

This page is intentionally left blank.

McCarthyjevi zapisi o jeziku
do 1957.

John McCarthy, Marvin Minski, Nathaniel Rochester i Claude Shannon, *Proposal for the Dartmouth summer research project on artificial intelligence*¹, 1955

Ambiciozan uvodni dio:

¹ Časopis “The message of the East” iz 1933. savjetuje: „Nemoj razvijati umjetnu inteligenciju, nego razvijaj inteligenciju koja je od Boga; od potonje proizlazi vrlina, od prve lukavost.“

“Predlažemo da deset istraživača dva mjeseca proučava umjetnu inteligenciju ... Pokušat ćemo postići da strojevi koriste jezik, stvaraju apstrakcije i pojmove, rješavaju probleme koje su do sada samo ljudi rješavali, poboljšavaju sami sebe. Vjerujemo da se značajan napredak može ostvariti ako pomno izabrana grupa znanstvenika radi zajedno tijekom ljeta.”

Predlagači navode i individualne planove.

McCarthy piše:

“Tijekom slijedeće godine i Ljetnog projekta istraživao bih odnos jezika i inteligencije. Čini se jasnim da direktna primjena metoda pokušaja i pogreške na odnos između opaženih podataka i kretanja ne može voditi do vrlo složenog ponašanja. Umjesto toga, nužno je metodu pokušaja i pogrešaka primjeniti na višim stupnjevima apstrakcije. Ljudski um očito rabi jezik pri obradi složenih pojava. Procesi pokušaja i pogrešaka na višem nivou često uzimaju oblik formiranja i testiranja hipoteza. Engleski jezik

posjeduje niz svojstava koji nedostaje svim dosad razvijenim formalnim jezicima.

1. Argumenti u engleskom, podržani neformalnom matematikom mogu biti koncizni.
2. Engleski jezik je univerzalan, u smislu: u engleskom jeziku se mogu definirati drugi jezici i onda koristiti kad je prikladno.

3. Govornik engleskog jezika može govoriti o sebi i opisivati svoj napredak pri rješavanju problema.
4. U engleskom jeziku mogu se definirati pravila za dokazivanje, kao i pravila za postavljanje hipoteza.

Logički jezici definirani do sada su bili ili nizovi instrukcija za računala, ili formalizacija dijelova matematike.

Čini se poželjnim definirati umjetni jezik koji bi računalo moglo koristiti za rješavanje problema koji zahtjevaju postavljanje hipoteze i refleksiju. Kratke rečenice i argumente na engleskom jeziku bi trebalo biti moguće izraziti kao kratke rečenice i argumente u umjetnom jeziku. Pokušati ću uobličiti takav jezik, koji bi dodatno, sadržavao i pojmove za fizikalne objekte i događaje. Nadam se da bi se njegovim korištenjem moglo programirati strojeve da uče igrati igre i izvršavati druge zadatke. “

McCarthy, A proposal for a compiler, M.I.T.
Computation Center, 13. prosinca 1957

“Kako bi dobar izvorni jezik trebao izgledati?”

To nije engleski jezik:

“Činjenica je, engleski jezik je vrlo loš za davanje složenih instrukcija.”

To nije uobičajeni matematički jezik:

“Prvo, matematički simbolizam je, uglavnom, korišten za izražavanje deklarativnih rečenica; programiranje treba imperativne rečenice. Drugo, novi pojmovi se u matematici definiraju gotovo uvijek u neformalnom jeziku.”

“Čini se najvjerojatnije da će biti razvijen posebni simbolički jezik koji će koristiti najvrijednije elemente prirodnih i matematičkih jezika”

“Pojmovi koji se u postojećim sistemima izražavaju na složen način će biti izražavani na jednostavan način. Zato će jezik biti teži za učenje od postojećih.”

“Programski jezik je koordinatni sustav u prostoru procedura.”

“Program može generirati drugi program u izvornom kodu i pozvati kompajler da ga kompajlira.”

“Izvorni jezik će biti dovoljno općenit da se u njemu napiše kompajler. Zato kompajler može biti napisan na ‘bootstrapping way’, tako da se rane, neefikasne verzije koriste za kompajliranje kasnijih, sa dodanim svojstvima.”

“Najvažnije svojstvo novog jezika je sloboda koju daje programeru da definira nove načine kojima će se izraziti.”

“1. ‘izjava o ekvivalenciji’ koja omogućuje uvođenje pokrate za bilo koji izraz.”

“2. Prevodioc počinje sa tablicama koje opisuju kako se izvorni kod prevodi. Izvorni kod može proširivati i mijenjati tablicu. To, naravno, uključuje i tablice koje određuju kako prevodioc reagira na promjenu tablice.”

3. Programer može modificirati kod kompajlera.

“4. Izračunavanje se može opisati tako da se konačno stanje stroja zada u

pojmovima početnog stanja stroja, bez potrebe da se brine o pojedinačnim promjenama korištenih varijabli”

“5. Prošireni skup osnovnih veličina i operacija u usporedbi sa Fortranom, uključujući riječi sa fiksnim zarezom, logičke riječi, veličine od jednog bita. “

“6. Direktan način rukovanja iskazima, predikatima i uvjetnim funkcijama kojim se može eliminirati mnoga grananjanja u izvornom programu. “

“7. Krupno poopćenje pojma indeksiranih varijabli, tako da indeksi mogu biti bilo koji uređeni skupovi, ne samo cijeli brojevi. Posebno, indeksi izraza mogu biti bilo koji drugi izrazi.”

“8. Način da se opiše tok programa odvojeno od naredbi programa.”

“9. Kompajliranje naredbi koje koriste liste i tablice”

“10. Mogućnost definiranja funkcija i drugih otvorenih i zatvorenih potprograma na moćan način”

“11. Mogućnost referiranja na strojne registre.”

“12. Mogućnost kompajliranja naredbi koje mijenjaju druge naredbe.”

“13. Mogućnost kompajliranja interpretera i pisanja koda za interpretere”

“14. Mogućnost definiranja vlastitih tipografskih konvencija, uključujući i mogućnost da se definira što se radi kad ništa nije naređeno.”

“Biblioteka na magnetskoj traci sadrži ne samo otvorene i zatvorene potprograme, nego i niz definicija za uvođenje novih vrsta kvaniteta ...”

Kvantiteti mogu biti: brojevi sa višestrukom preciznošću, kompleksni brojevi, kvaternioni, vektori, Cliffordovi brojevi, funkcije reprezentirane

na neki način, tablicama, formulama, nizovima koeficijenata, općenitije, elementi prostora funkcija, stringovi, liste ... ti kvantiteti neće biti definiran “u sistemu”, nego kao biblioteke.

Simboli su nizovi slova i brojeva, pri čemu je prvi znak slovo. Služe za reprezentaciju kvantiteta.

Izrazi mogu biti složeni, kao u Fortranu. (Nema specifičnosti kasnije uvedenih u Lisp).

Pseudo-funkcije - funkcije, samo što vrijednost funkcije ne ovisi samo o vrijednosti argumenata.

“Programiranje još nije dostiglo stupanj pri kojem sva izračunavanja mogu biti opisana neovisno o činjenici da stroj ima memoriju podijeljenu u numeriranje registre.”

Primjeri pseudo-funkcija:

Ako je $A=(1\ 2\ 3)$ onda

$CAR(A) = 1,$

$CDR(A) = (2\ 3),$

$NAME((1\ 2\ 3)) = A.$

Iskazni kvantiteti i funkcije (ono što bi danas bile logičke vrijednosti i operacije nad njima.)

Složene naredbe za izračunavanje. Primjerice, zamjena vrijednosti simbola X i Y:

X		Y
Y		X

Složene naredbe (općenitiji oblik)

Kvantifikator	Kvantitet	Uvjet	Vrijednost
$I \in (1 \ (2) \ 11)$	$A(I)$		$B(I)+C(I)$
$J \in L$	$A(B(J)+C(J))$	$B(J)>5$	$R(J)*S(J)$
$I \in (1 \text{ to } N)$			L
$K \in (1 \text{ to } M)$	$C(I, K)$		$\sum_{J=1} A(I*J)*B(JK)$

Funkcije sa više vrijednosti. Problem opisa kompozicije.