

SP *a professional paper*

---

UMPIRE: AN AUTOMATIC KRIEGSSPIEL REFEREE  
FOR A TIME-SHARED COMPUTER

by

John F. Burger

17 February 1967

SYSTEM

DEVELOPMENT

CORPORATION

2500 COLORADO AVE.

SANTA MONICA

CALIFORNIA

90406

---





17 February 1967

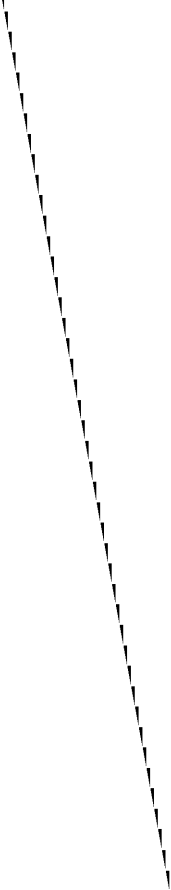
-1-  
(page 2 blank)

SP-2752

ABSTRACT

This paper describes a computer program that performs all of the functions of referee for the game of Kriegsspiel, a variation on the game of chess. The rules of the game are explained, users' instructions are given, and a description of the program is provided. While not a game-playing or heuristic program in itself, the program contains all the necessary routines for the basic representation of a dynamic chess game, and may thus be used as the core upon which could be built the heuristics for an actual chess-playing program.

47



17 February 1967

-3-  
(page 4 blank)

SP-2752

ACKNOWLEDGEMENTS

I am grateful to George Baylor for his paper, which laid out much of the procedure for automatically dealing with chess-board position and move control; to Stan Kameny and Clark Weissman, of the Research and Technology Division, for their help in working out some of the input-output problems in LISp; to Luke Krebs, formerly of the Time-Sharing Implementation Staff, for his patient explanations of the TSS JOIN command; and to many of the SDC Kriegsspielers, for their comments and help in testing the final versions.



INTRODUCTION

Kriegsspiel (German for "War Game") is a popular variation on the game of chess. To the knowledge of this author, no official rules for Kriegsspiel have ever been set down. Because the game is essentially well defined, however, this lack has resulted in only minor variations in the rules as observed in different locations. Two opponents play a game of chess in such a way as to prevent each from seeing the position of the other's pieces and pawns. Since, under the rules of chess, the position of one's opponent often determines the legality of a move (one cannot move into check, for example), a third person, called the referee, is present in Kriegsspiel to prevent illegal moves and to announce to both players such events as the capture of pieces or pawns.

This paper describes a computer program called UMPIRE that performs all the functions of an actual live referee for the game of Kriegsspiel. The program is written in LISP 1.5 for the System Development Corporation Q-32 Time-Sharing System. With slight modifications to the I/O functions, UMPIRE could be made operable on any computer having a LISP 1.5 compiler. The basic functions of the program are translatable into LISP 2.

UMPIRE, while not a game-playing or heuristic program in itself, contains all the programming for the basic representation of a dynamic chess game, and may thus be used as the core upon which could be built the heuristics for an actual chess-playing program. The extensibility of LISP allows for ease in making programming changes, and the list processing structure of the language provides efficient and easily accessible storage of the inhomogeneous data.

DESCRIPTION OF THE GAME

As normally played, Kriegsspiel uses standard chess rules, two players, a referee and kibitzers. Each player has a complete chess set (a board with black and white men). The players may not see each other's boards or men; the referee is seated so as to see both boards simultaneously. At SDC a vertical divider is usually placed between the two boards, with the players facing this partition on either side, and the referee seated at one edge of the divider. The following rules are quoted from an informal RAND Corporation memorandum on Kriegsspiel [1]:

- "5. A referee monitors the game (preferably using a third chess set, which neither player may see), and announces
  - 5.1 whose turn it is to move,
  - 5.2 on which squares the mover's pawns have currently valid options ('tries') to make captures,
  - 5.3 each rebuff ('no') experienced by the mover in attempting to move,
  - 5.4 the fact that a capture has taken place, where it happened, and the category of the man captured (according to the dichotomy 'pawn' or 'piece'),
  - 5.5 checks, which are announced by whichever of the following is (are) correct:
    - 5.5.1 check on a long diagonal,
    - 5.5.2 check on a short diagonal,
    - 5.5.3 check on a rank (or 'horizontal'),
    - 5.5.4 check on a file (or 'vertical'),



- 5.5.5 check by a knight,  
(The diagonals considered are the pair which intersect at the king.)
- 5.6 the fact that a pawn promotes (but not the piece promoted to or the location of the promotion),
- 5.7 checkmate and stalemate.
- 6. The referee does not review announcements more than one move old, and does not recapitulate losses.
- 7. The referee does not announce in the usual manner rebuffs obtained by attempts to move which are illegal per se.  
(E. g., moves to or through a square occupied by one's own man; diagonal movement of rook; attempts to remove check other than by moving king, interpose or capture compatible with the announced character of the check.)  
A special rebuff (e.g., 'hell, no!') is used here.
- 8. The referee tries to eliminate errors. E.g., if the player misidentifies the square named in an announcement, the referee will correct him, if it can be done without significant information accruing to the player. Referee's blunders range from trivial to catastrophic; the remedies include general reprimand, reverse play, declaring game void.
- 9. A player may, before moving, demand a count of the rebuffs (no's) sustained by his opponent on the last move; he may in fact demand recapitulations before the opponent completes his move.

10. A player may attempt any move which is compatible with his own situation (men and deployment) and with the referee's current announcement. He is not required to have memory of previous plays, or to make logical inferences.
11. A move is completed when a piece touches the board or a presumed enemy piece on a legally admissible square.
12. En passant capture options are announced in the same manner as other options are announced; the fact that they are en passant is not specified.
13. Castling may be done in the presence of the enemy provided the king does not use or transit an affected square.
14. The referee's arrangement is always final in the event of a dispute over the position of players' pieces. Indeed, all referee's decisions are final.
15. Check by a pawn is announced as if it were a bishop or queen, but without stating that it is a pawn."

#### THE ROLE OF THE REFEREE

The task of refereeing a game of Kriegsspiel can be an exacting one. During the course of a game, even with an experienced referee, incorrect announcements are not infrequent. For example, a pawn capture that exposes one to check is obviously not a legal pawn capture, yet such a capture possibility ("try") is often announced (and later retracted) by even the best referees. There are no heuristics involved in refereeing a game, no "best" set of announcements that can be made by the referee for a given attempted move by a player in a given board situation. For each

attempted move in each board situation there is a particular set, and only one set, of announcements that the referee can make (e.g., the set "White to move, file check, pawn try K5," the single-member set "No," etc.).

The UMPIRE program has the task of the referee. It is not, nor is it intended to be, a game-playing or heuristic program. It makes its announcements solely on the basis of the current board configuration and the attempted move by the player. However, it can and does keep track of the game as it progresses and make accurate announcements of any possible board configurations and attempted moves.

#### USER INTERACTION WITH UMPIRE

UMPIRE, when it is on the disc, may be loaded with the usual TSS command !LOAD UMPIRE. It takes 46792 words of drum space to load the program. If it is not on the disc, tape numbers are available upon request from the author.

Following the usual TSS GO command, UMPIRE will print a Header and then ask the user (i.e., the player initiating the game) DO YOU WANT INSTRUCTION Y/N? A YES (or Y) answer to this question will cause printout of some subset of the instructions given in this paper. UMPIRE then asks for the channel (i.e., teletype) number you are using. It will then ask for the number of your opponent's channel. This latter must be a channel that is not currently in use; that is, its TSS status must be \$NO LOGIN. A space (blank) is required after entering each of these numbers and before striking the carriage return.

After entry of these two numbers, the TSS JOIN command is executed, causing UMPIRE to run on both opponents' channels at the same time. It is important to remember that, from this time on, the program is quite

sensitive to receiving input at the wrong time, or from the wrong channel. Whether you are on the channel that initially loaded the program, or on the opponent's channel, UMPIRE will request input from you by printing four asterisks (\*\*\*\*) and ringing the teletype bell twice. Even if your TSS status is \$WAITING FOR TTY, input to the program should not be made unless this request is printed out first. (The system can be wrecked if it receives input when it is not expected, or from an unexpected channel.)

After the join is effected, the opponent's channel receives a message informing him of the join and he is offered the option of having the instruction set printed out. The initiating user is then asked whether he is to play White or Black. (This can be decided using the TSS DIAL command, or the UMPIRE MSG command (see below)). Both channels receive a message stating the decision and the game begins. The message (WHITE TO MOVE) is printed on both teletypes and the channel selected for White is asked for input.

Moves are input in the form of a modification of the Standard American Chess notation. Regardless of whether you are playing White or Black, the square on which your King rests at the opening of the game is called "K1." A move is entered by indicating, first, the square on which the piece (or pawn) you wish to move rests (rather than the name of the man) and, second, the name of the square to which you wish to move. An initial move might be (for either side) K2 K4, which is the same as P-K4 in Standard American Chess notation and means that the King's pawn is moved forward two squares. Castling is recognized by UMPIRE as the letter C followed by either the letter K or the letter Q for King's

side and Queen's side castling, respectively. Thus, CK indicates to the program that you wish to castle on the King's side. Responses from UMPIRE are printed on both teletypes. During the game you will see the responses to both your moves and your opponent's moves. Responses you may receive are as follows:

(1) After you have entered your move:

(NO)

(HELL NO)\*

(REPEAT NO)

(NOT UNDERSTOOD-ENTER AGAIN)

\*This is a technical term meaning you should have known better.

You are then asked to enter another move.

(2) When it becomes your move, but before you are asked for input:

(WHITE TO MOVE)\*

(BLACK TO MOVE)\*

(PAWN TRY xx)\*\*

(PAWN GONE xx)\*\*

(PIECE GONE xx)\*\*

(RANK CHECK)

(FILE CHECK)

\* Not both of these on the same move.

(LONG DIAGONAL CHECK)

(SHORT DIAGONAL CHECK)

\*\* Where xx is the name of some square.

(KNIGHT CHECK)

(WHITE PROMOTES)\*\*\*

\*\*\* Not both of these at the same time.

(BLACK PROMOTES)\*\*\*

(3) When your move has been successfully completed and

it is now your opponent's turn to move:

(WHITE TO MOVE) or (BLACK TO MOVE)

There are three special commands that you may use anytime you receive the request for input (\*\*\*\*):

MSG: If you wish to send a message to your opponent.

You will receive another request for input and everything you type on this next line will be printed on both teletypes.

RELIST: If you wish to compare your board position with the (correct) position of your men as contained in the program.

The positions of your men only will be printed on your teletype.

RESIGN: If you give up.

Two "short cuts" were taken in program design, to prevent a great deal of unnecessary computation:

(1) UMPIRE will not recognize checkmates and stalemates, and a game will be terminated only if one of the opponents resigns. Most games end in a resignation anyway, and a checkmate or stalemate can be tested for at any time.

(2) When a pawn reaches the eighth rank and is promoted, it automatically becomes a Queen. If it is shown to be desirable, the program will be revised at some future date to allow the player to choose the promotion piece.

#### PROGRAM DESCRIPTION

The basis of the scheme for recording the positions of the men on the board, and for controlling their movements, was taken from a paper by George Baylor, formerly an SDC summer associate, in which he describes a program that evaluates mating combinations in ordinary chess [2].

In UMPIRE the squares are numbered S1...S64, beginning in White's QR1 and continuing across the rank from left to right and from bottom to top. S8 is thus White's KR1, S9 White's QR2 and so on to S64, which is White's KR8. The ranks themselves are numbered R1 through R8 from White's bottom to top, and the files are numbered F1 through F8 from White's left to right. Each square is contained in a list associated with both the rank and file that intersect on that square. For example, S22 is contained in the list R3 and in the list F6. The list RANKS contains, in turn, the names of the lists R1, R2, ..., R8; and the list FILES contains the lists F1, F2, ..., F8. The diagonals are numbered D1 through D30 and, as with the ranks and files, each diagonal contains a list of the squares on that diagonal.

The men are numbered M1 through M32, beginning with White's QR and ending with Black's KR. Each square is associated with a list of attributes: RANK, FILE, LD, SD, KING, KNIGHT, and MO which contain, respectively, the following values: The rank, and the file, in which the square appears, the longer of its two diagonals, the shorter diagonal, the list of squares to which a King can normally move, the list of squares to which a Knight can normally move, and the man currently resting on that square. The attribute MO (man on) is the only dynamic attribute for any square. Most of these attributes are initially set at compile time of the program by a function called MAKEIT which is compiled, run once, and then deleted from LISP's binary program space, since it will never be used again.

The operating program consists of approximately 47 functions. Four of these, TOP, START, EVALU, and EXEC, are supervisory and are used only as control functions to initiate the system and allow supervisory debugging. For example, EVALU allows a mode of operation analogous to the Evalquote mode in the LISP system.

The functions JOIN, UNJOIN, INPUT, OUTPUT, and ONEOUT are I/O functions. JOIN and UNJOIN modify the LISP File call into a TSS Join or Unjoin call, thereby permitting Q-32 LISP 1.5 to use two interactive time-shared consoles concurrently. Either teletype can then be selectively asked for input by the INPUT function. OUTPUT transmits data to both teletypes with one call, while ONEOUT selectively outputs data to only the console addressed.

INTERP and OUTERP are translating functions that translate the Standard American Chess notation, as input by the user, into the internal language used by the program and back again for output.

The top-level supervisory function (MOVEEXEC) which is entered at the beginning of a game, loops throughout the entire game, flipping a Boolean variable (WB) from "true" to "false" once during each loop and thus determining whether it is White or Black that has the move. A move is requested for input from the proper teletype and is first examined for legality by the function POSSIBLE. If the move is impossible to make (a Rook moving on the diagonal, for example) the message (HELL NO) is output and another move is requested. When a move passes the first legality check, another legality check is made, this time based upon the position of the opposing men. For pieces that may move over multiple squares (such as Bishops, Rooks and Queens), the intervening squares



are examined using the function BETWEEN, for the presence of opposing men. (If the intervening squares were occupied by his own men, function POSSIBLE would have generated a HELL NO message.) If no intervening men are found, the values for the property MO (man on) of the "from" and "to" squares given in the attempted move are revised (function MAKEMOVE), updating the program's representation of the game to account for the move. MAKEMOVE also returns a message indicating the capture of a piece or a pawn. The player's King is then examined for a possible check (functions CHECK and ROWCH). If no check is discovered, the move is considered legal. If, on the other hand, the player's King is found to be in check, the move is retracted (function UNMOVE), the message (NO) is output, and another move is requested.

Following a successful move, the function PAWNTRYIS searches out any pawn capture possibilities for the opponent, the function CHECK determines if the opponent is now in check, the appropriate messages are output (including the capture message, if any, generated by MAKEMOVE), and MOVEEXEC flips the WB variable to start another loop.

During the game, each successful move is stored on a list and is returned as the value of MOVEEXEC. This list can be requested by, and listed for, either or both players.

SUMMARY

UMPIRE represents a somewhat unique combination of computer usages: It is first of all a game-playing program that, while still non-heuristic, is considerably more complex than the tic-tac-toe players and other deterministic programs. As a time-shared program, it is also a multiple-console command and control system that requires more than one interactive user. Finally, viewed as a command and control system, it provides a decision making program capable of supervising the use of intricately interrelated symbolic data.

17 February 1967

-17-  
(last page)

SP-2752

REFERENCES

1. Williams, J. D. Kriegsspiel rules at RAND. The RAND Corporation, Santa Monica, California, Memorandum, April 17, 1950, 3 pp.
2. Baylor, G. W. Report on a mating combinations program. SDC document SP-2150, June 4, 1965, 78 pp.