is generality – my micro does a lot more things than play chess, whereas COMPU-CHESS can do nothing else. The point is, if you make the purpose of an object simple enough – it will be simple compared to the general-purpose object (re the calculator vs the IBM 360). Very nearly half of my memory was used for things that understood nothing about chess – and therefore contributed nothing to the performance of the program.

"Now, why did SD CHESS get beaten at San Jose? First and foremost – time. Even in spite of being compiled, compiled BASIC programs are still ten times **slower** than cleverly coded assembly language. (The reason people buy BASIC compilers is that the resulting programs run ten times **faster** than **interpreted** BASIC programs.) If one examines the games of SD CHESS vs the CHALLENGER and BORIS, he will see that SD CHESS lost on **time** – 120 minutes of CPU compared to 5 or 10 minutes for the CHALLENGER or BORIS. If SD CHESS were ten times as fast it would have had a total of 12 minutes clock time – very nearly even. One also sees that the positions are more or less equal when SD CHESS ran out of time. It would be extremely interesting to see what would have happened had the games been played **without** a clock. Moral: if you want to run a program fast, code it cleverly in machine code.

"I did it in BASIC partly for fun and mostly because it only took me two weeks of evenings to build the program. I'm sure the people that built SARGON (the winning team) invested much more effort in building their program. Against the two BASIC programs (*Watson* and *Tenberg*) SD CHESS did very well. These two programs played positionally only. *Watson* went home much chagrined convinced that his program absolutely needed lookahead to prevent it from doing downright stupid things. (He finished in last place.) For instance, – in one game, *Watson's* program posted a knight where it forked two major pieces and a pawn. What better position can one ask for than that? Well. . . how about the same position in which the knight is susceptible to capture for free? *Watson's* program lost that knight. . . and the game.

"*Tenberg* played a very good positional game. (He, too, was tied for 6th place.) It was so interested in position, in fact, that it moved its white bishop clean through two of its own pawns to mount a vicious attack. This kind of bug should not have showed up at the tournament. In spite of the illegal attack (we couldn't get *TENBERG* BASIC to take the move back and decided to continue for the heck of it,). SD CHESS managed to hold onto its marbles and the attack petered out about the same time as the tournament did.

"Against *Watson* and *Tenberg*, we decided essentially to ignore the clocks (all the programs were BASIC and slow). Only the game between SD CHESS and *Watson* was completed to checkmate. Here, SD CHESS rammed a set of pieces down on *Watson's* castled king, stripped off his pawn cover, and queened a pawn to force checkmate. I was extremely surprised at SD CHESS's end game – it was much better than I thought against *Watson.*

"I will admit that SD CHESS needs better positional play. *Mark Watson* and I had plenty of time to discuss the differences between our two programs while they played each other. A note, now, concerning your chart in last month's issue. My impression was that TENBERG BASIC played on an LSI-11. (I've never seen an F8 with diskette drives and a nice BASIC. Perhaps *Tenberg* will send in an explanation or correction.)

"I was very impressed with BORIS and the other chess-playing commercially available 'boxes'. They are relatively easy to use, and play well enough to give a rank amateur like me a hard time. In conclusion, I want to point out that the San Jose awarded three 1st places: 1) Machine code by individuals with 8K; 2) Commercial boxes (CHESS CHALLENGER, BORIS, etc.) and 3) BASIC programs. SD CHESS took first place in category 3."

Microcomputer Tournament in San Jose held March 3-5, 1978
Response from Software Dynamics Chess (SD Chess)

Source: Personal Computing – July 1978

The Second West Coast Computer Faire

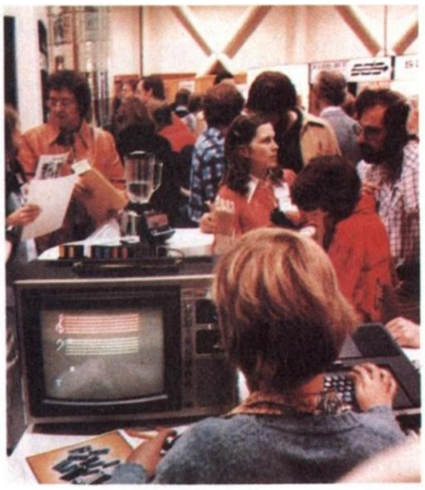


Photo 1: Some of the 14,000+ crowd amble by a young hacker programming music on a Video Brain computer.

Photo 2: Robot trials at the Dynabyte booth, a popular attraction at the Second West Coast Computer Faire.

By Chris Morgan, Editor

San Jose was the place to be last March 3, 4 and 5 for the Second West Coast Computer Faire. The Convention Center was easily able to handle the crowd of 14,169 who came to see the latest developments in personal computing.

A quick examination of some of the hundreds of manufacturers' booths revealed some trends: floppy disks are on the increase, with new models being shown or promised by Heathkit, Apple, Radio Shack and many others; more and more personal computers are now being offered with built-in floppy disks; peripherals and add-ons are

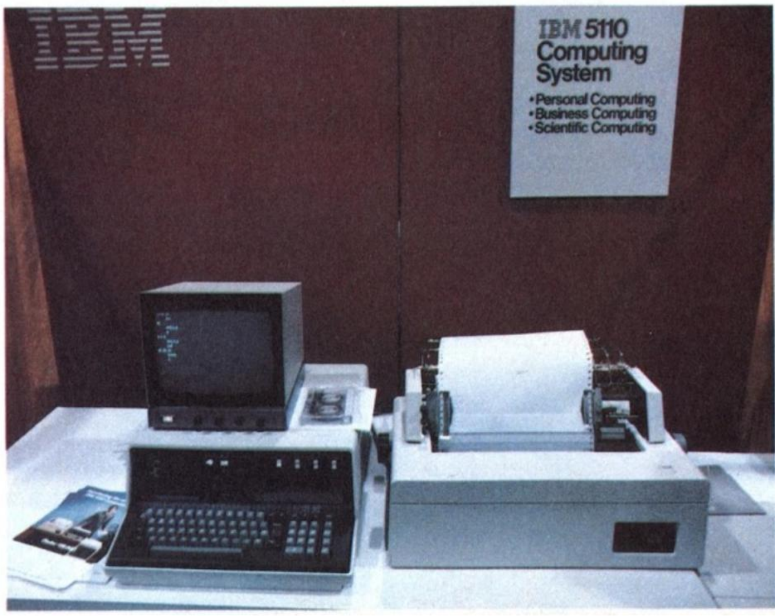


Photo 3: IBM's booth, an auspicious addition to the show.

The Second West Coast Computer Faire &
Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Byte magazine – July 1978

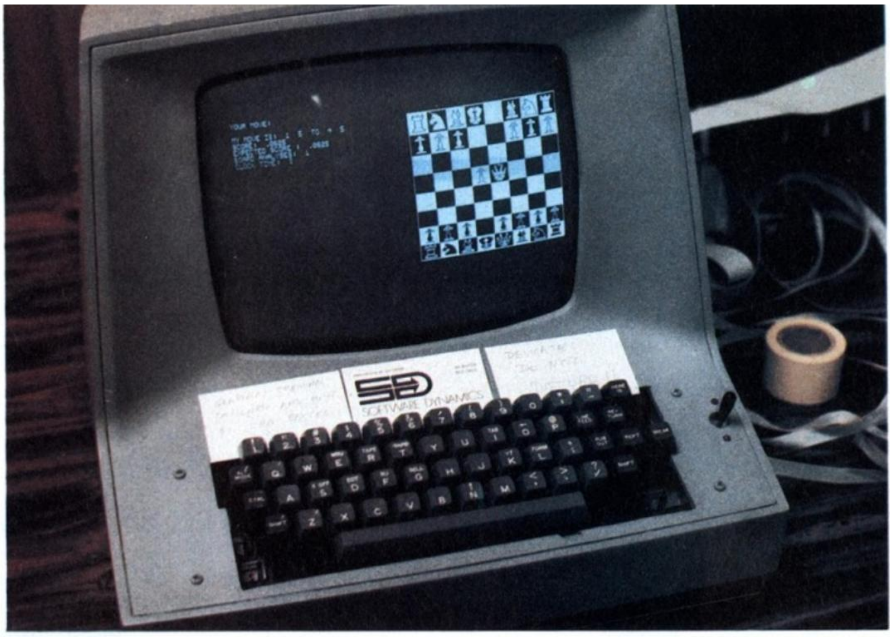


Photo 4: Ira Baxter's chess playing system display, which competed in the Microcomputer Chess Tournament at the Faire.

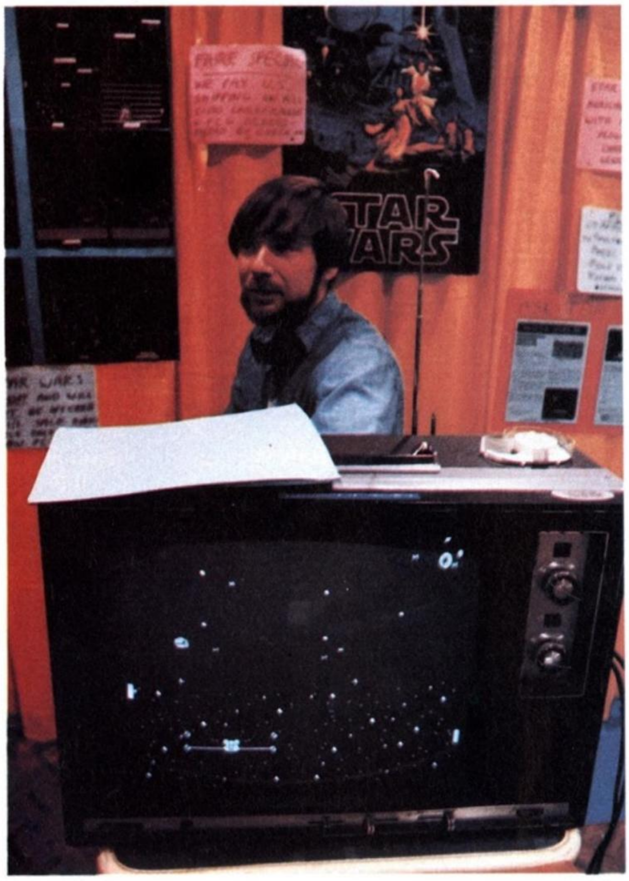
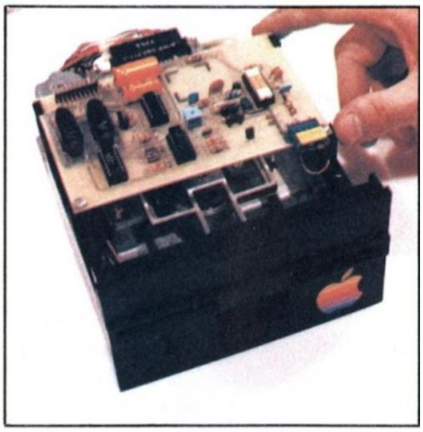


Photo 6: Objective Design's Larry Weinstein displays Star Wars graphics.

Photo 5: Apple Computer's new minifloppy drive.



now available for a wide variety of computer buses.

I enjoyed the many special features of the show, particularly the excellent computer generated art on display in the lobby. The microcomputer chess tournament proved to be one of the hits of the show. Larry Wagner from Atari presided over the 3 day battle of the processors, taking time out to give me a guided tour of the tournament. The level of play was impressive, and the winning program, called SARGON, was a 16 K byte Z-80 assembler program written by a husband and wife team, Kathe and Dan Spracklen. It beat some highly touted com-

The Second West Coast Computer Faire & Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Byte magazine – July 1978

Photo 7: Heath's new H27 dual floppy drive, scheduled to be available later this year.

petition. (A copy of the SARGON program is available for $15 postpaid from the Spracklens, 10832 Macouba Pl, San Diego CA 92124.)

I was impressed with the professional appearance of the show, which held its own with many of the established engineering and computing shows. The Third West Coast Computer Faire will be held this coming November 3, 4 and 5 in Los Angeles. Plan to see it if you can.■

Photo 8: Students from Mills College Center for Contemporary Music in Oakland demonstrate a digital and analog hybrid music synthesizer system, one of many special exhibits at the Faire.

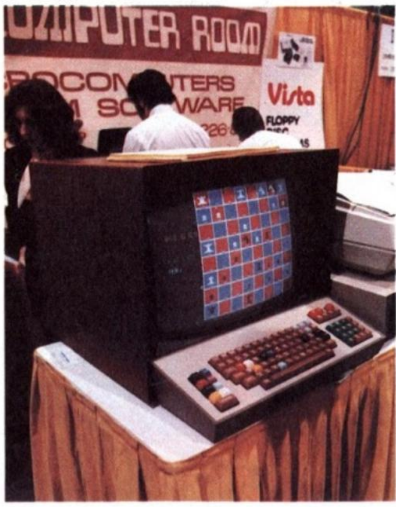


Photo 9: Cromemco color video unit displays chess program at the Computer Room of San Jose booth.

The Second West Coast Computer Faire &
Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Byte magazine – July 1978

San Jose Micro Tournament

. . . . Larry Wagner, tournament director of the first microcomputer chess tournament, held at San Jose, CA in March 1978, sends along the following report: "For the 2½ days of the Faire a little less than a dozen microprocessors (and their memories, I/O, power supplies, etc.) along with their programmers competed in a computer versus computer chess tournament. This was the first computer chess tournament *without* terminals and telephone hook-ups to remote machines whose values could be measured in millions of dollars. The most expensive computer entered was around $6000 while the lowest cost entry was a ,homebrew' metal box containing $85 worth of parts. When it was all over *Sargon,* a program for a Z-80 developed by a husband and wife programming team, finished in first place with a convincing 5 wins out of 5 games played. After the field of entries was established, the class structure was defined to be:

CLASS A	MICROCOMPUTERS WITH 8K OR GREATER MEMORY
CLASS B	MICROCOMPUTERS WITH LESS THAN 8K MEMORY
CLASS C	PROGRAMS RUNNING IN BASIC

"The prize certificates were awarded by the class. However, all the entries were played against each other. As the tournament progressed, we had to make adjustments in the rules and scheduling. During the first day, two of the programs running in BASIC could not keep up with the time control of 50 moves in 2 hours. It was decided that they would play each other in a single 9 hour match on the second day while the other competitors were playing in two 4 hours matches. Several matches had to be restarted due to loss of power when someone pulled out a plug by mistake, or the realization that there was a notation misunderstanding, or for undetected move entry errors.

"We noticed that the machines had a tendancy to make repeated moves or perpetual checks even when they were significantly ahead in the material. In order to prevent these games from resulting in a draw, we allowed the programmers to adjust the machines to increase or decrease the look-ahead level or response time to try and get out of these 'lockup' situations. At the conclusion of the computer versus computer tournament, Alan Benson, Senior Regional Vice President of the United States Chess Federation and ICCF Postal Master, played a simultaneous exhibition against all the computers (and a few humans too). Following is one of the game scores (*Sargon* vs. Steve Stuart) with Alan's chess commentary.'

White: Sargon **Black: Steve Stuart**

1. P-Q4	P-Q4
2. N-QB3	N-QB3
3. B-B4	B-B4
4. N-B3	N-B4
5. P-QR4(a)	P-K5(b)
6. PxP	N-KN4
7. P-R3	KNxPK5
8. BxN	NxB
9. NxN	P-Q5
10. P-KN4?	B-K3?(c)
11. N-N5	B-N5ch
12. P-QB3	B-QB4
13. PxP	B-N5ch
14. N-B3	Q-O
15. B-N2	P-QB3
16. Q-Q3	BxNch
17. QxB	Q-Q3(d)
18. O-O	P-B3
19. P-K3	Q-QB2
21. KR-Q1	QR-Q1
22. P-B4	K-R1
23. K-B2(e)	P-QR1
24. P-B5	B-N1
25. P-K4	Q-B5ch
26. K-K2(f)	KR-K1
27. N-R4	R-K2
28. Q-B5	R-Q3(g)
29. N-B4	QR-Q2
30. N-N6	Q-N6
31. R-KN1	R-Q3(h)
32. N-B8(i)	R-Q1(j)
33. NxR	B-N6
34. R-R3	P-KN5
35. Q-N6(k)	R-Q2
36. RxB	Q-R7
37. N-N6ch(1)	PxN
38. K-B2	Q-B5ch
39. K-K2	Q-R7
40. K-B2	Q-B5ch
41. K-K2	Q-R7
42. P-Q5(m)	KNPxP
43. PxQBP	R-K2
44. Q-Q1ch	R-K1
45. QxRch	K-N2
46. RxPch	K-R3
47. Q-R5 mate	

Alan Benson's Annotations

(a) In *Sargon's* programming these rook pawn moves appeared often. Normal would be 5. P-K3.

(b) A miscalculation.

(c) Missing a golden opportunity with 10 . . . PxN! 11. PxB (Best is 11. PxP immediately giving back a piece) 11 . . . . PxP and now white's best is 12. N-Q3 PxR=Q 13. QxQ since 12. R-N1 B-N5ch and mates next move, or 12. QxQ1ch RxQ13. R-Q1 B-N5ch 14. R-Q2 P-N8=Q mate.

(d) This move eventually loses a tempo. Practically speaking black should play 17 . . . . Q-B2 followed by developing the rooks to the center files.

(e) Strange for the white king to be taking a casual walk into the center.

(f) 26.K-N1 would be much safer.

(g) Here 28 . . . . QR-K1 followed possibly by 29 . . . . B-Q4 putting pressure on white's KP would have given the best practical chances.

(h) 31 . . . . R-QB2 was better.

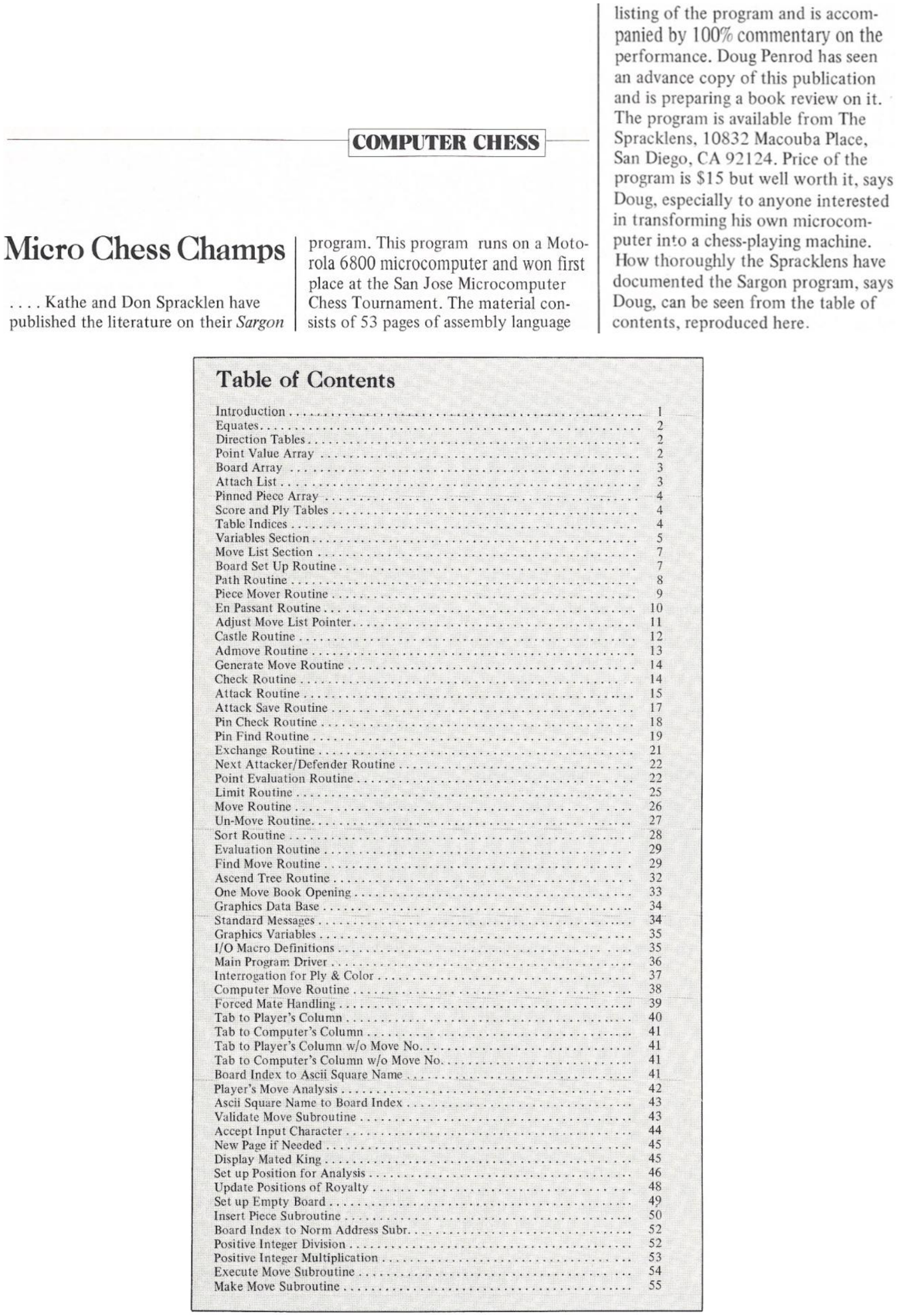
(k) Very nice – attacks both the rook and bishop.

(l) Also possible was 37 K-B2 Q-B5ch 28. R-KB3 Q-Q7ch 39. K-N3 RxP 40. Q-B7 R-Q6 41. Q-B8ch(If 41 RxR?? Q-B5 mate) 41 . . . . R-Q1 42. Q-K6 winning easily.

(m) *Sargon* was allowed a three move look ahead with this move to avoid the repetition of moves. It plays the final part of this game very well.

Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Personal Computing – August 1978



COMPUTER CHESS

Micro Chess Champs

. . . . Kathe and Don Spracklen have published the literature on their *Sargon* program. This program runs on a Motorola 6800 microcomputer and won first place at the San Jose Microcomputer Chess Tournament. The material consists of 53 pages of assembly language listing of the program and is accompanied by 100% commentary on the performance. Doug Penrod has seen an advance copy of this publication and is preparing a book review on it. The program is available from The Spracklens, 10832 Macouba Place, San Diego, CA 92124. Price of the program is $15 but well worth it, says Doug, especially to anyone interested in transforming his own microcomputer into a chess-playing machine. How thoroughly the Spracklens have documented the Sargon program, says Doug, can be seen from the table of contents, reproduced here.

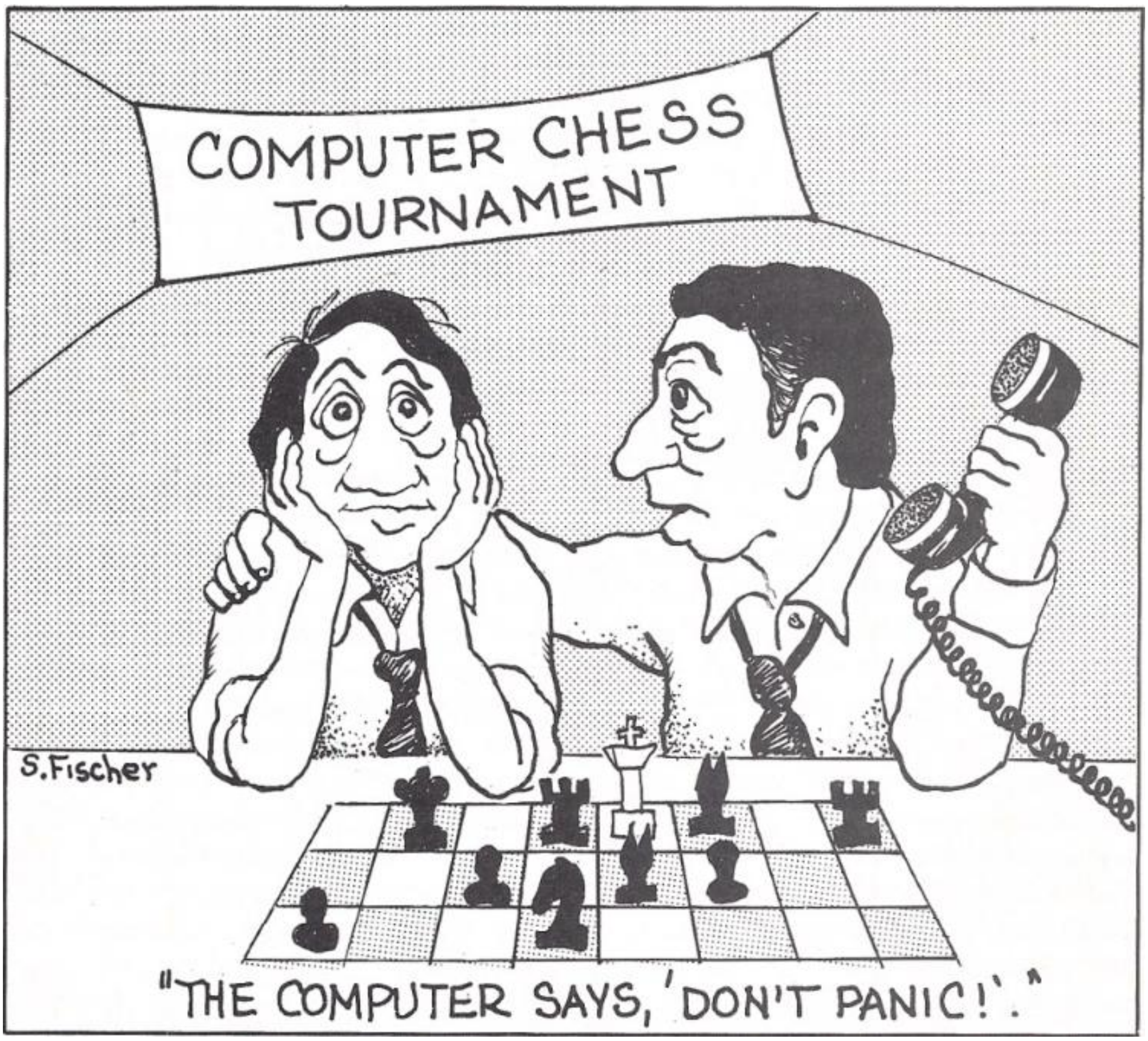
Table of Contents

Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Personal Computing – August 1978

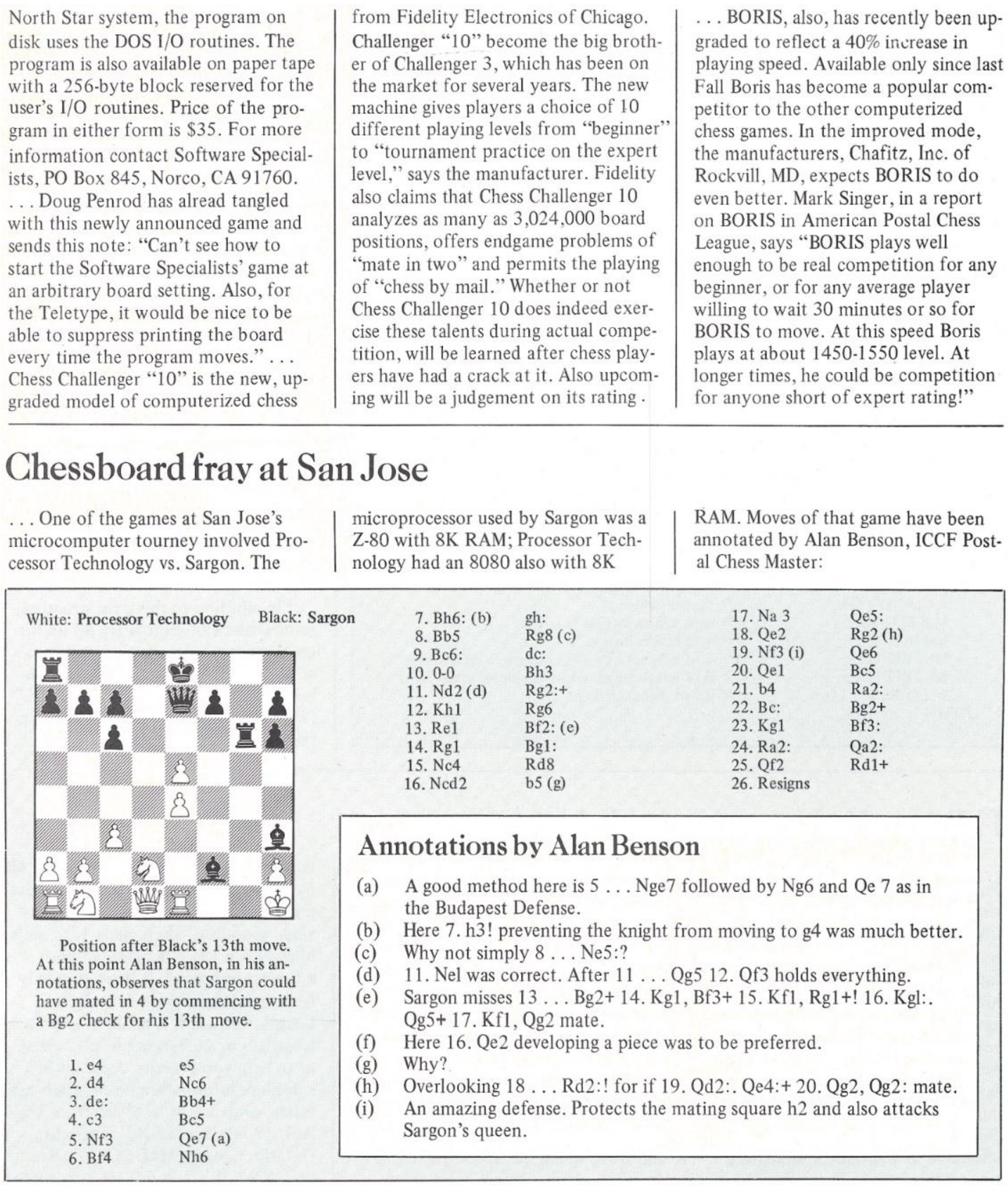
Tidbits of information and chess chatter

. . . Sargon, winner of the San Jose Microcomputer Chess Tourney, is reportedly coming out with Sargon II, a stronger version of the original program. . . . CYBERCHESS, a new chess-playing "machine" that is not a machine at all and uses no electricity, is manually operated by inserting preprogrammed cards into the gadget and manipulating these cards to reveal player's or "machine's" next move. Rating-levels of cards are graded from 900 to 2200 and the "machine" is claimed to be a means of improving your game of chess. For more information write to Cyber, PO Box 2066, Cerritos, CA 90701. Basic price is $29.95 to which must be added handling charges plus applicable sales tax. . . . Logical Systems, PO Box 303, Minneapolis, MN, 55440, manufactures "CompuChess." This computerized chess-game finished in 9th place at San Jose's Microcomputer Chess Tourney in March of '78. It uses an F8 microprocessor (same as Boris and Chess Challenger) and has 2K ROM and ¼K RAM (compared to Chess Challenger's Leve II 4K ROM and ½K RAM.) CompuChess is available at around $170 direct from factory. It plays at 6 levels, solves mate-in-two problems and serves as an excellent teaching aid for children and adults, says the manufacturer. A note to above address will bring complete information on CompuChess and will explain its capabilities in more detail. . . . Microchess-1.5 cassettes were originally conceived as Microchess-1.0 programs which would play reasonably good games of chess using a minimum of computer hardware. The taped programs are written in Z-80 machine language and are designed to make optimum use of the features of the TRS-80 microcomputer. Each of the new upgraded programs is a full 4K bytes in length and includes a graphics driver to display the chess board on a video monitor. "Microchess-1.5," says Mirco-Ware Ltd., its manufacturers, "is a tireless opponent. It is always ready for a quick blitz game or a slow thoughtful one. It is ready at any time to assist you in learning to play chess, or to help you practice chess skills." For more information on program cassettes, drop a note to Micro-Ware Limited, 27 Firstbrooke Rd., Toronto, Ontario, Canada, M4E 2L2. . . . Software Specialists announces a computer chess program for 8080 and Z-80 based microcomputers. This assembly language program conforms to all rules and conventions, says the company, including castling, *en passant* captures and promotion of pawns. The entire program, including I/O routines, will run in 8K of RAM. For users with a


COMPUTER CHESS TOURNAMENT
S.Fischer
"THE COMPUTER SAYS, 'DON'T PANIC!'."

Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Personal Computing – October 1978



North Star system, the program on disk uses the DOS I/O routines. The program is also available on paper tape with a 256-byte block reserved for the user's I/O routines. Price of the program in either form is $35. For more information contact Software Specialists, PO Box 845, Norco, CA 91760. . . . Doug Penrod has alread tangled with this newly announced game and sends this note: "Can't see how to start the Software Specialists' game at an arbitrary board setting. Also, for the Teletype, it would be nice to be able to suppress printing the board every time the program moves." . . . Chess Challenger "10" is the new, upgraded model of computerized chess from Fidelity Electronics of Chicago. Challenger "10" become the big brother of Challenger 3, which has been on the market for several years. The new machine gives players a choice of 10 different playing levels from "beginner" to "tournament practice on the expert level," says the manufacturer. Fidelity also claims that Chess Challenger 10 analyzes as many as 3,024,000 board positions, offers endgame problems of "mate in two" and permits the playing of "chess by mail." Whether or not Chess Challenger 10 does indeed exercise these talents during actual competition, will be learned after chess players have had a crack at it. Also upcoming will be a judgement on its rating . . . . BORIS, also, has recently been upgraded to reflect a 40% increase in playing speed. Available only since last Fall Boris has become a popular competitor to the other computerized chess games. In the improved mode, the manufacturers, Chafitz, Inc. of Rockvill, MD, expects BORIS to do even better. Mark Singer, in a report on BORIS in American Postal Chess League, says "BORIS plays well enough to be real competition for any beginner, or for any average player willing to wait 30 minutes or so for BORIS to move. At this speed Boris plays at about 1450-1550 level. At longer times, he could be competition for anyone short of expert rating!"

Chessboard fray at San Jose

. . . One of the games at San Jose's microcomputer tourney involved Processor Technology vs. Sargon. The microprocessor used by Sargon was a Z-80 with 8K RAM; Processor Technology had an 8080 also with 8K RAM. Moves of that game have been annotated by Alan Benson, ICCF Postal Chess Master:

White: **Processor Technology** Black: **Sargon**

Position after Black's 13th move. At this point Alan Benson, in his annotations, observes that Sargon could have mated in 4 by commencing with a Bg2 check for his 13th move.

1.	e4	e5
2.	d4	Nc6
3.	de:	Bb4+
4.	c3	Bc5
5.	Nf3	Qe7 (a)
6.	Bf4	Nh6
7.	Bh6: (b)	gh:
8.	Bb5	Rg8 (c)
9.	Bc6:	dc:
10.	0-0	Bh3
11.	Nd2 (d)	Rg2:+
12.	Kh1	Rg6
13.	Re1	Bf2: (e)
14.	Rg1	Bg1:
15.	Nc4	Rd8
16.	Ncd2	b5 (g)
17.	Na 3	Qe5:
18.	Qe2	Rg2 (h)
19.	Nf3 (i)	Qe6
20.	Qe1	Bc5
21.	b4	Ra2:
22.	Bc:	Bg2+
23.	Kg1	Bf3:
24.	Ra2:	Qa2:
25.	Qf2	Rd1+
26.	Resigns	

Annotations by Alan Benson

(a) A good method here is 5 . . . Nge7 followed by Ng6 and Qe 7 as in the Budapest Defense.

(b) Here 7. h3! preventing the knight from moving to g4 was much better.

(c) Why not simply 8 . . . Ne5:?

(d) 11. Nel was correct. After 11 . . . Qg5 12. Qf3 holds everything.

(e) Sargon misses 13 . . . Bg2+ 14. Kg1, Bf3+ 15. Kf1, Rg1+! 16. Kg1:. Qg5+ 17. Kf1, Qg2 mate.

(f) Here 16. Qe2 developing a piece was to be preferred.

(g) Why?

(h) Overlooking 18 . . . Rd2:! for if 19. Qd2:. Qe4:+ 20. Qg2, Qg2: mate.

(i) An amazing defense. Protects the mating square h2 and also attacks Sargon's queen.

Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Personal Computing – October 1978

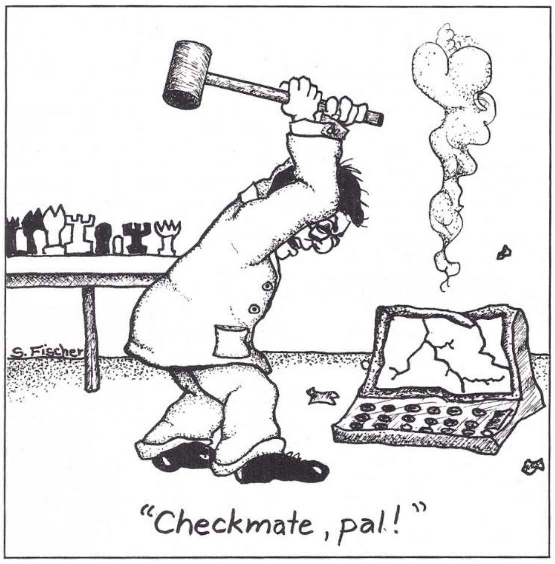
More on San Jose

. . . The following story appeared in CHESS VOICE, Official Publication of the North California Chess Association. It is the work of John Larkins, Editor of CHESS VOICE and Larry Wagner, director of the San Jose Tournament:

"For centuries,men have competed with each other over the chessboard. Then, with the advent of chess-playing computers, they started to play machines. Now the machines are playing each other. There have been national and international computer chess championships for several years. (The current world champion computer is America's *Chess 4.6,* which is now consistently playing at an expert level.) But these tourneys involve an assembly of computer terminals each connected by telephone hookups to remote machines, some thousands of miles away, whose value could be measured in millions of dollars. (It costs $38/second just to run the *Chess 4.6* program.)

"The most recent development in this field is the appearance of a number of chess-playing microcomputers. These are small, self-contained, relatively inexpensive machines that play chess at about Class D or E strength. Some are marketed commercially for the sole purpose of playing chess; others are programs that can be used with personal computers designed to carry out a variety of other tasks as well.

"The world's first microcomputer chess tournament (machines vs. machines) was held March 3-5 in San Jose at the 2nd West Coast Computer Faire. The event was organized and directed by Larry Wagner, assisted from the computer world by John Keary, Alan Miller, Ian Shepperd, Larry Kaplan, Craig Asher, Brad Stewart, John Mills and Daryl Elder, assisted from the chess community by Alan Benson, John Larkins, and John Spargo.



"The tournament had 11 participants, each with a distinct program. Five of the machines have been designed solely for playing chess. Three are already commercially available: 1) *Boris* ($300), 2) *CompuChess* ($160-220), and 3) *Chess Challenger* ($220-280). A fourth is scheduled to appear on the market this Fall – 4) *Commodore Chessmate* ($150-225). The fifth entry in this group was a home-built device – 5) *Steve Stuart's Metal Box,* with parts that cost a mere $85.

"An additional six chess-playing programs were used with personal or hobby computers. The programs cost in the neighborhood of $20; the computers vary from about $2,000 to $7,000. Two of these programs are commercially available: 6) *Processor Technology* and 7) *Compucolor.* Four others are individually-developed programs: 8) *SARGON,* 9) *Mark Watson's Program,* 10) *SD Chess,* and 11) *Tenberg Basic.*

"After the first round it became apparent that some of the machines programmed in BASIC could not keep up with the tournament schedule of two games per day. (Two of them were paired with each other for a single nine-hour game.) Unfortunately, this undermined the pairings and the clarity of the results, since, after six "rounds", some machines had played 5 games, some 4 games, and one – a single game.

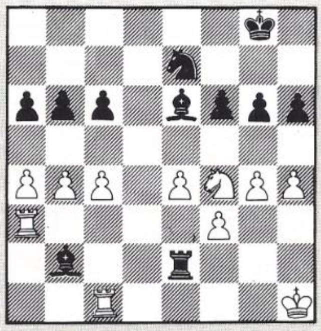
"However, there was a clear winner – SARGON, which won all of its five games. Its nearest competitors were Commodore Chessmate (2 wins, 2 draws, 1 loss), Boris (2 wins, 1 draw, 1 loss, 1 forfeit), and Chess Challenger (3 wins, 2 losses). SARGON defeated both Commodore Chessmate and Chess Challenger. Boris beat Chess Challenger, but lost to Commodore Chessmate. Chess Challenger got its three wins from two Class C machines plus Stewart's Metal Box.

"Among the unique problems encountered were the following: There were insufficient electrical cords and outlets to service all the electricity-gobbling machines. And, when all the machines had been hooked up, several had their cords inadvertantly kicked out of their sockets by passing feet. This completely erases the computer's memory of the game and requires all the necessary information to be reentered before the game can resume. Particularly troublesome were the difficulties in communicating moves. Since it is hard to shift the computers around, the moves are relayed verbally by the operators. But most of them are unfamiliar with standard chess notation and have evolved notation systems of their own, which are unknown to each other. Thus many errors in translation and notation were made – again resulting in having to restart several games where neither side was sure what the correct position was. Some of the machines play only the Black side of the board; others can play White, only if special arrangements are made. This raises havoc with the pair-

Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Personal Computing – November 1978

White – Processor Technology	Black – Commodore Chess Mate
1. e4	e5
2. d4	ed:
3. Qd4:	Nc6
4. Qd5(a)	Bb4+
5. Nc3	Nge7
6. Qd3	O-O
7. Nf3	d6 (b)
8. Bf4	Ne5 (c)
9. Be5:	de:
10. Qd8:	Rd8:
11. Ne5:	Rd4
12. f3	Be6
13. Be2	Rad8
14. O-O	Rd2
15. Bd3	Rd4
16. Nb1 (d)	Bc5 (e)
17. Nd2: (f)	Rd3:+
18. Kh1	Rd2:
19. Rfc1	f6
20. Nd3	Bd4
21. h4	c6
22. g4	a6
23. a4	g6
24. Ra3	b6
25. b4	h6
26. c4	Re2
27. Nf4(g)	Bb2
28. Ne2:	Ba3:
29. Rc3	Bb4:
30. Rc2	a5
31. Nd4	Bf7
32. f4	c5
33. Nf3	Nc6
34. Kg2	Nd4
35. Nd4: (h)	cd:
36. Kf3	d3
37. Rc1	d2
38. Rc2 (i)	d1=Q+
39. Re2	Bc4:
40. h5	Qe2:+
41. Kg3	Be1+
42. Resigns	



Position after Black's 27th move. Both players have achieved a forking position. Who profits from the resulting exchange? Read on.

Annotations by Alan Benson

(a) This opening theory only considers two main continuations here: 4. Qe3 and 4. Qa4.

(b) Why not 7 . . . d5 with the initiative?

(c) Loses a pawn.

(d) A fine move winning the exchange.

(e) Setting a little trap.

(f) Falls right into it! Instead 17. Kh1 keeps the material edge.

(g) The "knight fork" to which black has a clever resource.

(h) This makes it easy for black. Better was 35. Rc1 but after 35 . . . Be8 36. Ra1, Nc2 37. Ra2, Ne3+ followed by 38 . . . Nc4: black is winning.

(i) A surprising move. It's lost in any event as after 38. Rd1, Bc4: followed by 39 . . . Bb3 winning the rook.

ings — as did the occasional unavailability of operators who had other things to do at the Computer Faire.

"Compared to typical human tournament players, all the microcomputers tend to lack the kind of killer instinct needed to finish off an opponent once he (it) is on the ropes. Ten of the 22 games played at San Jose had to be abandoned midstream. Four ended when one machine was not able to make the time control: four more ended by adjudication after four hours of play; and two games were declared a draw "by lock-up" when one machine went into a cycle of repetitive moves. (The 22 completed games were all mates. None of the microcomputers is programmed to offer a draw or resign.)

"Special rules were set up to deal with the lock-up problem. The typical lock-up will occur in a rook and pawn endgame where one side has a won game but nevertheless insists on an endless series of rook checks. (A lock-up is like a hiccup.) To prevent these games from ending in a draw, the operator was allowed to increase the look-ahead capacity of the locked-up machine, with the hope that it could then see it wasn't getting anywhere, and try something different.

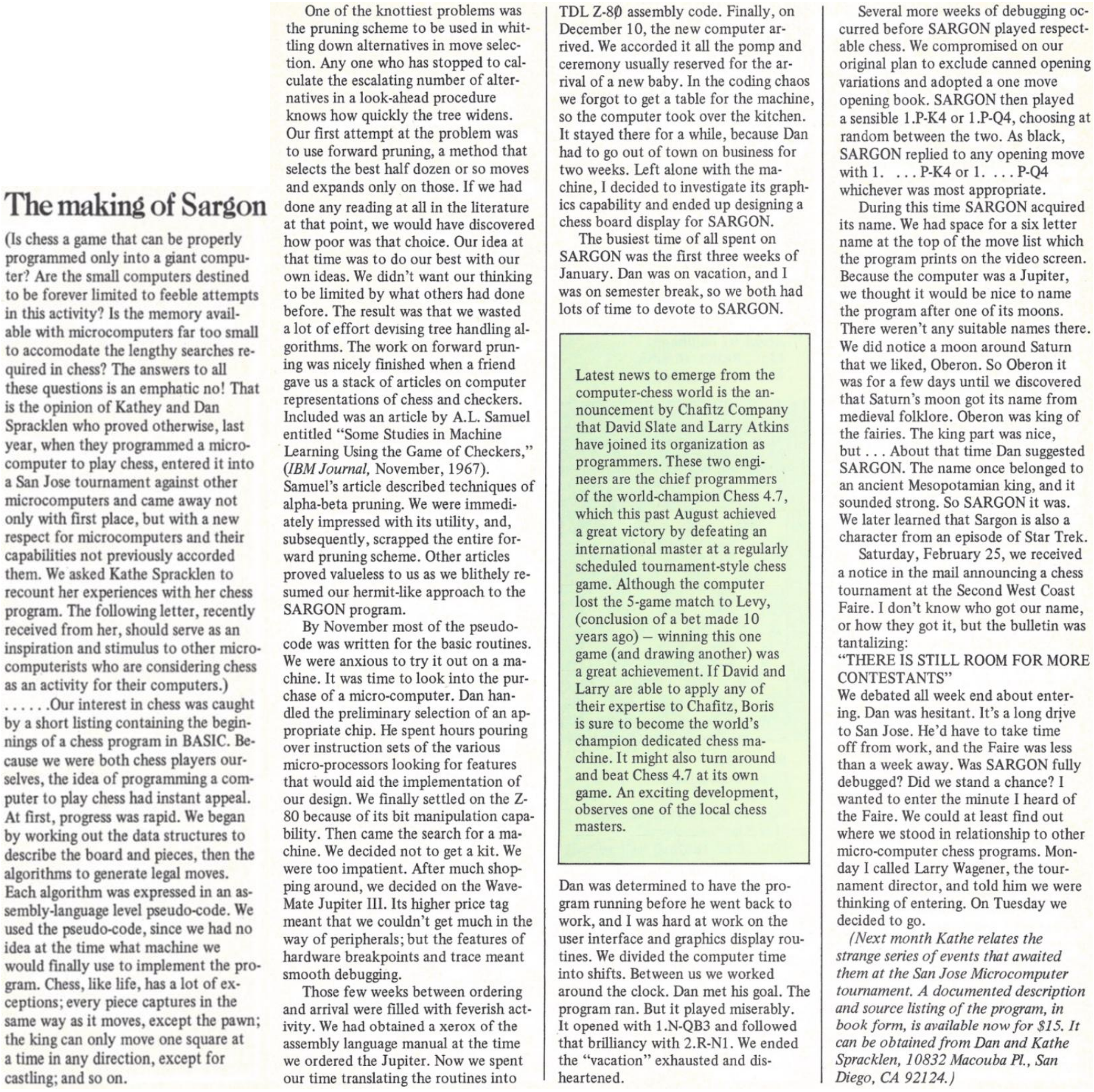
"Most chess-playing microcomputers have a variable look-ahead capacity. But they pay the price for deeper tree searches in greatly decreased response time. CompuChess, for example, is good at solving mate-in-two problems with its Level 6 mode — but it can take up to two days per move! Most of the machines at San Jose were playing below their maximum theoretical look-ahead capacity in order to meet the time control. CompuChess played at Level III; Boris at 2 minutes per move. In two cases, even with increased look-ahead, the machine continued to repeat itself — these are the two draws by Commodore Chessmate. In a number of other cases, the hiccup response was bypassed, allowing the machine with the extra material to survive to the four-hour time limit, where it could win by adjudication.

"Often the microcomputers play chess just like people do. But every once in a while, one gets the sense of a definitely alien intelligence at work. An amusing example occurred in the game between Processor and Commodore. After Black's 20th move they had arrived at a quiet but puzzling middle game position. Neither side could come up with a game plan, and, in the absence of any obvious tactical shots, both began an almost random pushing of wing pawns. Curiously, Processor appeared to have been taught to push its pawns two squares, when in doubt, while Commodore had been told to push them one square. (See moves 21 to 26 on preceding page.)

"Since the San Jose tournament, SARGON has undergone total revision. It now sees ahead twice as far and is called SARGON II. (The program is available for $15 from the Spracklins.) Steve Stewart, too, has a new program and a new metal box. Mark Watson is translating his BASIC program into assembly language." (Subscriptions to CHESS VOICE, or information, can be obtained by writing to Editor John Larkins, CHESS VOICE, 5804 Ocean View Drive, Oakland, CA 94618.)

Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Personal Computing – November 1978



The making of Sargon

(Is chess a game that can be properly programmed only into a giant computer? Are the small computers destined to be forever limited to feeble attempts in this activity? Is the memory available with microcomputers far too small to accomodate the lengthy searches required in chess? The answers to all these questions is an emphatic no! That is the opinion of Kathey and Dan Spracklen who proved otherwise, last year, when they programmed a microcomputer to play chess, entered it into a San Jose tournament against other microcomputers and came away not only with first place, but with a new respect for microcomputers and their capabilities not previously accorded them. We asked Kathe Spracklen to recount her experiences with her chess program. The following letter, recently received from her, should serve as an inspiration and stimulus to other microcomputerists who are considering chess as an activity for their computers.)

. . . . . .Our interest in chess was caught by a short listing containing the beginnings of a chess program in BASIC. Because we were both chess players ourselves, the idea of programming a computer to play chess had instant appeal. At first, progress was rapid. We began by working out the data structures to describe the board and pieces, then the algorithms to generate legal moves. Each algorithm was expressed in an assembly-language level pseudo-code. We used the pseudo-code, since we had no idea at the time what machine we would finally use to implement the program. Chess, like life, has a lot of exceptions; every piece captures in the same way as it moves, except the pawn; the king can only move one square at a time in any direction, except for castling; and so on.

One of the knottiest problems was the pruning scheme to be used in whittling down alternatives in move selection. Any one who has stopped to calculate the escalating number of alternatives in a look-ahead procedure knows how quickly the tree widens. Our first attempt at the problem was to use forward pruning, a method that selects the best half dozen or so moves and expands only on those. If we had done any reading at all in the literature at that point, we would have discovered how poor was that choice. Our idea at that time was to do our best with our own ideas. We didn't want our thinking to be limited by what others had done before. The result was that we wasted a lot of effort devising tree handling algorithms. The work on forward pruning was nicely finished when a friend gave us a stack of articles on computer representations of chess and checkers. Included was an article by A.L. Samuel entitled "Some Studies in Machine Learning Using the Game of Checkers," (*IBM Journal,* November, 1967). Samuel's article described techniques of alpha-beta pruning. We were immediately impressed with its utility, and, subsequently, scrapped the entire forward pruning scheme. Other articles proved valueless to us as we blithely resumed our hermit-like approach to the SARGON program.

By November most of the pseudo-code was written for the basic routines. We were anxious to try it out on a machine. It was time to look into the purchase of a micro-computer. Dan handled the preliminary selection of an appropriate chip. He spent hours pouring over instruction sets of the various micro-processors looking for features that would aid the implementation of our design. We finally settled on the Z-80 because of its bit manipulation capability. Then came the search for a machine. We decided not to get a kit. We were too impatient. After much shopping around, we decided on the Wave-Mate Jupiter III. Its higher price tag meant that we couldn't get much in the way of peripherals; but the features of hardware breakpoints and trace meant smooth debugging.

Those few weeks between ordering and arrival were filled with feverish activity. We had obtained a xerox of the assembly language manual at the time we ordered the Jupiter. Now we spent our time translating the routines into TDL Z-8Ø assembly code. Finally, on December 10, the new computer arrived. We accorded it all the pomp and ceremony usually reserved for the arrival of a new baby. In the coding chaos we forgot to get a table for the machine, so the computer took over the kitchen. It stayed there for a while, because Dan had to go out of town on business for two weeks. Left alone with the machine, I decided to investigate its graphics capability and ended up designing a chess board display for SARGON.

The busiest time of all spent on SARGON was the first three weeks of January. Dan was on vacation, and I was on semester break, so we both had lots of time to devote to SARGON.

Latest news to emerge from the computer-chess world is the announcement by Chafitz Company that David Slate and Larry Atkins have joined its organization as programmers. These two engineers are the chief programmers of the world-champion Chess 4.7, which this past August achieved a great victory by defeating an international master at a regularly scheduled tournament-style chess game. Although the computer lost the 5-game match to Levy, (conclusion of a bet made 10 years ago) — winning this one game (and drawing another) was a great achievement. If David and Larry are able to apply any of their expertise to Chafitz, Boris is sure to become the world's champion dedicated chess machine. It might also turn around and beat Chess 4.7 at its own game. An exciting development, observes one of the local chess masters.

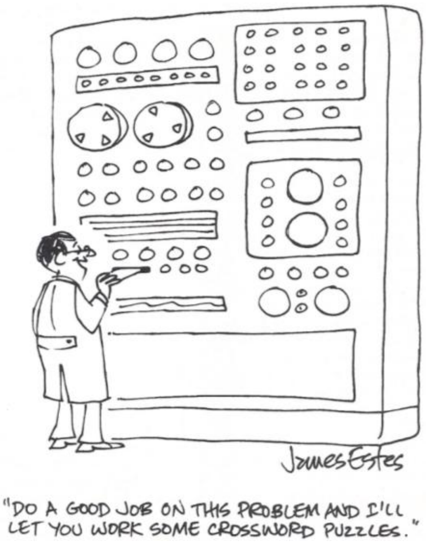
Dan was determined to have the program running before he went back to work, and I was hard at work on the user interface and graphics display routines. We divided the computer time into shifts. Between us we worked around the clock. Dan met his goal. The program ran. But it played miserably. It opened with 1.N-QB3 and followed that brilliancy with 2.R-N1. We ended the "vacation" exhausted and disheartened.

Several more weeks of debugging occurred before SARGON played respectable chess. We compromised on our original plan to exclude canned opening variations and adopted a one move opening book. SARGON then played a sensible 1.P-K4 or 1.P-Q4, choosing at random between the two. As black, SARGON replied to any opening move with 1. . . . P-K4 or 1. . . . P-Q4 whichever was most appropriate.

During this time SARGON acquired its name. We had space for a six letter name at the top of the move list which the program prints on the video screen. Because the computer was a Jupiter, we thought it would be nice to name the program after one of its moons. There weren't any suitable names there. We did notice a moon around Saturn that we liked, Oberon. So Oberon it was for a few days until we discovered that Saturn's moon got its name from medieval folklore. Oberon was king of the fairies. The king part was nice, but . . . About that time Dan suggested SARGON. The name once belonged to an ancient Mesopotamian king, and it sounded strong. So SARGON it was. We later learned that Sargon is also a character from an episode of Star Trek.

Saturday, February 25, we received a notice in the mail announcing a chess tournament at the Second West Coast Faire. I don't know who got our name, or how they got it, but the bulletin was tantalizing:
"THERE IS STILL ROOM FOR MORE CONTESTANTS"
We debated all week end about entering. Dan was hesitant. It's a long drive to San Jose. He'd have to take time off from work, and the Faire was less than a week away. Was SARGON fully debugged? Did we stand a chance? I wanted to enter the minute I heard of the Faire. We could at least find out where we stood in relationship to other micro-computer chess programs. Monday I called Larry Wagener, the tournament director, and told him we were thinking of entering. On Tuesday we decided to go.

(Next month Kathe relates the strange series of events that awaited them at the San Jose Microcomputer tournament. A documented description and source listing of the program, in book form, is available now for $15. It can be obtained from Dan and Kathe Spracklen, 10832 Macouba Pl., San Diego, CA 92124.)



"DO A GOOD JOB ON THIS PROBLEM AND I'LL LET YOU WORK SOME CROSSWORD PUZZLES."

Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Personal Computing – December 1978

The Playing of Sargon

(Last month Kathe Spracklen recounted the problems she and her husband, Dan, faced as they attempted to feed their chess program into the computer. She continues now with the events they encountered when they arrived at San Diego to participate in the microcomputer-chess tournament there.)

"Dan arranged for two days off from work and we left Thursday morning for the day-long drive. The weather was dreadful. Heavy rains followed us all the way up the coast. (In fact, the route we traveled was closed at Santa Barbara just a few hours after we had passed through because of heavy mud slides.) But our mood was as bright as the day was dark. Openly we assured each other that we would be happy if we managed to finish in the top half; although both of us secretly hoped for a first place victory. At 10 P.M. we arrived at my brother's apartment. Hot pizza, cold beer, and excited speculations filled the late evening hours.

"Friday morning at the Faire was a chaotic experience. The room assigned to the chess tournament was barely large enough to house the contestants. Ample electrical outlets had not been provided, nor were there enough extension cords to reach to all the machines. Some programs could not be brought up because of electrical interference. SARGON was set up in a corner and paired with two of the commercial boxes, Compu-Chess and Chess Challenger, for the first two rounds. The game with Compu-Chess was adjudicated a win after 58 moves. For SARGON it was the shakiest play of the whole tournament. Compu-Chess (set at level 4) had a pawn sitting on the 7th rank ready to queen, but fortunately for us, never pushed it. Finally SARGON captured that pawn, ending our worries. SARGON checkmated Chess Challenger (level 3) in 42 moves. On move 25 SARGON trapped Chess Challenger's Queen, and captured it on the following move. The room was jammed with crowds of people filtering through all day.

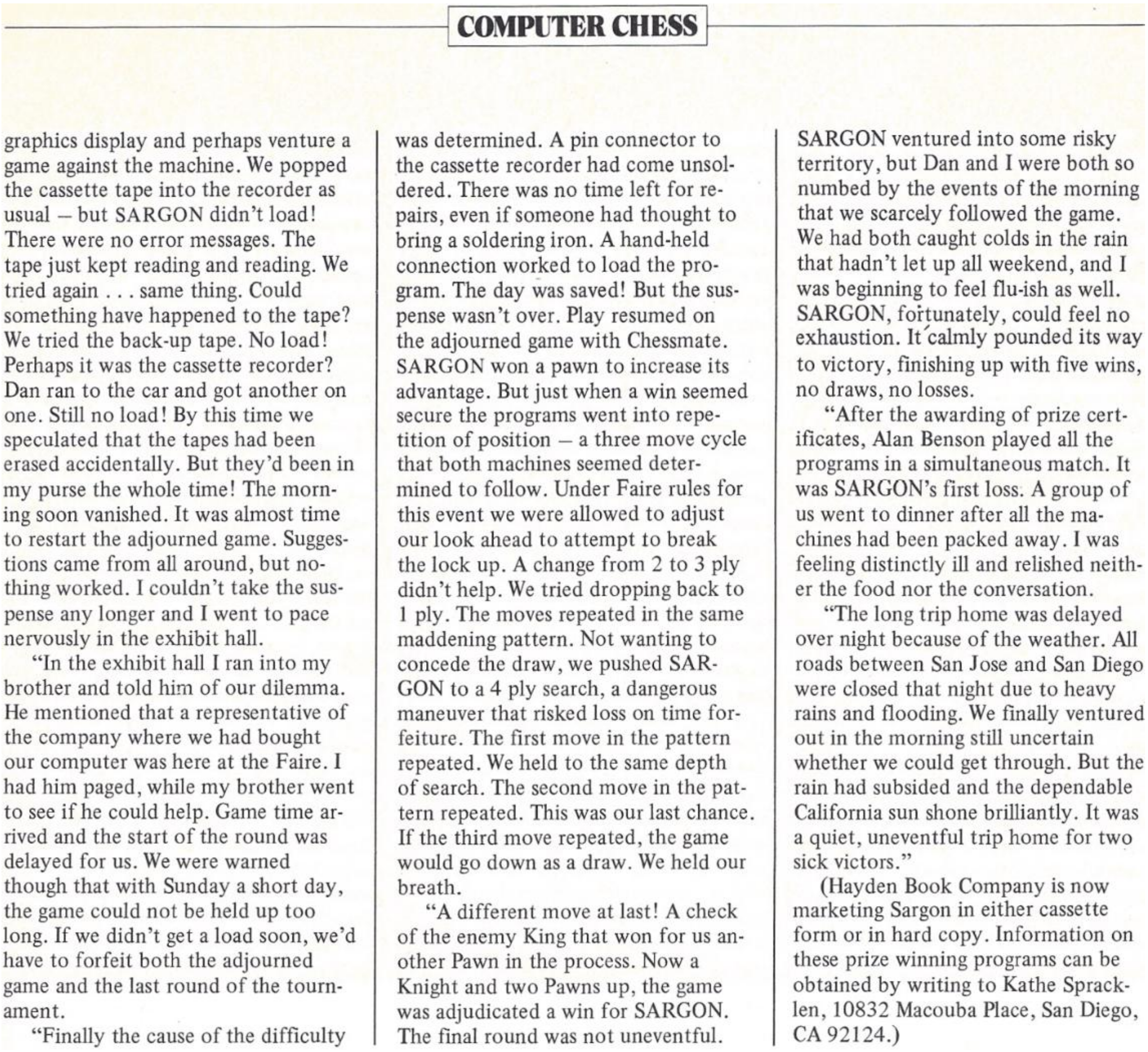
"With more extension cords available on Saturday, and more crowds expected, Larry Wagener and Roy Elder, tournament directors, decided on a rearrangement. Tables were placed in a semi-circle against one wall. The idea was to allow maximum room for spectators to filter through with minimum chance for game disruptions because of knocked out plugs. SARGON, now a two game winner, was beginning to draw more comment. We were paired with Processor Technology's 8080 Chess, fresh from its debut in Seattle against the giant computer there. The Processor Technology program was the early heavy favorite. In the tournament, so far, it had drawn a game with Boris and won against Compu-Color's program. Surprisingly, SARGON found it the easiest program to defeat. Processor Technology resigned after 25 moves, faced with a forced mate. During play of the game, a surprising Bishop-move into the line of attack of a pawn stunned assembled onlookers. A second look revealed that the pawn was pinned against the enemy king and was not only powerless, but doomed as well. The move was an emotional victory for Dan, since the pin finder routine had been the most difficult to debug and he had just put the finishing touches on it the night before we left for the Faire.

"The next round found us paired against a prototype model of Commodore's new Chessmate. The struggle went on all afternoon. Eventually SARGON won a Knight, but the outcome was still not clear after more than 60 moves. It was decided to adjourn the game until Sunday morning.

"On Sunday the Faire was scheduled to open at noon. The morning was our first chance to tour the exhibits. We arrived early and decided, before touring, to bring SARGON up so that those visiting the tournament room before game time could see the

Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Personal Computing – January 1979



graphics display and perhaps venture a game against the machine. We popped the cassette tape into the recorder as usual – but SARGON didn't load! There were no error messages. The tape just kept reading and reading. We tried again . . . same thing. Could something have happened to the tape? We tried the back-up tape. No load! Perhaps it was the cassette recorder? Dan ran to the car and got another on one. Still no load! By this time we speculated that the tapes had been erased accidentally. But they'd been in my purse the whole time! The morning soon vanished. It was almost time to restart the adjourned game. Suggestions came from all around, but nothing worked. I couldn't take the suspense any longer and I went to pace nervously in the exhibit hall.

"In the exhibit hall I ran into my brother and told him of our dilemma. He mentioned that a representative of the company where we had bought our computer was here at the Faire. I had him paged, while my brother went to see if he could help. Game time arrived and the start of the round was delayed for us. We were warned though that with Sunday a short day, the game could not be held up too long. If we didn't get a load soon, we'd have to forfeit both the adjourned game and the last round of the tournament.

"Finally the cause of the difficulty was determined. A pin connector to the cassette recorder had come unsoldered. There was no time left for repairs, even if someone had thought to bring a soldering iron. A hand-held connection worked to load the program. The day was saved! But the suspense wasn't over. Play resumed on the adjourned game with Chessmate. SARGON won a pawn to increase its advantage. But just when a win seemed secure the programs went into repetition of position – a three move cycle that both machines seemed determined to follow. Under Faire rules for this event we were allowed to adjust our look ahead to attempt to break the lock up. A change from 2 to 3 ply didn't help. We tried dropping back to 1 ply. The moves repeated in the same maddening pattern. Not wanting to concede the draw, we pushed SARGON to a 4 ply search, a dangerous maneuver that risked loss on time forfeiture. The first move in the pattern repeated. We held to the same depth of search. The second move in the pattern repeated. This was our last chance. If the third move repeated, the game would go down as a draw. We held our breath.

"A different move at last! A check of the enemy King that won for us another Pawn in the process. Now a Knight and two Pawns up, the game was adjudicated a win for SARGON. The final round was not uneventful. SARGON ventured into some risky territory, but Dan and I were both so numbed by the events of the morning that we scarcely followed the game. We had both caught colds in the rain that hadn't let up all weekend, and I was beginning to feel flu-ish as well. SARGON, fortunately, could feel no exhaustion. It calmly pounded its way to victory, finishing up with five wins, no draws, no losses.

"After the awarding of prize certificates, Alan Benson played all the programs in a simultaneous match. It was SARGON's first loss. A group of us went to dinner after all the machines had been packed away. I was feeling distinctly ill and relished neither the food nor the conversation.

"The long trip home was delayed over night because of the weather. All roads between San Jose and San Diego were closed that night due to heavy rains and flooding. We finally ventured out in the morning still uncertain whether we could get through. But the rain had subsided and the dependable California sun shone brilliantly. It was a quiet, uneventful trip home for two sick victors."

(Hayden Book Company is now marketing Sargon in either cassette form or in hard copy. Information on these prize winning programs can be obtained by writing to Kathe Spracklen, 10832 Macouba Place, San Diego, CA 92124.)

Microcomputer Tournament in San Jose held March 3-5, 1978

Source: Personal Computing – January 1979

External & Internal Links:

https://www.chessprogramming.org/MCCT_1978

https://alchetron.com/Sargon-(chess)

https://archive.computerhistory.org/projects/chess/related_materials/oral-history/spacklen.oral_history.2005.102630821/spracklen.oral_history_transcript.2005.102630821.pdf

https://www.schaakcomputers.nl/hein_veldhuis/database/files/01-2016,%20Rob%20van%20Son,%20Sargon%20fought%20until%20the%20very%20end!.pdf

https://www.schaakcomputers.nl/hein_veldhuis/database/files/03-1989,%20Modul,%20Goran%20Grottling,%20Ein%20Interview%20mit%20Kathe%20Spracklen.pdf

https://www.schaakcomputers.nl/hein_veldhuis/database/files/05-1983,%20Martin%20Gittel%20-%20SARGON%20Portrat%20eines%20Schach-Programms.pdf

https://www.schaakcomputers.nl/hein_veldhuis/database/files/10-2011,%20Rob%20van%20Son,%20De%20Assyrische%20krijgsman%20die%20vaak%20won.pdf