On Turing Completeness of One Minimal Set of Built-in Functions for Functional Programming Languages

G. A. Martirosyan

Chair of Programming and Information Technologies, YSU
E-mail: gevorg.martirosyan@gmail.com

Many functional programming languages operate on *S-expressions*. The sets of built-in functions of those languages contain *car*, *cdr*, *cons*, *atom*, *eq*, *if_then_else* functions. It is shown that Turing computable functions defined on *S-expressions* can be presented in such functional programming languages which have *car*, *cdr*, *cons*, *atom*, *eq*, *if_then_else* built-in functions. In other words, if the set of built-in constants of a functional programming language contains all these functions, then that language is Turing complete. The following two results are obtained for the minimality of the set of built-in functions $\Phi = \{car, cdr, cons, atom, eq, if\_then\_else\}$.

1. Φ is minimal for functional programming languages which use more than two atoms.

2. The function *eq* is representable in a functional programming language which uses only two atoms and the set $\Phi \setminus \{eq\}$ of built-in functions; the set of built-in functions $\Phi \setminus \{eq\}$ is minimal for functional programming languages which use only two atoms and it is the only proper subset of the set Φ, which is minimal for such languages.

References

1. S.A. Nigiyan, "Functional Languages", *Programming and Computer Software*, Vol. 17. pp. 290-297, 1992.

2. S.A. Nigiyan, "On interpretation of functional programming languages", *Programming and Computer Software*, Vol. 19. pp. 71-78, 1993.

3. L.E. Budaghyan, "Necessary and sufficient condition of completeness of computation rule for strongly typed functional programs", Proceedings of the Conference on Computer Science and Information Technologies (CSIT-2005), Yerevan, 2005, p. 16-19