



THIS PAGE IS
INTENTIONALLY
LEFT BLANK.

Kazimir Majorinc

LISP 2 (drugi dio)

Povijest Lispa 28.



Razmjena vještina
Hacklab u mami
4. svibnja 2013.

Definiranje funkcija

Funkcije se u SL i IL definiraju izrazima oblika

$$t \text{ FUNCTION } n \text{ } (x_1, x_2, \dots, x_n); d_1, \dots, d_k; e \\ (\text{FUNCTION } (n \text{ } t) \text{ } (d_1 \text{ } d_2 \text{ } \dots \text{ } d_k) \text{ } e),$$

gdje je t tip vrijednosti funkcije, n je ime funkcije, x_i su imena parametara, d_i su deklaracije (u IL-u sadrže i ime varijable) i e je izraz koji se izračunava u vrijednost funkcije.

Deklaracije

Kako izgledaju deklaracije? Nažalost, nisam našao primjere. Znamo da je sa svakom varijablom asociran tip (tip **SYMBOL** dozvoljava sve vrijednosti), način spremanja (engl. storage mode) i način prijenosa (engl. transmission mode). Postoje tri načina spremanja: fluidni, vlastiti i leksički. Fluidne varijable se mogu pozivati izvan programa u kojima su definirane, a leksičke ne mogu. Fluidne varijable su općenitije ali i izloženije konfliktu imena. Uglavnom se koriste za komunikaciju između odvojeno kompajliranih programa. Konačno, vlastite varijable koje su dizajnirane za komunikaciju između programa. Varijable mogu označavati podatak direktno ili indirektno.

Funkcije mogu imati neodređeni broj argumenata.

```
REAL FUNCTION SUMSQUARE(X(I));  
BEGIN INTEGER J; REAL Y;  
  FOR J*-1 STEP 1 UNTIL > I DO  
    Y ← Y + X(J)|2;  
  RETURN Y  
END
```

```
(FUNCTION (SUMSQUARE REAL)  
  ((X INDEF I))  
  (BLOCK ((J INTEGER) (Y REAL))  
    (FOR J (STEP 1 1 GR I)  
      (SET Y (PLUS Y (EXPT (X J) 2))))  
  (RETURN Y)))
```

Prevodioci sintakse i META

Programi pisani u SL-u prevode se u IL kompajlerom koji se naziva „prevodioc sintakse“ (engl. „syntax translator.“) No, sam prevodioc sintakse je rezultat kompajliranja.

Definiran je kompajler-kompajler META koji učitava opise sintakse SL-a u „proširenoj Backus-Naur formi“ i kompajlira te opise u „prevodioce sintakse.“ „Prevodioci sintakse“ su bili neefikasni; posebno su loše surađivali sa garbage collectorom, što je zahtjevalo ulaganje napora u optimizaciju.

Računalo na kojem je implementiran, IBM AN/FSQ-32V computer at the System Development Corporation, Santa Monica, California, je koštalo 7 milijuna dolara (danas oko 50 milijuna dolara) i bilo teško 90 tona. Imalo je 48 K memorije (riječ 48 bita).

Joseph Weizenbaum, MIT, kritizira LISP 2 u IEEE transactions, **1967**. Negativno. LISP 2 pokušava nadoknaditi dva glavna nedostatka LISPa: sintaksu i neefikasnost. Sintaksa je nešto ljepša, ali programer mora koristiti i SL i IL jezik.

LISP 2 uopće nema interpreter što čini izračunavanje koda (npr. funkcija) generiranih za vrijeme izvođenja programa vrlo sporim, i to će biti opterećenje za programere. **Weizenbaum** smatra LISP 2 razočaranjem.