

BCPL (OCODE)

Michael Nestler

BCPL

- Basic Combined Programming Language
- Martin Richards –1966 – University of Cambridge
- "Grandfather of C"
- Basis für Amiga

Hauptquelle: <https://www.cl.cam.ac.uk/~mr10/>
(bcplman.pdf, bcpl.tgz)

BCPL – Beispiel

```
GET "libhdr"
```

```
LET start() = VALOF
```

```
{ FOR i = 1 TO 5 DO writef("fact(%n) = %i4*n", i, fact(i))
```

```
    RESULTIS 0
```

```
}
```

```
AND fact(n) = n=0 -> 1, n*fact(n-1)
```

OCODE

- "Object Code"
- Zwischencode für Compiler
- Abstrakte Maschine mit 4 Registern:
 - G – Pointer zu globalem Vektor
 - PC – Instruction Pointer
 - P – Frame Pointer
 - A – Result Register
 - S – Kein Register, "current stack frame size"

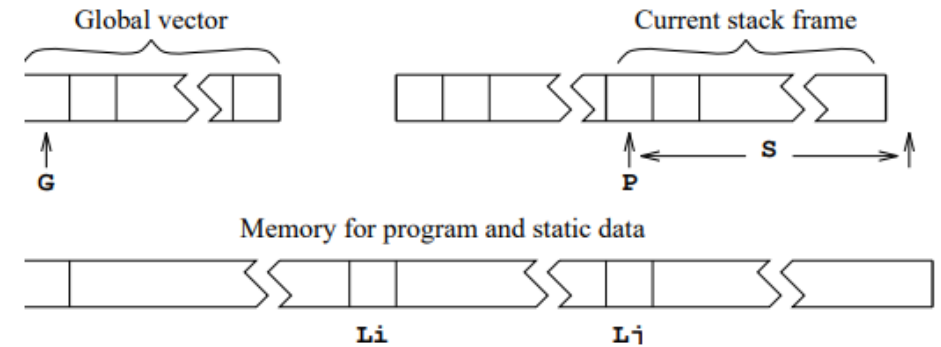


Figure 8.2: The BCPL abstract machine

```
LSTR 5 'H' 'e' 'l' 'l' 'o'
LP 3
GETBYTE
SL L36
```

Implementierung

- Interpreter in TypeScript
- Wortgröße 32-bit
- Unoptimierter Switch-Dispatch
- Speicher als Int32Array
 - 64 Mi-Einträge → 256 MiB

Stdlib

- 17 Funktionen implementiert
- Arg parsing
- Konsolen-IO
- File IO (nur für Konsole)
- Datum, Datumsformatierung

Benchmark

- "Coins" – Möglichkeiten, Betrag mit Münzen zu zahlen
- Cintsys: 0.8s
- TypeScript (Bun): ~30s
- TypeScript (UI): ~40-48s