

## A lambda-kalkulus és a számítástudomány

Csörnyei Zoltán

Eötvös Loránd Tudományegyetem,  
Programozási Nyelvek és Fordítóprogramok Tanszék  
`csz@inf.elte.hu`

Az 1930-as évek elején Church megalkotott egy függvényfogalmon alapuló formális elméletet, amit a matematika megalapozásának szánt, de két doktorondusz hallgatója bebizonyította, hogy a rendszer inkonzisztens. Ekkor Church ebből a formális elméletből törölte a problémát okozó részeket, lényegében csak a függvényekre vonatkozó rész maradt meg, amit  $\lambda$ -kalkulusnak nevezett.

Ez az elmélet azonban rövid időn belül rendkívül sikeres lett. Kleene bebizonyította, hogy a rendszerrel definiálható függvények osztálya azonos a parciális rekurzív függvények osztályával, és Turing bizonyítást adott arra, hogy a definiálható függvények pontosan a Turing-kiszámítható függvények.

1941-ben Church elkészítette a  $\lambda$ -kalkulus típusos változatát, és az 1940-es évek közepén megjelent számítógépek szinte kivirágoztatták ezeket az elméleteket. Alkalmazásra kerültek a LISP-től kezdve az Algol-on keresztül a mai modern funkcionális nyelvekben, sőt például a C++ template metaprogramozásban is. A  $\lambda$ -kalkulus nem csak a fogalmak, mint például az objektum-elvű programozás fogalmainak pontos leírására használható, hanem mint programozási nyelv, mint az implementáció eszköze, vagy mint egy magasszintű nyelv és a gép közötti interfész jelenik meg napjainkban is.

### Hivatkozások

- [1] Henk Barendregt: The Impact of the Lambda Calculus in Logic and Computer Science, *The Bulletin of Symbolic Logic*, **3**, No. 2. (1997), 181–215.
- [2] Felice Cardone, J. Roger Hindley: *History of Lambda-calculus and Combinatory Logic*, Swansea University, Mathematics Department, Research Report No. MRRS-05-06 (2006), 1–93.
- [3] Csörnyei Zoltán: *Lambda-kalkulus*, Typotex, Budapest, 2007.