TECHNICS AND INFORMATICS IN EDUCATION
6th International Conference, Faculty of Technical Sciences, Čačak, Serbia, 28–29th May 2016
TEHNIKA I INFORMATIKA U OBRAZOVANJU
6. međunarodna konferencija, Fakultet tehničkih nauka, Čačak, Srbija, 28–29. maj 2016.

UDK: 004.41/.42:519.68 Pregledni naučni rad

Izračunljive funkcije i lambda račun

Marko Stanković[1]
[1] Univerzitet u Nišu, Pedagoški fakultet u Vranju, Vranje, Srbija
e-mail markos@ucfak.ni.ac.rs

Rezime: *Lambda račun jedna je od najpoznatijih formalizacija efektivnog postupka sa velikom primenom u funkcionalnom programiranju. S tim u vezi, glavni cilj ovog rada je da pokaže način interpretacije nekih izračunljivih funkcija lambda termima. Zato je u radu dat poseban osvrt na interpretaciju Bulovskih funkcija, Čerčovih numerala i najbitnijih aritmetičkih operacija na njima. Takođe, dat je pregled nekih kombinatora koji su se pokazali korisnim u radu sa lambda termima. Na kraju, izložena je ideja dokaza da je klasa lambda izračunljivih funkcija ekvivalentna klasi primitivno rekurzivnih funkcija.*

Ključne reči: *lambda račun;lambda definabilnost; Čerčovi numerali;izračunljivost*

1. UVOD

λ-račun je skup nekoliko formalnih sistema, zasnovanih na notaciji koju je osmislio Alonzo Čerč tridesetih godina prošlog veka. Osmišljen je da opiše najjednostavnije načine kako se operatori ili funkcije mogu kombinovati radi dobijanja novih operatora.

U praksi, svaki λ-sistem ima beznačajno različitu gramatičku strukturu, u zavisnosti od svoje namene. Neki imaju dodatne simbole konstanti, a većina je dobijena sintaksičkim ograničenjima (na primer, restrikcijom na tačno određene tipove).

U razvoju teorije algoritama došlo se do više vrsta formalizacije efektivnog postupka. Najpoznatije formalizacije, pored λ-računa, svakako su Tjuringove mašine, neograničena registarska mašina (URM), primitivno rekurzivne funkcije, μ-rekurzivne funkcije, Postovi sistemi itd. Iako su oni naizgled međusobno potpuno različiti, svi pristupi određuju jednu istu klasu funkcija.

1.1. Primitivno rekurzivne funkcije

Za funkciju $f: \mathbb{N}^k \to \mathbb{N}$ kažemo da je izračunljiva akko postoji efektivna procedura, takva da za proizvoljnu k -torku $(x_1, \dots, x_k)$ prirodnih brojeva daje vrednost $f(x_1, \dots, x_k)$ (Enderton, 2002:209).

Definicija. Klasa primitivno rekurzivnih funkcija sadrži osnovne funkcije:

- nula funkciju $z(n) = 0$, za svako $n \in \mathbb{N}$,
- funkciju sledbenika prirodnog broja n, $s(n) = n + 1$ i
- funkcije projekcije $u_i^n(x_1, \dots, x_n) = x_i$, $1 \leq i \leq n$,

i sve funkcije koje se od njih dobijaju konačnim brojem primena osnovnih operacija:

- *kompozicije*, tj. ako su već definisane funkcije $g: \mathbb{N}^m \to \mathbb{N}$ i $h_1, \dots, h_m: \mathbb{N}^k \to \mathbb{N}$, definisana je i funkcija $f: \mathbb{N}^k \to \mathbb{N}$ tako da je za sve $(x_1, \dots, x_k) \in \mathbb{N}^k$:

$f(x_1, \dots, x_k) = g\big(h_1(x_1, \dots, x_k), \dots, h_m(x_1, \dots, x_k)\big)$ i

- *primitivne rekurzije*, tj. ako su već definisane funkcije $g: \mathbb{N}^m \to \mathbb{N}$ i $h: \mathbb{N}^{m+2} \to \mathbb{N}$, definisana je i funkcija $f: \mathbb{N}^{m+1} \to \mathbb{N}$ tako da je za sve $(x_1, \dots, x_m) \in \mathbb{N}^m$:

$$f(0, x_1, \dots, x_m) = \quad g(x_1, \dots, x_m),$$
$$f(n+1, x_1, \dots, x_m) = \quad h(f(n, x_1, \dots, x_m), n, x_1, \dots, x_m) \text{ za } n \in \mathbb{N}.$$

Ograničenje ove klase jeste zatvorenost za operaciju minimizacije, koja mora biti ograničena. "Eliminacijom ovog ograničenja (uvođenjem operacije neograničene minimizacije) dobija se klasa *parcijalno rekurzivnih funkcija*, koja se poklapa sa npr. Tjuring-izračunljivim funkcijama" (Ognjanović, Krdžavac, 2004: 27).

2. ČERČOVI NUMERALI, DEFINABILNOST

2.1. Prirodni brojevi

λ-račun je teorija funkcija i algoritama, pa je jedini mogući način za interpretiranje brojeva u λ-računu posmatranje istih kao da su algoritmi. Međutim, broj nije algoritam, već podatak. Iako se situacija čini nemogućom, moguće je posmatrati broj kao algoritam. Na primer, broj 2 možemo „izgraditi“ tako što ćemo na nulu dva puta primeniti funkciju sledbenika. Odgovarajući λ-term koji opisuje ovu proceduru je $\lambda fx.f(fx)$. Bitno je napomenuti da je algoritam nezavistan od toga šta su zapravo nula i funkcija sledbenika.

Dakle, prirodni broj n (i nula), u oznaci $\underline{n}$, interpretira se kao $\underline{n} \equiv \lambda fx.f^n x$, gde f^n definišemo na sledeći način: $f^0 x = x$; $f^{k+1} x = f(f^k x)$. Ovakvo kodiranje prirodnih brojeva osmislio je Alonzo Čerč (engl. Alonzo Church) i termi su po njemu nazvani *Čerčovi numerali*. Čerčovo kodiranje jedan je od mnogo mogućih načina (poznato je još i Mogensen – Skot kodiranje, koje je dobilo naziv po matematičarima Torbenu Mogensenu (engl. Torben Mogensen) i Dejniju Skotu (engl. Dana Scott)). Tehnika da se podaci posmatraju kao algoritmi može se proširiti na sve (induktivno definisane) podatke.

2.2. Lambda definabilnost

Za funkciju koja se može interpretirati λ-termima kažemo da je λ-definabilna i formalno je definišemo na sledeći način.

Definicija (λ-definabilnost). Za parcijalno rekurzivnu funkciju $f: \mathbb{N}^k \to \mathbb{N}$ kažemo da je *λ-definabilna* ako za neki term F važi:

$$f(n_1, \dots, n_k) = m \Rightarrow F\underline{c}_{n_1}, \dots, \underline{c}_{n_k} =_\beta \underline{c}_m$$
$$f(n_1, \dots, n_k) = \uparrow \quad \Rightarrow F\underline{c}_{n_1}, \dots, \underline{c}_{n_k} \text{nema normalnu formu.}$$

Za term F kažemo da λ-definiše funkciju f.

Takođe, važi i

$$f(n_1, \dots, n_k) = m \Leftrightarrow F\underline{c}_{n_1}, \dots, \underline{c}_{n_k} =_\beta \underline{c}_m$$
$$f(n_1, \dots, n_k) = \uparrow \quad \Leftrightarrow F\underline{c}_{n_1}, \dots, \underline{c}_{n_k} \text{nema normalnu formu.}$$

i

$$f(n_1, \dots, n_k) = \downarrow \Rightarrow F\underline{c}_{n_1}, \dots, \underline{c}_{n_k} =_\beta \underline{c}_{f(n_1, \dots, n_k)}.$$

Dakle, u mnogo jednostavnijem slučaju kada $f: \mathbb{N} \to \mathbb{N}$ za f kažemo da je λ-definabilna ako i samo ako postoji λ-term F tako da je $F(\underline{n}) =_\beta \underline{f(n)}$.

2.3. Bulovske vrednosti

Definišimo interpretacije bulovskih vrednosti $true$, $false$ i term koji interpretira if (ako).

$$true \equiv \lambda xy.x \qquad false \equiv \lambda xy.y \qquad if \equiv \lambda pxy.pxy$$

Drugim rečima, $true$ je funkcija dva argumenta koja vraća svoj prvi argument, a $false$ je funkcija koja vraća drugi argument. Term $ifPQR$ treba čitati „ako P onda Q inače R“. Takođe, važi i $ifPQR \rhd_\beta PQR$. Ako je $P = true$ dobija se:

$$if\ trueQR \rhd_\beta trueQR = (\lambda xy.x)QR \rhd_\beta (\lambda y.Q)R \rhd_\beta Q.$$

Slično se dobija $iffalseQR \rhd_\beta R$. Prethodna razmatranja važe za proizvoljan term P tako da $P \rhd_\beta true$ ili $P \rhd_\beta false$.

Na osnovu Čerč-Roserove teoreme (Hindley, Seldin, 2008:14), sledi da je $true \neq false$ jer su $true$ i $false$ različite normalne forme. Konjunkciju, disjunkciju i negaciju interpretiramo na sledeći način:

$$\wedge \equiv \lambda pq.if\ pq\ false \qquad \vee \equiv \lambda pq.if\ p\ true\ q \qquad \neg \equiv \lambda p.if\ p\ false\ true$$

Definicije se mogu proveriti jednostavnim računom. Na primer, $\wedge\, true\, true \rhd_\beta true$. Često se umesto oznake $true$ koristi $\mathbf{1}$, a umesto $false$ koristi se oznaka $\mathbf{0}$. „Treba naglasiti da je bulovska vrednost terma **0** ista kao i numerala $\underline{0}$, dok se bulovska vrednost **1** razlikuje od numerala $\underline{1}$“ (Krivine, 1993:32).

3. ARITMETIKA NA ČERČOVIM NUMERALIMA

Definišimo interpretacije funkcija za računanje sa Čerčovim numeralima.

Funkciju *sabiranja* definišemo kao: $add \equiv \lambda mnfx.mf(nfx)$. Proverimo definiciju sabiranja na sledeći način:

$$\begin{aligned}
add\ \underline{m}\underline{n} \equiv\ & \big(\lambda mnfx.mf(nfx)\big)\underline{m}\underline{n} \rhd_\beta \big(\lambda nfx.\underline{m}f(nfx)\big)\underline{n} \rhd_\beta \\
\rhd_\beta\ & \big(\lambda fx.\underline{m}f(\underline{n}fx)\big) \equiv \lambda fx.(\lambda gx.g^m x)f\big((\lambda hx.h^n x)fx\big) \rhd_\beta \\
\rhd_\beta\ & \lambda fx.(\lambda gx.g^m x)f\big((\lambda x.f^n x)x\big) \rhd_\beta \lambda fx.(\lambda gx.g^m x)f(f^n x) \rhd_\beta \\
\rhd_\beta\ & \lambda fx.(\lambda x.f^m x)(f^n x) \rhd_\beta \lambda fx.f^m(f^n x) \equiv \lambda fx.f^{m+n}x \equiv \underline{m+n}
\end{aligned}$$

Funkciju *množenja* definišemo kao: $mult \equiv \lambda mnfx.m(nf)x$. Proverimo definiciju množenja na sledeći način:

$$\begin{aligned}
mult\ \underline{m}\underline{n} \equiv\ & (\lambda mnfx.m(nf)x)\underline{m}\underline{n} \rhd_\beta (\lambda nfx.\underline{m}(nf)x)\underline{n} \rhd_\beta \lambda fx.\underline{m}f(\underline{n}f)x \\
\equiv\ & \lambda fx.(\lambda gx.g^m x)(\underline{n}f)x \rhd_\beta \lambda fx.(\underline{n}\,f)^m x \equiv \lambda fx.\big((\lambda hx.h^n x)\,f\big)^m x \\
\rhd_\beta\ & \lambda fx.(\lambda x.f^n x\,)^m x \rhd_\beta \lambda fx.(f^n)^m x \equiv \lambda fx.f^{m\cdot n}x \equiv \underline{m \cdot n}
\end{aligned}$$

Stepenovanje definišemo termom $expt \equiv \lambda mnfx.nmfx$.

Funkcija *sledbenika* prirodnog broja definiše se termom $succ \equiv \lambda nfx.f(nfx)$. Funkcija $succ$ definisana je tako da važi redukcija i $succ\ \underline{n} \rhd_\beta \underline{n+1}$. O tome detaljnije govore Mazzola, Milmeiste i Weissmann, (2005:327).

Funkcija koja ispituje da li je *numeral jednak nuli* interpretira se kao $iszero \equiv \lambda n.n(\lambda x.\mathbf{0})\mathbf{1}$. Funkcija je definisana tako da važe redukcije $iszero\ \underline{0} \rhd_\beta \mathbf{1}$ i $iszero\ \underline{(n+1)} \rhd_\beta \mathbf{0}$.

Funkciju *prethodnik* prirodnog broja možemo definisati kombinacijom nekih funkcija uvedenih iznad. Najpre formiramo uređeni par $(n, n-1)$ i zatim izdvojimo drugi element kao rezultat. Uređeni par (a, b) u λ-računu može se interpretirati termom $\lambda z.zab$. Iz uređenog para možemo izdvojiti prvi (drugi) element upotrebom funkcije $\mathbf{1}$ ($\mathbf{0}$). Funkcija $\Phi \equiv \left(\lambda pz.z\big(succ(p\mathbf{1})\big)(p\mathbf{1})\right)$ generiše iz uređenog para $(n, n-1)$ (što je kratko označeno sa p), uređeni par $(n+1, n-1)$.

Podizraz $p\mathbf{1}$ izdvaja prvi element iz para p. Novi par formira se korišćenjem ovog elementa, koji je u novom paru uvećan za 1, dok je drugi element samo kopiran u novi par.

Prethodnik broja n dobija se tako što se primeni n puta funkcija Φ na uređeni par $(\lambda z.z\underline{0}\underline{0})$ i zatim se izdvoji drugi element uređenog para: $pred \equiv (\lambda n.n\Phi(\lambda z.z\underline{0}\underline{0})\mathbf{0})$.

Treba naglasiti da je vrednost funkcije $pred$ primenjene na nulu, nula.

Funkciju *oduzimanja* definišemo kao $subtract\ \underline{m}\underline{n} \equiv \underline{n}\ pred\ \underline{m}$. Funkcija će n puta primeniti funkciju $pred$ na interpretaciju broja m, što će dati željeni rezultat.

Na primer, sada možemo interpretirati funkciju prirodnih brojeva f u λ-računu. Neka je funkcija f zadata sa

$$f(n,m) = \begin{cases} 2+4m, & n = 2, n = 0, \\ n+5m, & \text{inače}, \end{cases}$$

onda je odgovarajući λ-term[1]:

$$\lambda nm.if\left(\vee\left(iszero\big(pred(pred\ n)\big)\right)(iszero\ n)\right)\big(add\ \underline{2}(mult\ m\ \underline{4})\big)\big(add\ n(mult\ \underline{5}\ m)\big).$$

Funkcije za testiranje *jednakosti* i *nejednakosti*. Funkcija koja testira da li je broj x veći ili jednak od broja y definiše se kao $G \equiv \big(\lambda xy.iszero(x\ pred\ y)\big)$. Ova funkcija primenjuje x puta funkciju prethodnika na y i ako je rezultat nula, odatle sledi da je $x \geq y$. Ako je $x \geq y$ i $y \geq x$, onda je $x = y$. Ovo nas dovodi do definicije funkcije E koja testira jednakost dva broja: $E \equiv \left(\lambda xy.\wedge\left(iszero(x\ pred\ y)\right)\big(iszero(y\ pred\ x)\big)\right)$. Na sličan način mogu se definisati interpretacije funkcija $x > y$, $x < y$ ili $x \leq y$.

4. IZRAČUNLJIVE FUNKCIJE U LAMBDA RAČUNU

4.1. Kombinatori

Kombinator je λ-term bez slobodnih promenljivih. Intuitivno, kombinatore možemo shvatiti kao „potpuno određene operacije“, jer oni nemaju slobodne promenljive.

Definicija: Za term Q kažemo da je fiksna tačka terma M ako važi da je $MQ =_\beta Q$.

Tabela 1. *Kombinatori*

$K \equiv \lambda x.(\lambda y.x)$	Kombinator koji formira konstantnu funkciju
$B \equiv \lambda x.\left(\lambda y.\left(\lambda z.x(yz)\right)\right)$	Kombinator koji komponuje dve funkcije

[1] Zbog jednostavnosti zapisa $\vee$ nije zamenjeno ogovarajućim termom. Slično važi i za funkciju E u nastavku.

$S \equiv \lambda x.\left(\lambda y.\left(\lambda z.(xz)(yz)\right)\right)$	Operator jače kompozicije
$C \equiv \lambda x.\left(\lambda y.(\lambda z.xzy)\right)$	Kombinator koji menja mesta argumentima
$Y \equiv \lambda f.\left(\lambda x.f(xx)\right)\left(\lambda x.f(xx)\right)$	Karijev kombinator
$\Theta \equiv \left(\lambda x.\left(\lambda f.\left(f(xxf)\right)\right)\right)\left(\lambda x.\left(\lambda f.\left(f(xxf)\right)\right)\right)$	Tjuringov kombinator

Karijev kombinator Y ima svojstvo da se YX i $X(YX)$ redukuju na isti term. Tjuringov kombinator ima svojstvo da se za svaki λ-term X i ΘX redukuju na term $X(\Theta X)$. Za detalje videti Hindley i Seldin, (2008:34).

Teorema (teorema o fiksnoj tački): Svaki λ-term ima najmanje jednu fiksnu tačku.

Dokaz: Potrebno je dokazati da za svaki term F postoji term X tako da je $FX = X$. Neka je $W \equiv \lambda x.F(xx)$ i $X \equiv WW$. Tada je $X \equiv WW \equiv \left(\lambda x.F(xx)\right)W = F(WW) = FX$.

4.2. Izračunljivost i definabilnost

Teorema: Funkcija f je izračunljiva akko je λ-definabilna.

Dokaz: Pokažimo da su inicijalne funkcije, funkcije dobijene kompozicijom i rekurzijom iz inicijalnih funkcija, λ-definabilne funkcije. Inicijalne funkcije možemo interpretirati na sledeći način:

- funkciju $z(n) = 0$, interpretiramo termom $\underline{0}$,
- funkciju $s(x) = x + 1$ interpretiramo funkcijom $succ$,
- funkcije projekcije $u_i^n(x_1, \dots, x_n) = x_i$, $1 \le i \le n$ interpretiramo sa $u_i^n \equiv \lambda x_1, \dots, x_n.x_i$.

Neka je f funkcija sa k promenljivih i neka su $g_1, \dots, g_k$ λ -definabilne funkcije sa n promenljivih. Neka je $h(x_1, \dots, x_n) = f\left(g_1(x_1, \dots, x_n), \dots, g_k(x_1, \dots, x_n)\right)$ funkcija dobijena kompozicijom funkcije f i $g_1, \dots, g_k$. Ako su F, $G_1, \dots, G_k$ termi koji interpretiraju funkcije $f, g_1, \dots, g_k$, tada je funkcija h λ-definabilna i term koji je predstavlja je

$$H \equiv \lambda x_1, \dots, x_n.F\left(G_1(x_1, \dots, x_n), \dots, G_k(x_1, \dots, x_n)\right).$$

Sada ćemo izložiti ideju kako se rekurzija može interpretirati u λ-računu, a detalji dokaza mogu se videti u Hindley i Seldin, (2008:50). Neka je h λ-definabilna funkcija i neka je H njena interpretacija u λ-računu. Term F koji interpretira funkciju f definiše se na sledeći način: $F \equiv \lambda u x_1, \dots x_n.(\boldsymbol{R}(Hx_1, \dots x_n)(\lambda uv.Huvx_1, \dots x_n)u)$, gde je sa $\boldsymbol{R}$ označen rekurzivni kombinator. Ovaj kombinator odlikuje se svojstvom da za sve terme X, Y i k važi $\boldsymbol{R}XY\ \underline{0} \rhd_\beta X$ i $\boldsymbol{R}XY\ \underline{(k+1)} \rhd_\beta Y\underline{k}(\boldsymbol{R}XY\ \underline{k})$.

Neka je Ginterpretacija funkcije g. Tada operaciju minimizacije $\min\limits_m(g(x_1, \dots x_n, m) = 0)$ interpretiramo termom $\left(Y\left(\lambda f.\lambda m.if\left(iszero\left(G(x_1, \dots x_n, \underline{m})\right)\right)\underline{m}\left(f(succ\ \underline{m})\right)\right)\right)\underline{0}$.

Ovim je pokazano da klasa λ-definabilnih funkcija sadrži sve inicijalne funkcije i da je zatvorena za kompoziciju, rekurziju i minimizaciju. Dakle, ova klasa sadrži sve izračunljive funkcije. Izložimo ideju obratnog tvrđenja.

Pretpostavimo da je funkcija f predstavljena λ-termom X. Ako je funkcija f n-arna, pretpostavimo da je njen argument $(x_1, \dots, x_n)$ i zapišimo term $X\underline{x_1} \dots \underline{x_n}$. Zatim se vrše redukcije sve dok ne dostignemo numeral i vraćamo to kao odgovor; inače, funkcija nije definisana. Na osnovu Čerčove teze, sledi da je funkcija izračunljiva.

5. ZAKLJUČAK

Iako je Alonzo Čerč dizajnirao λ-račun kao bazu za konstruktivnu logiku, ova formalizacija postala je jedan od modela izračunavanja funkcija sa ogromnom primenom u računarstvu. Zbog njegove ekstremne ekspresivnosti, moguće je lambda termima izraziti i složene računarske podatke, kao što su brojevi, bulovske vrednosti, liste, binarna stabla i sl, dok je izračunavanje funkcija upravo β-redukcija. Rekurzivne funkcije interpretiraju se na osnovu teoreme o fiksnoj tački i Y kombinatora. O povezanosti λ-računa i programiranja najbolje svedoči korespondencija između λ-računa i programskog jezika ALGOL 60 (Landin, 1965), kao i činjenice da je λ-račun osnova programskog jezika LISP. O osnovama funkcionalnog programiranja i lambda računa govori Michaelson (2011).

LITERATURA

[1] Enderton, H. (2002). *A Mathematical Introduction to Logic* (Second ed). USA: Elsevier.

[2] Hindley, R., Seldin, J. (2008). *Lambda-Calculus and Combinators, an Introduction*. New York: Cambridge University Press.

[3] Krivine, J. (1993). *Lambda-calculus, types and models*, (Translated from french by René Cori). Paris, Ellis Horwood.

[4] Landin, P. (1965). *Correspondance between ALGOL 60 and Church's Lambda-notation: Part I*. Communications of the ACM CACM, 8(2), 89-101. doi:10.1145/363744.363749

[5] Mazzola, G., Milmeiste, G., Weissmann, J. (2005). *Comprehensive Mathematics for Computer Scientists 2*. Berlin: Springer.

[6] Michaelson, G. (2011). *AN INTRODUCTION TO FUNCTIONAL PROGRAMMING THROUGH LAMBDA CALCULUS*. Dover Publications.

[7] Ognjanović, Z., Krdžavac, N. (2004). *Uvod u teorijsko računarstvo*. Beograd – Kragujevac.