



THIS PAGE IS
INTENTIONALLY
LEFT BLANK.

Kazimir Majorinc

**Algebarski jezik za obradu
simboličkih izraza (II).**

Revizije.

Povijest Lispa 4.

Razmjena vještina

Hacklab u mami

28. srpnja 2012.

Prethodno

Prijedlog Ljetnog projekta, kolovoz 1955. Filozofski.

Prijedlog kompajlera, prosinac 1957. Dinamičan, refleksivan, skoro paralelan, ali još nije Lisp.

Prijedlog programskog jezika, travanj 1958. Budući Algol 58. Uvjetno grananje, makro naredbe.

Algebarski jezik za obradu simboličkih izraza (MIT AIM-001), rujan 1958. Pravi Lisp. Podjela riječi, =, **go**, reprezentacija izraza kao jednostruko vezanih lista, **add**, **dec**, **car**, **cdr**, **consw**, **erase**.

Varijante funkcije cons

Funkcija **consw**(w) sprema w u prvu slobodnu riječ, briše tu riječ iz liste slobodnih riječi i vraća njenu adresu.

Funkcija **consel**(a, d) konstruira riječ

a	d
-----	-----

, sprema je u prvu slobodnu riječ, koju onda briše iz liste slobodnih riječi i vraća njenu adresu.

Funkcije za pomicanje pokazivača

Funkcija **Point(J, K)** kreira listu

K	0
----------	----------

 i postavlja adresu u **J**.

Mova(J) pomiče pointer na adresu koja se nalazi u adresnom dijelu riječi. Npr. nakon **mova(J)**, **J** ima vrijednost **K**.

Movd(J) pomiče pointer na adresu koja se nalazi u d dijelu riječi. Npr. nakon **movd(J)**, **J** ima vrijednost **0**.

Puno sličnih funkcija.

Osnovne operacije nad cijelom listom.

Potprogram **eralis** briše cijelu listu (i upisuje adrese u listu slobodnih adresa.) Najstariji je sačuvani program u Lispu.

<code>subroutine eralis (J)</code>	
<code>/ J = 0 → return</code>	Vertikalna zagrada /.
<code>go (a (cir(J)))</code>	cir = indikator, dio riječi
<code>a(1) jnk = erase (car (J))</code>	Prvi element nije lista. Briši!
<code>a(0) eralis (dec (erase(J)))</code>	Ne briši prvi element!
<code>return</code>	
<code>a(2) eralis (car (J))</code>	Prvi element je lista. Briši!
<code>\ go (a(0))</code>	Vertikalna zagrada \.

Potprogram **eralis** za slučaj da se original ne vidi dobro:

subroutine eralis (J)		
/	J=0 → return	Vertikalna zagrada /.
	go (a (cir(J)))	cir = indikator, dio riječi
a(1)	jnk = erase (car (J))	Prvi element nije lista. Briši!
a(0)	eralis(dec(erase(J)))	Ne briši prvi element!
	return	
a(2)	eralis (car (J))	Prvi element je lista. Briši!
\	go (a(o))	Vertikalna zagrada \.

Osnovne operacije nad cijelom listom.

Funkcija **copy** stvara kopiju liste u slobodnoj memoriji. Vraća adresu kopije.

Potprogram **search**(L, J, p, \dots) sprema u J adresu prvog elementa liste koji zadovoljava uvjet p . Puno opcija.

Funkcija **maplist**($L, J, f(J)$) stvara listu ($f(L_1), \dots, f(L_n)$) gdje L je (L_1, \dots, L_n) i vraća adresu.

Funkcija **list**(i_1, \dots, i_n) vraća (i_1, \dots, i_n). Predikat **equal**(L_1, L_2), u smislu “by value.”

John McCarthy

A revised version of maplist

Oznaka MIT AIM-002. Izgubljen. MIT posjeduje AIM \geq 006. **Stoyan** je na Webu objavio lošu kopiju prve strane AIM-002. U zbirci Computer History Museuma, koja uključuje i Stoyanovu zbirku, tog memoa nema. No, čini se da je AIM-002 u potpunosti obuhvaćen sa AIM-004.



Herbert Stoyan

John McCarthy
Symbol-manipulating language.
Revisions of the language.



Oznaka MIT AIM-003. 21. listopad, 11 strana, šapirografirano. Nekoliko unapređenja u smjeru “elegancije.” Jezik je dizajniran za obradu aritmetičkih izraza, programa u mašinskom ili višim jezicima, iskaznom računu i drugim formalnih jezicima matematičke logike. Bavimo se samo imperativnim naredbama. Deklarativne rečenice izražavat će se u advice takeru.”

Vanjski oblik izraza

Simboli su nizovi duljine od 1 do 120 znakova, sa najviše jednom decimalnom točkom. Dakle, i **3.14** je simbol.

Izrazi su definirani na slijedeći način:

1. Simbol je izraz.
2. Ako su e_1, e_2, \dots, e_n izrazi, onda je $i(e_1, e_2, \dots, e_n)$ izraz.

Posebno, ako je e_1 izraz, onda je (e_1) izraz, različit od e_1 .

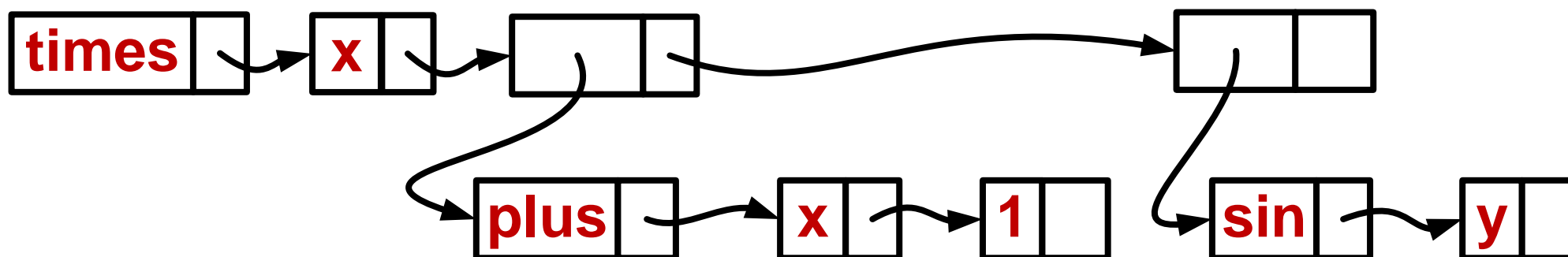
Npr. $x \cdot (x+1) \cdot \sin y$ je predstavljen kao

(times, x, (plus, x, 1), (sin, y)).

Ima li ovaj izraz 4 ili 28 elemenata?

Unutrašnji oblik izraza

Jednostruko vezane liste, kako su opisane. Odbačeni su “tagovi”: sve podliste su dio lista. Npr. $x \cdot (x+1) \cdot \sin y$



Pravokutnik	Riječ u memoriji
Početak strelice	Adresa riječi na koju pokazuje strelica
Simbol	Adresa liste svojstava u memoriji
Prazan	0

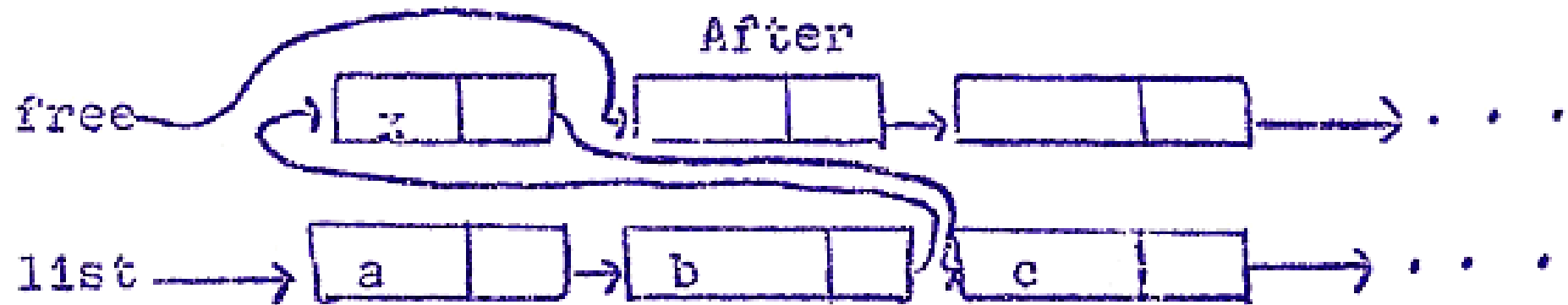
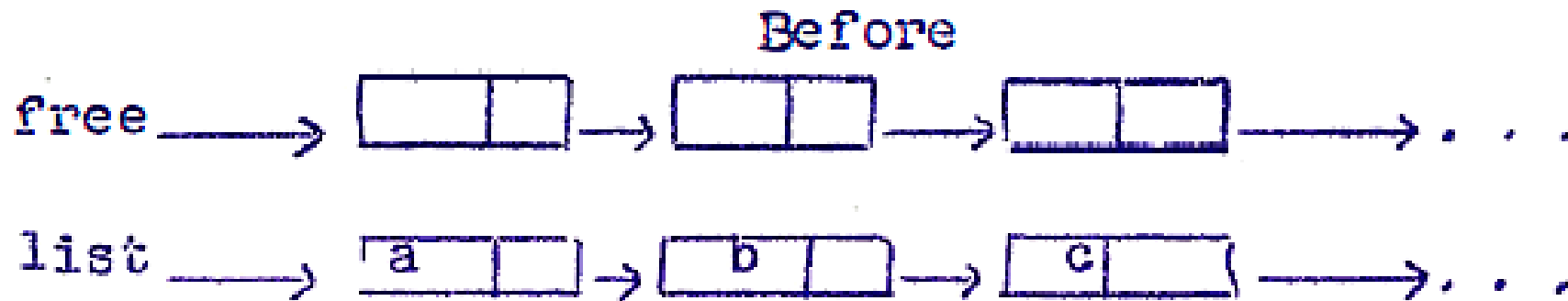
Lista svojstava

Nije baš prava lista, ali neke operacije definirane na listama mogu biti primjenjene na nju.

Sadrži podatke poput imena simbola; lokacije liste svojstava, informacije je li simbol zapravo broj, npr. **1** ili funkcija, npr. **sin**, je li varijabla ili konstanta, ako je funkcija onda može u listu svojstava biti upisana derivacija ili interal funkcije itd.

Lista slobodne memorije

Prava lista.



Problem prazne liste.

Slijedeći put:

McCarthy, *Programs with common sense*, izlaganje na konferenciji “Mechanisation of thought processes”, London, 1958.