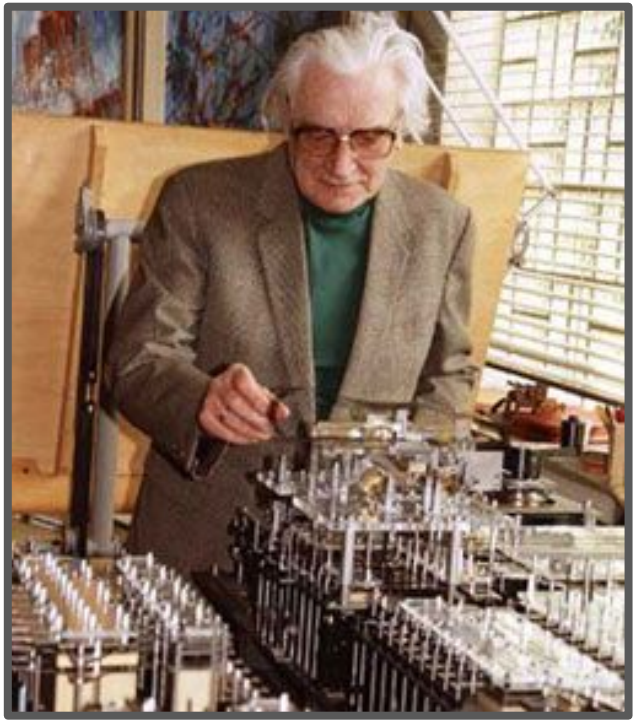
“Plankal-cool”

The First Ever Programming Language

Konrad Zuse

- 22 June 1910 – 18 December 1995
- German Civil Engineer, Computer Scientist and Inventor.
- Worked in an aircraft factory where he had to do many routine calculations by hand.
- Built the **first** programmable computer (1941)
- Z1 contained some 30,000 metal parts
    - original blueprints destroyed by a British air raid.
- “Zuse completed his work entirely independently of other leading computer scientists and mathematicians of his day.”



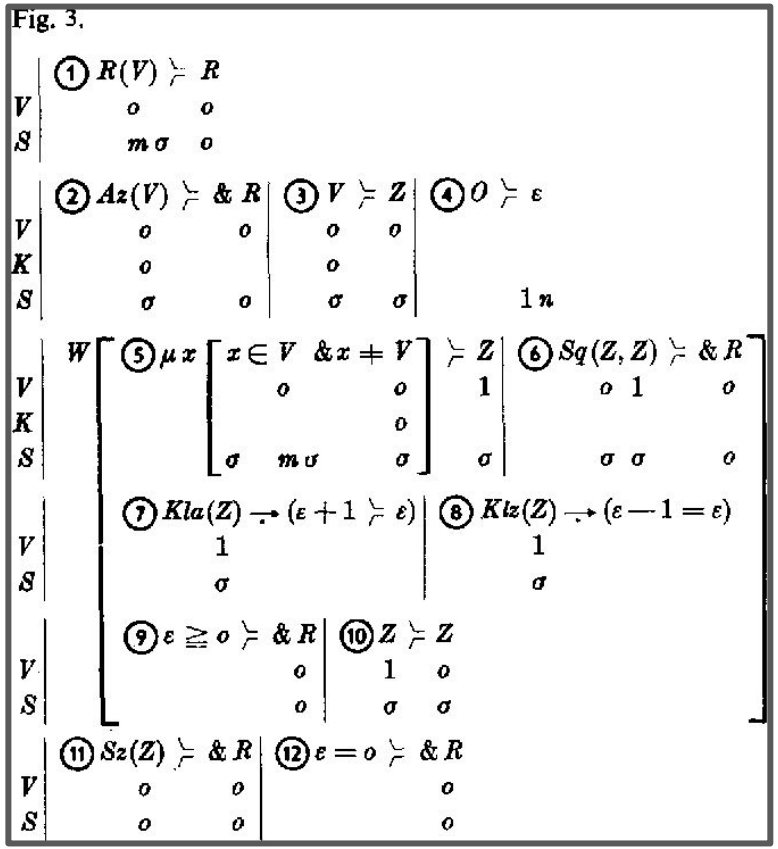
The first ever programming language?

- Working in machine code was too complicated
- PhD Thesis years ahead of its time.
- In comes the first ever high-level programming language

Plankalkül

- Designed during 1942-1945
- Chess Engine ‘example’ program

Fig. 3.




```
P1 max3 (V0[:8.0],V1[:8.0],V2[:8.0]) → R0[:8.0]
max(V0[:8.0],V1[:8.0]) → Z1[:8.0]
max(Z1[:8.0],V2[:8.0]) → R0[:8.0]
END
P2 max (V0[:8.0],V1[:8.0]) → R0[:8.0]
V0[:8.0] → Z1[:8.0]
(Z1[:8.0] < V1[:8.0]) → V1[:8.0] → Z1[:8.0]
Z1[:8.0] → R0[:8.0]
END
```

What is Plankalkül?

- Plankalkül is a typed high-level imperative programming language.

- Programs are reusable functions, and functions are not recursive.
- Variables are local to functions (programs).
- Fundamental data types are arrays and tuples of arrays, but there are also floating point, fixed point, complex numbers; records; hierarchical data structures; list of pairs.
- There is no GOTO construct

- Assignment operation (e.g.: V1 + V2 => R1).
- Conditional statement (e.g.: V1 = V2 => R1. This means: Compare the variables V1 and V2: If they are identical then assign the value true to R1, otherwise assign the value false.

- Possibility for defining sub-programs and loops.
- Logical operations (predicate logic and Boolean algebra).

- Single primitive type (bit / boolean) = S0

$A1$	$S1 \cdot 3$	Coordinate of chess board (it has size 8x8 so 3 bits are just enough)
$A2$	$2 \times A1$	square of the board (for example L00, 00L denotes e2 in algebraic notation)
$A3$	$S1 \cdot 4$	piece (for example, 00L0 — white king)
$A4$	$(A2, A3)$	piece on a board (for example L00, 00L; 00L0 — white king on e2)
$A5$	$64 \times A3$	board (pieces positions, describes which piece each of 64 squares contains)
$A10$	$(A5, S0, S1 \cdot 4, A2)$	game state ($A5$ — board, $S0$ — who moves, $S1 \cdot 4$ — possibility of castling (2 for white and 2 for black), A2 — information about cell on which En passant move is possible

Hello World!

```
R1.1(V0[:sig]) => R0
R1.2(V0[:m x sig]) => R0
0 => i | m + 1 => j
[W [ i < j -> [ R1.1(V0[i: m x sig]) => R0 | i + 1 => i ] ] ]
END
R1.3() => R0
'H';'e';'l';'l';'o';',';' ';'w';'o';'r';'l';'d';'!' => Z0[: m x sig] R1.2(Z0) => R0
END
```

In summary - Zeus was pretty kül

- Developed the first known formal system of algorithm notation capable of handling branches and loops.
- Zuse discovered that the calculus he had **independently** devised already existed and was known as p**ropositional calculus.**
- He never got to finish his chess program..

“In 1945, Zuse described Plankalkül in an unpublished book...

The **collapse of Nazi Germany, however,** prevented him from submitting his manuscript.”

Thank you!