Issue #2 **Price: 75¢, but $1.25 Overseas**

Issue #1 has been sent to 447, and Issue #2 has been requested by 136 computer chess fans so far, with requests for both arriving daily. I delayed preparation of Issue #2, hoping to be able to go to Toronto the the IFIPS meeting, where the 2nd World Computer Chess Tournament was held last month, but had to stay here for radiation treatment. I may, however, attend the ACM meeting in Seattle next month, where the annual U.S. Computer Chess Tournament will be held.

In Issue #1 I omitted two books from the list:

"The World Computer Chess Championship" by Jean Hayes and David Levy, 1976, and "Advances In Computer Chess", edited by Clarke, 1977. Both are published by Edinburgh University Press.

I must thank Don Cyr and Russ McNiel for their help.

Russ sent his Chess Challenger to the factory with $75, and it was returned with improvements including proper notation and 2 more levels of play; even the first level is improved. Tom Crispin and Dennis Cooper have played against it; I hope they'll have some comments in the next issue.

Another chess machine and another available program are mentioned in this issue. I hope that available machines and programs will be entered in enough (human) tournaments to get USCF ratings.

John Ford, Tom Crispin, and Dennis Cooper (all within my reach) have promised articles for future issues, as have several qualified correspondents, so help for beginners should be on the way. In the meantime the listed books and the material listed in their bibliogaphies should provide enough to keep one busy. -- Doug Penrod, editor

Paul Copeland
2 Stephen Crescent
Croydon Victoria
Australia 3136

Dear Mr. Penrod,

I read about your computer chess newsletter in Kilobaud. Enclosed is a cheque to cover costs of the first newsletter.

Although I do not have a microcomputer, I am very keen to obtain a chess program available in Basic. Perhaps you could advise.

I have never played chess against a computer, and what I have to say regarding programs is what I would like to see in an utopian chess playing maching. Suggestions-----

1. The program would be written in such a way that the computer responses to moves are never identical. At each computer move, the computer should be able to randomly choose between two 'best' moves.
2. For endgame study, the player should be able to enter an end game position into the computer. Play would commence from the position entered.
3. It would be useful if the player could at any time ask the computer for an analysis of the next four or five moves with variations. To this request, the computer would respond with black and white's next four moves or so.
4. The computer could be programmed to play against itself (with minimal time delay between moves). This would or should result in drawn games.
5. After checkmate, or a drawn game, the computer should be able to go through the game again pointing out to the player the good and bad moves that were played.

You may wish to mention some of these points in the next issue.

The unidentified "Tom", author of a letter printed in the first issue and of the program Patzer 451, is Tom Crispin, now at P.O. Box 1055, Goleta, CA 93017. Here are excerpts from another letter from Tom:

My experience with Bobby Fischer's Chess Challenger is that it should be considered weaker than 1000, perhaps as low as 700. It falls for almost any two-mover. (ed. note: Tom Crispin and Dennis Cooper have played against Russ McNiel's upgraded Chess Challenger and agree that it is much stronger).

I'll be writing an article or two on BASIC chess programs. However, for a program as complex as chess - complex to make it play well - BASIC as an interpretive language is much too slow. As I mentioned, it currently plays at about 2 min. a move, but a FORTRAN version on the same microprocessor should be 10-50 times faster.

My own preference is to sell listings of the program directly to micro-owners. It seems stupid to me to let large companies sell games packages to TV owners when for a little more $ the same owner could have a micro, with all the same games: but also a computer. I want the micro-industry to follow the pattern of hi-fi. Software should be sold much like LP's - if it isn't simply placed in the public domain.

I am hoping to collaborate with Bobby on the chess programming. I can provide the equipment and programming expertise; he could provide a somewhat better evaluation of the computer's play than I (though as I am rated somewhere near 2100 I'm not too bad in that department).

Part of my interest in the chess programs stems from my interest in AI. I want to work on either that aspect or games programming - or perhaps to develop a better BASIC with compiling options - it would be so useful in scientific work.

The limits of the Shannon approach (worth an article) have really only surfaced as a result of the past work. There is an article in the book Advances In Computer Chess (pub. by Edinburg Univ, edited by Clarke, 1977, $9.66) by the KAISSA programmers discussing better methods of tree pruning. My interest is to write a program that does no tree search. The motivation comes from the observation (this is yet another article!) that masters and grandmasters play rather good quality chess at blitz tempo - if the tempo is fast enough we can be reasonably certain that they are not searching a tree. True, they often make tactical oversights, but some form of pattern recognition allows them to play three, four or longer move combinations. Also, their positional play is often exceptional in fast games. So - why not try to emulate that aspect of master play on the micros? Suppose that, purely with my "static" approach, I achieved 1490 level chess on an 8080. Put the program on a Cyber 176, add a tree search, and I bet we have a master level program.

The Greenblatt program played in a number of (human) tournaments and has a rating. I would not accept as definitive any computer/program rating that is not based on tournament play. The human player should have something at stake to avoid his experimenting to see what the computer will do.

Michael Klos
Box 1053
Cortland, New York 13045

Mr. Penrod.

Enclosed is a money order for $1.00 to help cover costs. I would prefer to pay a little extra for first class mail.

I have copies of almost every article on computer chess, and some thesis papers. I will make photo copies of any for cost for any of your readers. The few I am missing I am trying to get. If someone has those I'd be glad to trade, etc.

If I can help you in any way please let me know.

C. Robert Leach *Systems Manager*
PETERBOROUGH NH 03458 (603) 924-3873

Kilobaud

Dear Doug:

Congratulations on your first issue of the newsletter! Well done! I'm sure each succeeding issue will be better than the last.

Doug, I would appreciate your passing along the following information to the letter writer named "Tom" and to anyone else you think might be interested.

> The Kilobaud Software Library (soon to be initiated) is interested in locating high quality programs....systems programs....application programs.....CHESS programs...for all major hobbyist systems.
>
> We will pay programmers a royalty (quarterly) of 15% , and will distribute cassette tapes of the programs nationwide.
>
> Any interested parties should contact me at the Kilobaud Microcomputer Lab, 73 Pine St. Peterborough, New Hampshire 03458.

Thanks, Doug, and best of luck with your new venture!

Dear Doug,

I'm glad to see something started in a computer chess newsletter. As you know my goal was to be able to play a fair game of chess at home with a computer. Not knowing much about software or hardware it seemed nigh onto impossible. As a starter I've obtained two programs hoping that parts of them might be improved when more is learned. I haven't decided on hardware yet but in the meantime I purchased the "Chess Challenger" made by Fidelity Electronics at a good discount. It is supposed to beat an average player 25 to 75 percent of the time. After June they will add additional hardware for $75 to improve its game even more.

I can't be of any help to anyone already into this but maybe for someone just getting started I can offer some references on how to go about it. As we discussed one time, maybe if enough people worked on this problem someone will come up with a better approach to enable a computer to play better with less tree searching. As you know the "big" machines are examining 75 to 500-thousand positions per move and still not playing top chess. There has to be a better way so that hobby computers canplay an excellent game.

If anyone thinks I can be of any help or wants to trade information I will be glad to communicate with them.

Sincerely,

Russell McNiel
1343 La Manida
Carpinteria, Ca. 93013

Egbert Meissenburg

D209 Winsen/Luhe, July 2, 1977
Ilmerweg 70 a
Bundesrepublik Deutschland

Dear Mr. Penrod:

I just received issue ## 1 of your COMPUTER CHESS NEWSLETTER. It is a very good idea to edit a fully specialised journal for computer chess. I should like to receive all future issues of COMPUTER CHESS NEWS and all you publications on computer chess. I add $ 5.-- for subscription in banknote.

I miss your mentioning of Hayes/Levy Edinburgh 1976.

I offer you to compile a list of literature on computer chess from Russian sources as the material of Levy 1975 seems to be very incomplete.

Yours sincerely

Micro-Ware Ltd 27 FIRSTBROOKE ROAD, TORONTO, ONTARIO, CANADA. M4E 2L2.

Dear Doug,

Enclosed please find $3.00 for the first 4 issues or so of your newsletter. I look forward to reading some interesting articles and may contribute myself if time permits. I will be attending the World Computer Chess Championships and may find time to comment for you.

Your readers may be interested to know that MICROCHESS 2.0 is close to being completed. It will be initially available for the 6502, followed closely by a 6800 version, and then an 8080 version. It will offer a fairly sophisticated playing strategy as well as many of the frills that users of MICROCHESS have suggested in their letters to me. E.g. English notation, move validation, a set-up procedure, etc. The playing skill seems to be adequate to challenge the average club player. MICROCHESS 1.0 or the Chess Challenger are easily wiped by the program. I will be playing it against CHEKMO and COKO 3 in the near future and will let you know the results. If possible I will try to arrange other games with people (or computers) at the WCC in August.

Let me know if there is anything specific I can do to help you and your newsletter.

Sincerely,

Peter R. Jennings

SAMPLE GAMES PLAYED BY MICROCHESS

Human	MICROCHESS (Ruy Lopez)		MICROCHESS	(Queen's Indian)	Human
1. P-K4	P-K4	(1)	1. P-Q4	(1)	N-KB3
2. N-KB3	N-QB3	(1)	2. P-QB4	(1)	P-K3
3. B-N5	N-B3	(1)	3. N-KB3	(1)	P-QN3
4. O-O	NxP	(1)	4. P-KN3	(1)	B-N2
5. P-Q4	B-K2	(1)	5. B-N2	(1)	B-K2
6. Q-K2	N-Q3	(1)	6. O-O	(1)	O-O
7. BxN	NPxB	(1)	7. N-B3	(1)	N-K5
8. PxP	N-N2	(1)	8. Q-B2	(1)	NxN
9. N-B3	O-O	(1)	9. QxN	(1)	P-Q3
10. R-K1	B-N5	(37)	10. P-K4	(58)	BxP
11. B-Q2	Q-K2	(46)	11. R-K1	(55)	P-Q4
12. P-QR3	B-R4	(59)	12. B-B4	(65)	N-Q2
13. P-N4	B-N3	(47)	13. PxP	(87)	PxP
14. N-K4	P-Q4	(60)	14. Q-B6	(91)	N-KB3
15. PxP (e.p.)	PxP	(68)	15. N-N5	(122)	R-K1
16. P-B4	B-KB4	(65)	16. B-R3	(158)	B-N5
17. P-QB5	PxP	(82)	17. R-K2	(198)	R-K2
18. NxP	Q-B2	(121)	18. R-B1	(134)	Q-KB1
19. Q-K7	QxQ	(170)	19. R-R1	(139)	QR-K1
20. RxQ	BxN	(58)	20. R-B1	(119)	B-Q6
21. PxB	NxP	(45)	21. RxR	(187)	PxR
22. R-QB1	N-N6	(50)	22. R-Q1	(98)	R-K8+
23. RxBP	KR-Q1	(50)	23. RxR	(13)	BxR
24. R-B7	K-B1	(53)	24. BxP	(68)	P-N3
25. RxBP+	K-N1	(4)	25. QxN	(65)	B-Q7
26. RxB	P-KN3	(32)	26. N-KB3	(63)	Q-N5
27. R(5)-B7	R-Q6	(27)	27. Q-Q8+	(61)	K-N2
28. B-N4	R-Q8+	(25)	28. B-K5+	(60)	K-R3
29. N-K1	RxN+	(30)	29. Q-R4+	(60)	
30. BxR	R-K1	(14)	(mate)		
31. R-N7+	K-R1	(2)			
32. R-R7+	K-N1	(2)			
33. R(B7)-N7+	K-B1	(2)			
34. B-N4+	N-B5	(2)			
35. BxN+	R-K2	(2)			
36. BxR+	K-K1	(1)			
37. B-B5	P-R3	(3)			
38. R-R8+ (mate)					

(60) indicates the approximate time in seconds for the computer to make its move

The following is from a letter from Paul Burega, 1 Pleasant Bay, Winnipeg, Manitoba, Canada R2K 0C9:

I myself am in the process of writing a computer chess program. It is written in structured WATFIV FORTRAN and I am currently debugging what I have written on our local university's IBM 370-168. I currently have 600+ statements occupying about 30,000 bytes. At this point, all the program can do is input a human's move, check its legality, if it is a legal move, go through various update procedures and then generate all possible legal moves available for the computer. I am just starting to write the evaluation routine which will evaluate each move, give it a value, and then output the move with the highest value.

I hope that your newsletter will be a good one and I hope that I will be able to pick up some good pointers which I can use in my program,

CHARLES F. WILKES 6512 Rockhurst Road Washington, D.C. 20034

23 June 1977

Doug Penrod, editor
COMPUTER CHESS NEWSLETTER
1445 La Cima Road
Santa Barbara, CA 93101

Dear Doug:

Received your first newsletter, for which many thanks, and much encourgement for the future. I am enclosing $3.00 - please accept this in payment for the current issue, and as advance payment for the next three issues. I know you only asked for the next one issue in advance. But sending money one issue at a time is a pain. If in fact it turns out that there is no second, third or fourth issue, please accept the unearned balance as a donation for your investment in the first (and/or subsequent) issue(s).

I don't recall writing you - I write many people, so this is perhaps not as strange as it may sound. I also read Kilobaud, so may have, and then put it out of mind. It is also possible that you received my name from Monte Newborn, for a reason you will understand shortly.

The reason is that my son - Charlie - and I, are the joint authors of a computer chess program, which we named THE FOX. We competed with THE FOX in Atlanta at ACM three years ago, where we won against MIT in our first game, although by a time forfeit. You will find our game published in the SIGART bulletin of the time. We lost our second game to Newborn's OSTRICH by checkmate, both games on Sunday. We were playing on a timesharing system, which on Sunday was all ours. Unfortunately during gametime on Monday & Tuesday evenings, there were many other users of our system, and we lost both games by time forfeit. We think our game played very well, however, but have been unable to play in competition since due to the cost involved. We are continuing work on our program, and exercise it frequently.

We think our program is unique in many ways. For one thing, it is written in APL, a language which we feel is uniquely adapted for expression of the basic chess algorithm. For another thing, we wrote it entirely without knowledge of how others had to that point programmed. Due to our use of APL, we aren't even sure if it would have been any help had we known the details of other programs. We can recognize similiarities between our programs and descriptions of others. For example we use tree search, and our own equivalent of alpha-beta pruning.

For the record, I would like to detail how our game came to be. In the summer of 1972, my son and I both read George R.R.Martin's article in Analog magazine entitled THE COMPUTER IS A FISH. This concerned the earlier ACM computer vs. computer chess tournaments. My son was just entering the College of William and Mary in Williamsburg, Virginia, starting with the summer session. Since he had learned APL by going with me to my office on Saturdays over many years - I work for the IBM Corp. - Charlie already knew APL exceptionally well. William & Mary provide APL services to all students without charge. I challenged Charlie to write a Chess program, and that he did - using the College's machine. He came home Christmas with a crude but working version, which I promptly transferred to my own in-house IBM machine, which I was entitled to use.

Over the next few months, Charlie and I both worked on the program, and various people in my office played it. Since it won its very first game against the office chess expert, it got a lot of attention from our very active chess group of lunch-time players. By the time Charlie came home for the Spring break, we had an excellent program. Charlie had spent his time working on the guts of mid-game play. I had spent my time working on providing a sophisticated move acceptance routine, using descriptive notation, board display, etc., and in implimenting a book opening routine. Charlie had developed what we term a "programmed opening" as well, which we enter following departure from the book when a move is not found, and stay in until a threat of a certain level forces us to mid-game play. We stay in mid-game until the end, having no special end-game play (we would like to some day).

We signed up to compete at ACM at the end of the summer - 1973, representing the College of William & Mary, with the specific approval of its President. When Charlie came home for the summer, we both worked many many late late nights, all weekends, polishing the program. We obtained the assistance of a local chess master - Allan Savage, who was of great assistance. Through him we introduced many excellent strategies - i.e. Pawn strategies, etc., which greatly help the game. He played it the final game before the tournament, over a period of three nights. It was 62 moves to a checkmate for Savage of course, but he said the first 60 moves were a dead tie, until he passed a pawn. He then mated in two moves. Savage said he could not rate the program, but in a human would estimate between 1200 to 1600. The problem is that certain moves when made by a human, would indicate that a certain level of chess knowledge was present, which would apply in other situations as well. In the computer, however, "knowledge" is uneven. While it may make brilliant moves for a long sequence, it may then make an unforgiveable "dumb" move, which no human capable of making the good sequence would ever do. I think this is typical of most computer chess, except for the very very best.

There are many copies of THE FOX playing on IBM in-house APL systems, and periodically I get game print-outs which players want to show me. Our problem with chess in APL, is the tremendous demands made upon the host computer by the program. We played at ACM on a 370/145, using microcoded APL. When our program was in progress, we literally stopped the machine - we were in a 100% CPU bound execution mode, and of course could have used lots more speed. Even today, few machines execute APL any faster - I should know, since I am the individual within IBM who provides technical support for APL nation wide, within the data processing division of IBM. I am looking forward to faster machines - the 370/148 with microcode, and the 3033 when it is eventually available. I have a SPEED function which I use to measure machines for playing THE FOX, in order that I know it is capable of properly computing fast enough to make play reasonable. For example, I also have access to an IBM 5100 with APL, but this is far too slow to do anything but play through the book and programmed openings, coming to a complete halt with mid-game play.

Just a few details of THE FOX. We have a setting where we can alter the depth and width of the search. Our tournament level setting was as follows: Regardless of the setting, we make all possible moves for the first two half-moves - approximately 900 board positions. We then evaluate, and prune to the best three apparent moves, and make three more half-moves, for a total depth of five half moves. If we discover loss of material of major value, we may go wider than three, at the second half-move level. We are looking for an indication that each succesively position at the second half-move level (of the initial three)

generates a progressively worse evaluation at the fifth half-move. If it does, we make the assumption that an even wider search would not change this result. If we cannot prove this, we may go wider provided that our internal clock vs. move number indicates we have time to spare.

Obviously time is everything in a tournamet. We gain time points by our fast book at the opening, and our continuation of opening strategy via our programmed opening when we have to leave the book. We can move most mid-game moves in our allotted three minutes, except when we get in trouble - forcasting loss of material. Since we expect this to happen from time to time, we need the saved time from the opening moves in the bank to protect us much later in the mid-game, especially at the time call points.

Incidentally THE FOX can play either color, does play en passant of course - both allow & initiating such moves, and permits pawn promotion to any major piece value. It itself, however, will always promote to a Queen - the logic to test whether a lessor piece would be better was too much for the moment. The game can be started with any board setup. It can, in fact, play itself very nicely, flipping colors between moves. The capability of starting from a setup, and then play both colors, is most interesting to use in testing quality of moves from a given board position. All programs should have this ability for development purposes.

THE FOX was implemented in an APL workspace of 64K bytes. Today with much larger workspaces, this of course seems very small - we did struggle to keep it within this size, believe me. For example, the entire book is in the workspace initially, but only the moves for the one color to be played by the computer. Allan Savage personally selected the classical openings to be played by the computer for each color, omitting those openings which do not in the long run favor the player of the color being played by THE FOX. When the mid-game is entered, all code applicable to the opening is expunged, to provide room in memory for the expansion of the look-ahead three. The dynamic workspace storage management of APL is invaluable in this respect.

At my work at IBM these days, we have at least two fellow employees, who possess KIM-1 computers, exclusively for CHESS, using the available program. Of course they modify and improve the code as possible. The KIM-1 of course cannot begin to compete with THE FOX. I do think that the micro-computer is the future of computer chess however. Today I consider the Zilog and Mostek Z-80 to be the most appropriate CPU for the job - the availability of the bit instructions on the Z-80 are essential in my opinion. I know that sooner or later APL will also be available on the micro-computer. I don't know, however, if it will have the speed necessary to play satisfactory chess. I think there are strategies where multiple computers could work in consort to develop the move, and this may be the best approach.

In terms of development for THE FOX, we have one main strategy which we want to work on. The simple idea behind it really gripes us that we did not think of it earlier. This is to allow the computer to continue to "think" during the time the other computer is working out its move. Today we predict what we consider the best move for the other side to make is, as a function of generating our own move. We think we can take it, and allow the computer to make the assumption that this is the other sides move. If it is, we will signal through its entry at the proper

time that all the work to this point can be retained, and computation continue. If it is not, nothing is lost - merely return to the board position, make the actual move selected, and compute as we have done in the past.

My son Charlie has now graduated from William and Mary a year ago, and now has completed his first year as a medical student at the Univ. of Maryland school of medicine in Baltimore. He is deep at work in developing APL programs of interest in the field of medicine. I am certain that the development as an APL programmer which occurred with him in his work on THE FOX was so great, that he has great promise in this field in years to come. I know I am a much superior programmer today for the long hours of work on THE FOX - it taught me a great deal, particularly in how to write APL code for maximum performance on the computer, using minimum resources.

Thank you again for your efforts in behalf of Computer Chess. If you have the space in a future newsletter, you have my permission to use this letter, in the hopes that it may inspire some new developing author, as did George R.R.Martin's article in Analog for us.

Sincerely,

Charles F. Wilkes

IRA D. BAXTER
SOFTWARE DYNAMICS
17914 S. LAURELBROOK PLACE
CERRITOS, CALIF. 90701

DOUG PENROD
1445 LA CIMA ROAD
SANTA BARBARA, CAL. 93101

Sir:

This is to inform you that Software Dynamics has a Chess program written entirely in SD Compiler BASIC on a 6800 microcomputer, using a minimax tree lookahead scheme.

For BASIC, the program runs fairly quickly (blitz games are about 3 seconds a move). The program has a selectable lookahead minimum and maximum so that it can play from idiot-level chess to just plain mediocre (2 to 15 minutes a move). Lack of machine time prevents good to excellent play, but it is much better than all the other BASIC Chess games we have seen.

We would be happy to pit our 6800 running SD Chess against any other micro BASIC version of Chess.

If you have any questions, please call me at (213) 926-6492.

NORTHWESTERN UNIVERSITY
EVANSTON, ILLINOIS 60201

CRESAP LABORATORY OF
NEUROSCIENCE AND BEHAVIOR
2021 SHERIDAN ROAD
EVANSTON, ILLINOIS 60201

DEPARTMENT OF PSYCHOLOGY
312-492-7406

Mr. Douglas L. Penrod
1445 LaCima Road
Santa Barbara, California 93101

Dear Mr. Penrod:

Thank you for sending me the first issue of your newsletter. I hope your new venture is a successful one. I am enclosing a check for $3 for the first four issues. I have several observations on chess programming that may be of interest to your readers.

Past experience indicates that programs which play reasonable chess always involve a look-ahead search. In addition, they all have some scheme for assessing whether a terminal position is quiet or turbulent. If the position is not static, the search is continued. Programs which do not incorporate this strategy generally play a very poor game.

A reasonably comprehensive look-ahead search requires many, many CPU operations as well as considerable memory. It follows that a successful program should be written in assembly language to enhance the speed of execution or at least in a higher-level language for which a good complier is available such as Fortran. Writing in a language which uses an interpreter will generally increase the execution time for a deep search beyond ones patience or resources. A second implication of this line of reasoning is that individuals who wish to write a chess program on a personal computer should make every effort to develop extremely efficient code and should anticipate a need for at least 16K of RAM.

A second consideration is that much can be learned by spending a few hours in the library. There are certain references which are an absolute must. They include:

Shannon, C.E. Programming a computer for playing chess. Philosophical Magazine, 1950, 41, 256-275.

Berliner, H. Chess as problem solving: The development of a tactics analyzer. Unpublished doctoral thesis, Carnegie-Mellon University, 1974.

Knuth, D.E. and Moore, R. An analysis of alpha-beta pruning. Artificial Intelligence, 1975, 6, 293-326.

Slate, D.J. and Atkin, L.R. Chess 4.5 - The Northwestern University chess program. In Frey (ed.), Chess Skill in Man and Machine. New York: Springer-Verlag, 1977, pgs. 82-118.

The odds are about 100 to 1 that anyone who is writing a chess program and who has not read all of these references is probably wasting most of their time. A little reading can go a long, long way.

For those who like short cuts, a short Fortran program which incorporates a tree search was written several years ago by Jim Gillogly (Information Sciences Dept., The Rand Corp., 1700 Main St., Santa Monica, Calif. 90406). This program could probably be modified for home-brew systems. Another possibility for the game-playing enthusiast is to start with a more tractable game. An excellent starting point for an 8 or 16 bit CPU is the game of Hawaiian checkers (Konane). For a description of the rules, see an article (p. 5) by Joel Gyllenskog in the Feb., 1976 issue of the Sigart Newsletter (published by ACM).

Best regards,

Peter W. Frey

Peter W. Frey
Associate Professor

Dear Mr. Penrod,

Thank you for the copy of your newsletter. Enclosed is a small monetary contribution. I hope there is enough sustained interest to keep your publication going. When I opened it up, I wasn't sure what I'd find. I certainly didn't expect a letter from Bobby Fischer! That was quite interesting. I would like to have seen his games with the Greenblatt program.

There is a promising future for chess programs on micro-processors. Right now I think the field is held back by a lack of experience and know-how. People are still struggling with basic problems, like how to generate moves and grow trees, etc. I have had several inquires from computer-chess enthusiasts who would like to write a program but have little idea of where to start. Prospective chess-programmers should read the existing literature so they don't have to re-invent the wheel. This doesn't mean that they should try to emulate existing programs. There is room for lots of improvements and new ideas in chess programs for both little and big machines. To micro-processor chess programmers I also suggest :
1. That they equip their machine with a fair amount of memory, like 32k.
2. That they write modular, well-documented code.
3. That they carefully think out basic design issues, like the internal representation of moves, pieces, boards, trees, etc.

Some interesting recent games of Northwestern University's CHESS 4.5, which I co-authored with Larry Atkin, can be found in April, 1977 Sigart Newsletter, June, 1977 Chess Life and Review, and June, 1977 Scientific American.

Sincerely yours,

David J. Slate

David J. Slate

Stephen Stuart
2215 Rock St, Apt. 12
Mountain View, CA 94043

Dear Mr. Penrod:

I'd like to see a copy of your computer chess newsletter. Enclosed is my check for 75¢.

It might be of interest to you or your readers that there is some activity in this field where I work. An engineer at Amdahl Corp. (not myself) has put forward a challenge to the Amdahl Computer Club for microprocessor chess programs. The ground rules are only that the programs run in 2K (8 bits per word, presumably) or less, as the rumors say that the Fidelity Chess Challenger uses 2.5K.

So far I am the only person in the club with a working program, running on a Signetics 2650. The program (Revision 1) plays a little more poorly than the Chess Challenger, but I am currently at work on Revision 2, having learned a lot from my first shot at it. My club competitors are now only a few weeks away from their respective completions in some cases.

I brought my "toy" into work for a demonstration, which incidentally helped fire up interest in the tournament. The 2650 was soundly beaten by 90% of its human opponents over a 2-day period. It looks like once we can get a machine-vs.-machine game going, we will be seeing some low-level but interesting chess.

If you would like to hear more of this as it progresses, I would be glad to report on it.

This is from a letter from Arnold E. Jones, 2561 Alafaya Trail Apt. #118, Orlando, Florida 32807:

I am interested in your Chess special interest group, which you talked about in your letter published in Myte's Sept. issue. I think the idea of the Chess newsletter is a great idea. The Chess programs on the market today are being sold at high prices for the hobbyist.

I have the source to a good chess program in IBM 360 assembly language which I recently obtained. I would however like to get some source listings of a chess program in 8080 or Z-80 assembly. I have access to Zilog Development System, IBM 370/165, Varian 73, and a SOL. I am a freshman at Florida Tech. Univ. which has all the above Systems. I am planning to buy my own micro soon. I would appreciate any information you could send, and I can send a copy of the chess program to anybody who wants it.

John Griffin
1067 Enderby Way
Sunnyvale, CA 94087

Doug,

I received Issue #1 of the Newsletter recently. It looks like a good start, and I am looking forward to future issues.

I have been assembling notes and ideas for a computer chess program for a number of years now, and I am about to start coding. I have access to a Data General Eclipse and hope to achieve a system of some skill and sophistication. It is for this reason that I hope the newsletter will not be entirely devoted to chess on microcomputers.

Enclosed is a check for six more issues. Also enclosed is a copy of some material I recently received from USCF.

UNITED STATES CHESS FEDERATION
PUBLISHERS OF CHESS LIFE & REVIEW
186 ROUTE 9W • NEW WINDSOR, NEW YORK 12550

Dear Mr. Griffin,

Thank you for your letter of May 30th inquiring about computer programs in the USCF Rating Program. Enclosed you will find sheet of information which should help you greatly in pursuing your interest in this subject.

You'll be happy to know that our USCF rated computers are published right along with the humans. We list them as though the last name is Computer and the first name is the true name. Below is a partial list of the ratings you've requested. Most of them are provisional (that is, have played less than 25 games).

Computer, Blitz	1518
Computer, 4.5	1935
Computer, Dutchess	1351
Computer, Patzer 451	1006
Computer, Sneaky Pete	1209
Computer, Worcester	1295
Computer, Xenabor	1244
Computer, Zap	1057

HEATH
Schlumberger
HEATH COMPANY
BENTON HARBOR, MICHIGAN 49022
PHONE: 616-982-3344 TLX 72-9421

Dear Mr. Penrod:

I am enclosing a copy of our computer catalog as you requested. I hope you like what you see. And I hope these products live up to your expectations.

As for software, it comes with the computers only. It is not available separately to non-Heathkit computer owners. We do plan to offer applications software in the near future. We would like to do a chess program, but do not presently have the in-house expertise to do it. I'd like to find a program we could adapt to our machines and resell. Any suggestions?

Thanks for your interest in our products.

Sincerely,

Louis E. Frenzel (pr)

Louis E. Frenzel
Director, Computer Products

Thomas A. Fallis
P.O. Box 76242
Sandy Springs, GA 30328

Dear Mr. Penrod,

When I received the first issue of the Computer Chess Newsletter, I was surprised to learn how many other people were also interested in Computer Chess. While I do not own a "hobby" computer, I have been working on a design which should be easily adapted to micros and minis. My current project is to convert all known master games from the early nineteenth century to the present into a database. Once this is done, it will be available to anyone interested in analyzing actual play with computer generated play.

Enclosed is my check for $2.25 to cover the cost of the first three issues. I really enjoyed the first issue and hope the next issues will be even better. I would be interested in corresponding with other people with similar interests. I am currently using an IBM 360/30 with 128K bytes for most of my work. It would be beneficial to have the Newsletter provide a service to allow its readers to correspond with those who share or want to explore similar paths.

David Levy
104, Hamilton Terrace,
London, NW8 9UP
England

Dear Mr. Penrod,

Thank you for sending me the first copy of your newsletter. I am very pleased that at last someone has had the courage to start something like this.

I would very much like to receive the newsletter by airmail, and would be willing to write occasional articles, e.g. on the 2nd World Computer Championship in Toronto (August 7-10) and the ACM tournament in Seattle (October 17-19).

Whether or not you can find some individual or university to take over the running of the newsletter will probably depend on how many subscribers you get and how much people are willing to pay. So long as the income is sufficient to pay for mailing, printing and typing I would think that it should not be too difficult to find a permanent editor.

As to the matter of content, I think that it should be possible to persuade chess programmers to sent you short technical papers from time to time. Hopefully some of those who are on your list will do so without any special request from you. If you like I can try to get some support from the people who participate in Toronto and Seattle. Will you be able to attend either or both of these meetings?

Can we see the Fischer games in your next number?

This is from Donald S. Tork (WB6YJY), 3484 Hill Canyon Ave, Thousand Oaks, CA 91360:

Please add me to your list of computer fanatics. I have an Altair 8800 & disk and I'm in the process of writing an operating system. (I'm a programmer in real life). I have a USCF rating of 1457 and I'm interested in seeing hi-level designs for chess programs; especially the ones that participate in the ACM tourney. (If you can persuade the authors to write them)

The following is from a letter from Kevin McLoughlin, Caltech 1-58, Pasadena, CA 91126:

Howdy Computer Chess Newsletter

I just finished typing in, translating, and debugging Randy Miller's Altair Basic chess program for use on the PDP-10 system here, and have a number of comments, expletives deleted:

1) I was almost able to type it in translating as I went, but there was one section of code, 25 lines in 1-increment statement numbers, packed with multi-statement lines, which could only be rewritten from scratch. Our BASIC is bare-bones ANSI standard and doesn't permit such things as multi-statement lines, logical AND, OR, and NOT, ON-GOSUB, etc., with which Miller's program abounds. He should have at least spaced out his statement numbers a bit more, to allow for changes.

2) The programming itself is redundant and incompetent. Comment lines are absent where they are most needed; an entire subroutine is repeated statement for statement; blocks of code are repeated with only small, computable variations; the move generation and legality check routines are clumsy and slow. There is a statement inserted which bypasses the entire evaluation routine, which you are apparently supposed to delete if you want the program to play decent (non-random) chess. The board display routine must be intended for a 9200 baud line printer--it would take until next year to print the board on a TTY. Miller could have saved the core space or used it for a routine to accept P/K2-K4 type notation.

3) Once debugged, the program does indeed play chess, though it is a bad joke to have included a statement to make it easier. With this removed, the program plays at the level of a seven-year old who learned the game last month. Enclosed is a listing of a game it played against the Greenblatt chess program, which resides (protected against examination to the nth degree) on the system library here. As you can see, the Greenblatt program won in 15 moves, but the Miller program put up a good fight; its moves were mostly consistent, considering the one-ply lookahead. The evaluation routine could be rewritten to make better play possible.

4) A listing of the program, rewritten in ANSI standard BASIC (so it should run on most machines), renumbered, commented, relieved of some of its redundancies, is enclosed. You probably can't publish it because of copyright rules, but ask Randy Miller (or MITS, or whoever owns it) anyway; perhaps they will let you print it in a future issue. In this form it is a basic, usable, understandable, modifiable chess program, and might be of use to your readers.

5) The one thing the distributors of this program did right was to include with the program a report explaining the steps involved in writing a chess program, in general high-level language terms. With the problem defined and set before me, I intend as soon as I have the time and energy to write my own program. It will be written in PASCAL, a highly structured language, whose best feature is perhaps its near total lack of need for a GO TO statement (though it has one, for those who have not yet learned how to think). Once I have it working in basic form, I will translate it into BASIC and send you a copy (a process considerably easier than translating BASIC into PASCAL). Grunging through the Miller program did two good things for me; it showed me the advantages of good programming style, and it showed me, laid out step-by-step, exactly what is needed (and not needed) in a chess program.

The following is from North Star Chess, edited by Paul D. Shannon, Box 371, Osseo, Minnesota 55369. This was from Volume II, #II.

Computors in Chess
Editorial Opinion by Paul Shannon

Computors do not yet deserve to play tournament chess. Sportsmanship and fighting spirit simply do not compute. Playing a computor appears to be like a boxer fighting a wrestler; the aim is the same (to put the opponent down on the canvas) but the style is very different, and the result is novelty not true sport.

I was playing in the Minnesota Open when Chess 4.5 scored its remarkable 5-1 triumph. Most of its opponents 4 of 6 were formerly high rated competitors but had not played much for the past several years. Ronning and Fenner had not played at all for at least three years, while Stenberg and Armagost have played only once or twice each year recently. Quite frankly I think that just about any good A class player could have had a master performance against that group on that weekend.

One week later I was tournament director for the Minnesota Championship playoff. All of these players (1) were in practice, and (2) had prepared to face the machine. As a result, although the average rating of the opponents was about the same, the program scored only one win and one draw. The difference had to be one of the two factors above or more probably a combination.

Tournament promotors cannot do much about keeping their players in top form except to hold many events and hope that everyone will play in them. However, we can do something about the second factor, the computor's advantage of surprise.

When a player prepares to play in a Swiss System event, he anticipates human opponents with human weaknesses. Specifically, a good tactical player can prepare to force complicated variations and thus cause errors. Against a computor this same tactical player feels the opponent has "nerves of steel" and "x-ray eyes" which make it a true "superman" due to the ability to "see" all of the variations and never get frustrated. Thus preparation for humans, or practice against them will not help much against the machine.

Having watched several of the games of Chess 4.5, it seems that one must avoid tactics and play variations where the issue is in using superior planning to create long term weaknesses, which may then be exploited by further long term strategies and maneuvers. Since this is contrary to the wildly tactical style which I cultivate for 30/1 Swiss play, I'm not sure I could adapt to a positional style or the typical ten minute pairing notice of a weekend Swiss.

For this reason I feel it is unfair to play computors in Swiss System events except perhaps in a one round per day event (club tourney or the U.S. Open) where significant notice (several hours at least) can be given to the computor's opponents.

Another problem area for all concerned when a computor plays is finishing the game. Chess 4.5 was unable to resign or to accept a draw offer or offer one itself. Thus theoretically all games must be played on to mate or perhaps the 50 move rule, both of which it is good enough to avoid for hours. In the Minnesota Championship we had to resort to adjudicated draw in bishops-of-opposite-color ending. Elsewhere the operators have used their human judgement to make the decision, but neither procedure is good. It is not fair for a player to have to

continue and thus lose rest time before the next round when the result is obvious to all present.

The final question for this article is: Should computor Chess 4.5 have been allowed to play for a state championship title? At the risk of being accused of advocating discrimination against all machine intelligance, I have to disagree with that decision of our worth (?) state board. The publicity the event received had some small value as did the novelty of seeing it, but not enough to offset the cheapening a great sporting event, the Minnesota Championship. There is no way computor capabilities can be compared with human players except perhaps with a human of total recall, no fatigue, interchangeable brain cells and no emotions. It is not the mechanical moves which make chess great, it is the sport of mental battle with another human. The human spirit and sporting values are intangibles which no machine may ever comprehend.

Hopefully, various local and national groups will soon devise some specific policy statements on these issues. The very minimum statements should include (1) adequate notice of the pairing to the opponent (or the right to refuse the pairing!), (2) an agreement on a workable system of evaluating draw offers and a program for resignation by the machine when warranted, and (3) restriction on participation where titles or large cash prizes are at stake.

Of course this is just one opinion and we would be happy to print other comments or opposing viewpoints. Adress all correspondence to Editor, North Star Chess, Box 371, Osseo, MN 55369.

Robert Semko
8188 Sherwood Place
Riverside, Calif. 92504

Dear Doug,

Thanks for sending me a copy of your newsletter. I'm enclosing 2 dollars for the first issue and the second issue. Put me down for a subscription when they're available.

I need some advice. I'm a beginner with no hardware as of yet. I'm planning to buy an assembled system by the end of the year and I'd like to keep the price at about $1500. The programming will mainly be games, especially chess. I'd like to find out about graphics, where I can display the board and pieces directly on the CRT. I know a good chess program requires a lot of memory so I'd want the kind of hardware where I can add more memory as I can it.

I'm a quadriplegic, paralyzed from the neck down, and although I can handle a keyboard, any kind of kits are out of the question. If you have any ideas as to what kind of computer system will be the best for me, I'd appreciate hearing from you.

From Jonathon Steer, 21 Berkeley St., Nashua, NH 03060:

I was interested to read your letter in the June KILOBAUD as chess computing has been a hobby of mine for several years. I got interested in computers through chess and chess programming. As a USCF member, having a good program around to play against is great. Having helped write a program running on the Univ. of New Hampshire DEC-10, I appreciate all the information one can get. The present version, however, only plays in the 1000-1100 level with openings being the strongpoint.

I hope that this newsletter works out because up until now as you know only academic sources for help and information.

© 1977 Bobby Fischer

Greenblatt-White
Fischer-Black

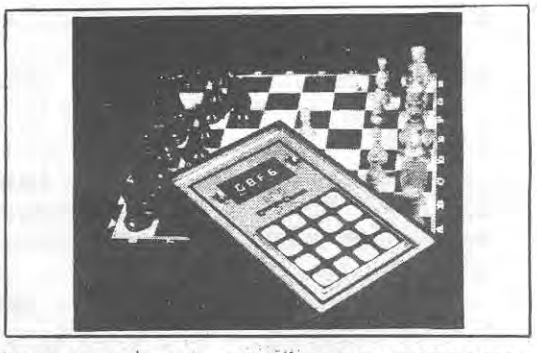
1.P-K4	P-QB4	21.QxQRP	N-K6
2.N-KB3	P-Q3	22.BxP	Q-N4
3.P-Q4	PxP	23.P-KN3	R-R1
4.NxP	N-KB3	24.B-R7	P-R4
5.N-QB3	P-QR3	25.Q-N7	P-R5
6.B-K2	P-K4	26.K-B2	PxP
7.N-QN3	B-K2	27.PxP	P-B4
8.B-K3	O-O	28.PxP	RxPch
9.Q-Q3	B-K3	29.K-K1	QR-KB1
10.O-O	QN-Q2	30.K-Q2	N-B5dbl ch
11.N-Q5	R-B1	31.K-B2	Q-N3
12.NxBch	QxN	32.Q-K4	N-Q3
13.P-KB3	P-Q4	33.Q-B6	R-B7
14.N-Q2	Q-N5	34.K-Q1	B-N5
15.N-QN3	PxP	35.BxR	Q-Q6ch
16.Q-Q1	N-Q4	36.K-B1	BxB
17.B-QR7	P-QN3	37.N-Q2	RxB
18.P-QB3	Q-K2	38.QxN/Q7	R-B8ch
19.PxP	N-K6	39.NxR	Q-Q8 Mate
20.Q-Q3	NxR		

Fischer-White
Greenblatt-Black

1.P-K4	P-K4
2.P-KB4	PxP
3.B-QB4	P-Q4
4.BxP	N-KB3
5.N-QB3	B-QN5
6.N-B3	O-O
7.O-O	NxB
8.NxN	B-Q3
9.P-Q4	P-KN4
10.NxKNP	QxN
11.P-K5	B-KR6
12.R-B2	BxKP
13.PxB	P-QB3
14.BxP	Q-N2
15.N-B6ch	K-R1
16.Q-R5	R-Q1
17.QxB	N-QR3
18.R-B3	Q-N3
19.R-QB1	K-N2
20.R-N3	R-KR1
21.Q-R6 MATE	

Greenblatt-White
Fischer-Black

1.P-K4	P-QB4	25.BxR	BxB
2.N-KB3	P-KN3	26.R-KB1	B-B4
3.P-Q4	B-N2	27.R-B2	P-KR4
4.N-QB3	PxP	28.R-K2	K-B2
5.NxP	N-QB3	29.R-K3	B-Q5
6.B-K3	N-KB3	30.R-KB3	K-K3
7.NxN	NPxN	31.P-B3	B-K4
8.P-K5	N-N1	32.R-K3	P-Q5
9.P-KB4	P-KB3	33.PxP	PxP
10.PxP	NxP	34.R-K1	P-Q6
11.B-QB4	P-Q4	35.P-KR4	P-Q7
12.B-K2	R-QN1	36.R-Q1	B-B6
13.P-QN3	N-N5	37.K-B2	B-KN5
14.B-Q4	P-K4	38.R-KR1	B-Q5ch
15.PxP	O-O	39.K-N2	K-Q4
16.BxN	Q-R5ch	40.P-R3	K-K5
17.P-KN3	QxB	41.R-KB1	K-Q6
18.QxQ	BxQ	42.K-R2	K-K7
19.R-KB1	RxR	43.K-N2	B-R6ch
20.KxR	P-QB4	44.KxB	KxR
21.B-B2	BxP	45.P-QN4	P-Q8(Q)
22.B-K1	R-KB1ch	46.K-R2	Q-K7ch
23.K-N2	R-B6	47.K-R3	Q-KN7 MATE
24.P-KR3	RxN		



COMPUCHESS I IS A MICROPROCESSOR BASED chess opponent with six selectable levels of intelligence, randomized strategy, programmable game situation capability, automatic or selectable pawn promotion, castling, and en pessant. Levels 1 and 2 are excellent teaching levels for children while 3 and higher levels are for practiced players. Level 5, for instance, will solve the most complicated of mate-in-two problems. Stock to 4 weeks at $159.95 from CompuChess, P.O. Box 65, Largo, Florida 33540.

The above appeared in a recent issue of Electronic Engineering Times

A number of Issue #1 were returned because the addressee had moved. If you know the current address of any, please let me know:

William B. Adams	Hyattsville, MD
Miki Alexander	Sperry-Univac, Cinnaminson, NJ
Luis Ayala	IBM, Mexico City
Jeffrey M. Bachman	Milwaukee, WI
Craig Barnes	UC, Berkeley, CA
David Barton	UC, Berkeley, CA
Steven Bellovin	Columbia U, New York City, NY
Victor Berman	Sperry-Univac, Cinnaminson, NJ
Paul E. Black	Cedar City, UT
W.W. Bledsoe	AI Lab, Cambridge, MA
Burton H. Bloom	Lexington, MA
Ted Brown	Boston, MA
Capt. Franklin Ceruti	Alexandria, VA
Anton Chernoff	Philadelphia, PA
Ben Cohen	Annapolis, MD
Robert W. Enden, Jr.	Omaha, NB
Charles H. Fisher	Highland Falls, NY
Jack Fox	Austin, TX
Lawrence Futrell	Georgia Tech, Atlanta, GA
John Gaydos	Moorestown, NJ
arry Goldstein	Belmont, MA
Stan Kugell	AI Lab, Cambridge, MA
Daniel S. Marcus	Flushing, NY
James P. O'Donnell	Forest Park, IL
C. J. Orton	U. of Ottawa, Ottawa, Ontario
Walter Perkins	Danville, CA
Robert Polansky	Boston, MA
Capt. Herbert Raymond	San Diego, CA
Prof. Helmut Richter	West Germany
Peter Rowe	UC, Berkeley, CA
Ira Ruben	Sperry-Univac, Cinnaminson, NJ
Aaron Samson	Chicago, IL
Robert Schuman	Winston-Salem, NC
Capt. Rolf Smith	Richard-Gebaur AFB, MS
Joel Stutman	Brooklyn, NY
William Toikka	Sperry-Univac, Cinnaminson, NJ
Bob Uliss	Winthrop, MA
John P. Walsh	Salem, NH
Albert Waltner	Huntsville, AL
David W. Zacharias	Fort Lauderdale, FL

As most of you have heard by now, Chess 4.6, by David Slate and Larry Atkin, running on the fast Cyber 176, won the World Computer Chess Tournament in Toronto last month. And although it was not matched against the former champion, Kaissa, by Michail Donskly and Vladimir Arlazarov of Moscow, an exhibition match was arranged, which Chess 4.6 won. I hope to print more about it next time, with an article by someone who attended the affair.

This issue is being mailed Third Class, except for overseas, which is being sent as Printed Matter by Air. The cost of air postage overseas results in a higher charge for the overseas subscribers.

You can tell how many more issues you will receive by the total amount which you have sent in thus far:

I expect to put out Issue #3 as soon as I am finished with this issue, so I'll be free to attend the ACM meeting if I'm otherwise able to.

$

-- Doug Penrod, editor

David Bryant
4371 Rigel Ave.
Lompoc, CA 93436

Hi Doug.

Sorry it took me so long to write, but between Digital Group and high school I'm kept pretty busy.

The chess playing program we have at school is written in PDP-8 assembly (I think he means machine code--ed.) by John Youngquist. It takes about a minute a move, but plays very good chess - (only a handful of students here at the high school can beat it!) I will have it play itself for a while and get you a copy of some of its interesting games.

I'm still trying to get that Basic chess program working on our PDP-8s. It looks like it will work on one of our compiler Basics except for two minor problems: This Basic doesn't know anything about line numbers bigger than 4095 or two-letter variables. I'm working on some Basic programs to re-sequence and convert multiple letter variables to regular Basic variables (i.e. XY to X1).

I'd like to compare the Basic and assembly chess programs, for both speed and strength, but I'm sure that assembly is the only way to make a good and fast chess program, at least on micros and minis.

I'd love to write a chess program myself, but I doubt if I'll have time for quite a while. I'll be writing Digital Group's disk monitor soon, and this summer I may write an RPG for the 8080. However, I might rewrite John Youngquist's PDP-8 assembly program for the 8080.

Excerpts from 2 letters from Daniel Grieser, 4326 Kenny Road, Columbus, Ohio 43220:

I have just gotten a Microchess (8080 version) up and running on my Z-80/Digital Group system. I'm planning to pit it against an HP-21MX minicomputer (at work) running an early version of MAC/6. .....

Our first computer vs. computer chess contest:

WHITE: HP-21MX minicomputer running Minitech (in Algol)
BLACK: Digital Group (Z-80) running 8080 Microchess
Result: Draw; White=47 minutes, Black=32 minutes

Doug Penrod, editor
COMPUTER CHESS NEWSLETTER
1445 La Cima Road
Santa Barbara
California 93101

P 3.00 B 1.50
PETER R. JENNINGS
MICRØ WARE LTD.
27 FIRSTBRØØKE RØAD
TØRØNTØ, ØNTARIØ M4E 2L2
CANADA

All Rights Reserved © Copyright 1977 by
Douglas L. Penrod

THIRD CLASS MAIL