The use of multiple representation for math education

Yusuf Tayfun Tepedelenren a*, Department of Mathematics, Yildiz Technical University, Davutpasa, 34210, Istanbul, Turkey.
Tahsin Dizren b, Departments of Mathematics, Bogazici University, 34210, Istanbul, Turkey.
Tugsan Soltan c, Departments of Mathematics, Arel Collage, 34210, Istanbul, Turkey.
Melike Yagmur Tepedelenren d, Profesor Doktor Fuat Sezgin Fen Lisesi, 342100, Istanbul, Turkey.

Suggested Citation:

Selection and peer review under responsibility of Prof. Dr. Jesus Garcia Laborda, University of Alcala, Spain. ©2017 SciencePark Research, Organization & Counseling. All rights reserved.

Abstract

The Math lessons for students have always been difficult. This may be caused by thinking capabilities of students or the lack of general method for teaching Math. In this study we develop a kind of algebraic approach to some basic arithmetical operations using base functions. First of all we define the base functions. Using these base functions we will develop a class of Computable Functions in terms of given base functions. We will see that it is possible to derive new knowledge from the old ones.

Keywords: Conditional form; recursive functions; base functions; formalism; effectively computable functions.
1. Introduction

This study is originally developed in order to study some mathematical aspects of effectively computable functions. This formalism has provided a strong foundations for the family of functionally expressed functions (Myers, 1988; Tepedeldiren, Özkan & Unal, 2011). We define these functions recursively. In our opinion it is important to build a strong bridge between recursive function theory and these functionally expressed forms. We use a method alike Alonzo Church’s; a kind of a way of effectively computable functions. In this study we mainly use conditionals forms given by John McCarthy (1967). Finally we present a formalism for describing functions which are computable in terms of given base functions (in this study functionals) (Tepedeldiren, 1996).

2. Methodology

Now we are ready to construct the class of computable functions in terms of given base functions. In order to do this we must define the main elements of this class and how to evaluate the conditional forms. Finally we construct the class as we desired. In this class of functions (mainly made up of basic arithmetical operations) the reader can see many of the aspects of these functions in a unique formula.

2.1. The main elements of conditional forms

The essential parts of conditional forms are given in the following:

\( \perp \) : undefinable process or structure

\( \uparrow \) : not ending process (emptiness)

\( e \) : constant

\( p \) : proposition

\( \rightarrow \) : corresponds to...

\( (p \rightarrow e) \) : simple conditional form

\( (p_1 \rightarrow e_1, p_2 \rightarrow e_2, \ldots, p_n \rightarrow e_n) \) : multi-conditional form

2.2. The evaluation of the conditional forms

(a) When given a simple conditional form as \( (p \rightarrow e) \), it is evaluated in the following manner

\[
(p \rightarrow e) = \begin{cases} 
  e & \text{if the proposition } p \text{ is true} \\
  \uparrow & \text{if the proposition } p \text{ is false} \\
  \perp & \text{for other all situations}
\end{cases}
\]

(b) If we evaluate the multi-conditional form \( (p_1 \rightarrow e_1, p_2 \rightarrow e_2, \ldots, p_n \rightarrow e_n) \), we follow the steps given below:

\[
(p_1 \rightarrow e_1, p_2 \rightarrow e_2, \ldots, p_n \rightarrow e_n) = \begin{cases} 
  e_1 & \text{if the proposition } p_1 \text{ is True} \\
  (p_2 \rightarrow e_2, \ldots, p_n \rightarrow e_n) & \text{if } p_1 \text{ is False} \\
  \perp & \text{for other all situations}
\end{cases}
\]
2.3. The design of integer functions using conditional forms

We develop a class of recursively definable functions on the set of non-negative integers. The notation \(\mathbb{I} = \{0, 1, 2, \ldots\}\) is the set of non-negative integers. Using successor and equality functions, we develop a class of functions \(\mathcal{C}\).

**Definition 1**: For \(n \in \mathbb{I}\) we define the successor function as \(\text{succ}(n) = n + 1\) and show it as \(n^+\).

**Definition 2**: For \(n_1, n_2 \in \mathbb{I}\) we define the equality function as \(\text{equ}(n_1, n_2) = (n_1 = n_2 \rightarrow T, T \rightarrow F)\).

We call both functions as base functions. We are ready to construct other functions using them. Firstly we define the predecessor function which find the one less than the given \(n \in \mathbb{I}\).

**Definition 3**: Let \(\mathbb{I}^+ = \{1, 2, \ldots\}\) and \(n \in \mathbb{I}^+\). We define the predecessor of \(n\) as

\[
\text{pred2}(n, 0) = \text{pred}(n) = n^-
\]

\[
\text{pred2}(n, m) = (m^+ = n \rightarrow m, T \rightarrow \text{pred2}(n, m^+))
\]

**Definition 4**: Let \(m, n \in \mathbb{I}\). We define the basic arithmetic operations given below:

(a) \(m + n = (n = 0 \rightarrow m, T \rightarrow m^+ \rightarrow n^-)\)

(b) \(m \cdot n = (n = 0 \rightarrow 0, T \rightarrow m + m \cdot n^-)\)

(c) for \(m \geq n\), \(m - n = (n = 0 \rightarrow m, T \rightarrow m^- \rightarrow n^-)\)

(d) \(n! = (n = 0 \rightarrow 1, T \rightarrow n \cdot (n-1)!\)

**Definition 5**: Let \(m, n \in \mathbb{I}\). The truth of the relation \(m \leq n\) can be found by the formula,

\[
m \leq n = ((m = 0) \lor (\neg (n = 0) \land (m^+ \leq n^-)))
\]

**Definition 6**: Let \(m, n \in \mathbb{I}\). The truth of the relation \(m < n\) can be found by the formula,

\[
m < n = (m \leq n) \land (\neg (m = n))
\]

**Definition 7**: Let \(m, n \in \mathbb{I}^+\)

(a) Integer valued division of \(m/n\) is given as

\[
m / n = ((m < n) \rightarrow 0, T \rightarrow ((m - n)/n)^+)\]

(b) The remainder from this division is given as

\[
\text{rem}(m/n) = (m < n \rightarrow 0, T \rightarrow \text{rem}((m - n)/n))
\]

**Example 1**:

\[
3 + 2 = (\text{if } 2 = 0 \rightarrow 3, D \rightarrow 4 + 1) = 4 + 1 = 5
\]

\[
4 + 1 = (\text{if } 1 = 0 \rightarrow 4, D \rightarrow 5 + 0) = 5 + 0 = 5
\]

\[
5 + 0 = (\text{if } 0 = 0 \rightarrow 5, \ldots \ldots) = 5
\]

The result is 5. What you see in this example is easy to understand. All of the properties of the summation are seen in this formula.
3. Conclusion

In this study, we have tried to explain how to construct a family of Computable Functions in terms of given Base Functions. These explanations and definitions are perfect to understand a compact formula and its properties. You see all of the properties of certain arithmetic operations in one formula and you can easily understand them. You can also easily construct new formulae from the old ones. These functions will be functionals since they use functions as their domains. It is a kind of a new algebraic method to have functionals using appropriate functions.

References


