Notes on the back story of this letter:

This is my reply to ***Michael Tarnowski***, who back in 1980 wrote a letter to me (alas, probably lost) with a number of questions about my program ***"52206 A Chess Game"*** for the *HP-67*, which I had earlier submitted to the ***HP Users' Program Library Europe***. It succeeded in catching Mr. Tarnowski's eye upon seeing its abstract in the *UPLE HP-67 Catalogue*, so he promptly got it from there.

He was mostly interested in the program's inner workings and thus I proceeded to explain them in great detail to him by means of this reply letter, and even included a hand-drawn flowchart of the program's logic used to determine its move, as well as relevant comments.

I also offered to exchange *UPLE* programs with him, and we shared our respective lists of the ones we had, but ultimately nothing came of it.

Valentin Albillo, 24-09-2021

Michael Tarnowski
Fritz Kalle Str. 4
6200 Wiesbaden
West GERMANY

Valentín [illegible]
Padre Rubio,61 - 2º C
Madrid 29
SPAIN

Dear Sir:

Pleasant to receive your letter, find included the flowchart you asked for. The idea of the program was first proposed to me by a friend of mine; after solving the puzzle, I decided it could be a good idea to program it as a game for the HP-67/97, and once I succeeded, it was submitted to the User's Library, hoping someone else would like it.

Now, I'll give you an explanation about how the machine determines its moves: you are right , the machine does not store the user's move, but it keeps it stored in the stack because, as you may see either in the flowchart or the Program listing, most operations performed by the program to determine its next move are simply comparisons; after each, the user move is taken back to the display, ready for another test.

The strategy used by the program is as follows: among the many thousands of different games, there are 21 main possibilities. In all of them, the two first moves performed by the calculator are:

1.P-K 4 , 2.Q-Kt 4

regardless of user's moves; the first user's move is not stored, as the machine moves do not depend on it. Once this two moves are performed, the 2nd user's move is considered: if it places the Black king onto the King's Bishop column, then the 3rd White move is:

3.Q-Q 7

otherwise, it will be: 3.Q-Kt 7

Upon the first case, the 3rd Black move is analyzed: if it places the Black king on the royal row (the eight row, counting upwards), then the 4th White move is:

4.P-QKt 3

After black moves, its King may be in R1 , KB 1 , KKt 1 ; in all cases, White plays and checkmates in two moves.

If the 3rd user's move didn't place its King on the royal row, then the 4th White move is:

4.P-Q 4

and after Black moves, its King is in KB 3 , KKt 3 , KR 4 or KR 5 , or Black has taken the White pawn. As before, White checkmates in two in all cases, except the last one, which is a mate in one.

If the 2nd user's move does not place its King on any location of the King's Bishop column, the 3rd White move will be:

3.Q-Kt 7

and, like before, if the next black move puts its King on the royal row, the 4th White move is:

4.B-Kt 5

after black moves, its king is on QR1, QKt1, QB1, Q1 , and a mate in two is performed,but the last one (mate in one).

If black does not move to royal row, White moves:

4.P-Q 4

then black moves and its king is in QKt3, QB3, Q3, K3, QR4 or QKt5 , following a mate in two.

As you may see there are 13 mates in six moves, and 8 mates in five moves, thus making 21 possibilities in all.

All possible games make use of no more than 22 different moves notation (several different moves have the same notation), so all possible White moves are stored onto a single magnetic card, which is the main reason for both the shortness of the program and its execution speed.

As a final remark, it may be deduced from the abo-

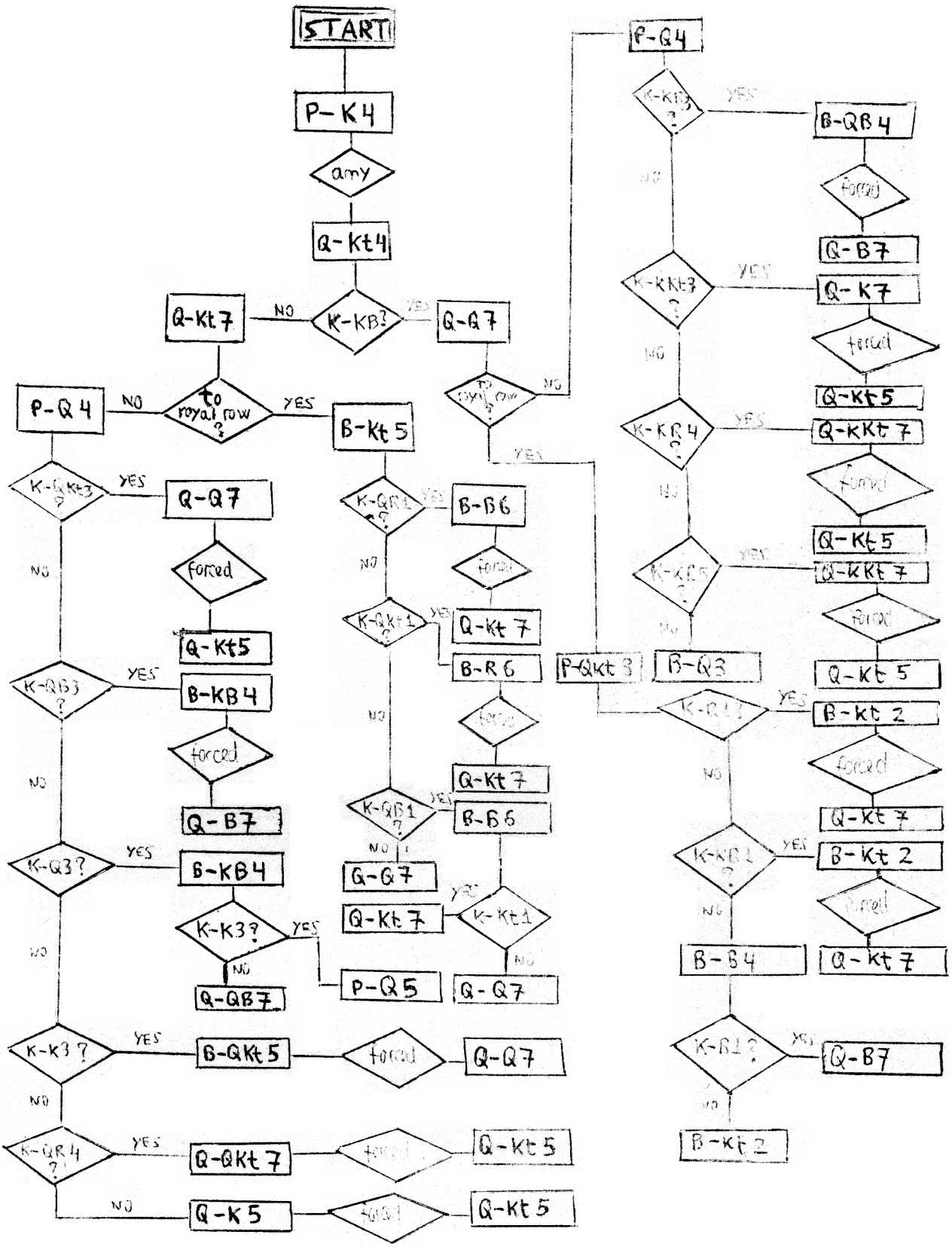
ve paragraphs that Black can never win. In all possible games, - he/she will be checkmated in no more than six moves , even less sometimes. That is the reason why I didn't include a flowchart - of the strategy, nor make any comments about the program listing: I tried to suggest that the user could beat the machine. That's - not so: the strategy followed by HP-67/97 has no flaws, thus ma - king a victory quite impossible.

I hope those comments will not ruin your - opinion about the program ; even if you know you can't win, it's- still a very curious program to show all your skeptic friends - what your little (or less little, if you have a 97) machine can do. Defy them to beat the machine: in most cases, they will be sure - beforehand that your machine couldn't do any well , but look at - their faces after being checkmated once and again ! Furthermore , the machine will take much less time to "think" its next move - than themselves.

That's all . I hope it could be of any help to you. I sold my HP-67 long ago, and got the new HP-41c , but I may dispose of a 97 if needed; if you need any additional informa tion, don't hesitate to write.

Best regards ,

P.D.- I've got well over two hundred programs from the User's Li - brary, and I am very interested over the possibility of ex - changing programs with you. If you are interested too, send- me two lists: one with the reference numbers of the programs you have, and the other with the numbers of the programs you would like to have , and I'll do the same.



BLACK INK = calculator moves

BLUE INK = user's moves

Yellow = checkmate