Biçimsel Diller ve Otomata Teorisi

Sunu II

İZZET FATİH ŞENTÜRK

A New Method for Defining Languages - Recursive

- Recursive Definition; a three-step process
  - Specify some basic objects in the set
  - Give rules for constructing more objects in the set from the ones we already know
  - Declare that no objects except those constructed in this way are allowed in the set
- Example: Define the set of positive even integers
- EVEN is the set of all positive whole numbers divisible by 2
- EVEN is the set of all 2n where n = 1 2 3 4 …
- The third method: recursive definition
  - Rule 1, 2 is in EVEN
  - If x is in EVEN, then so is x+2
  - The only elements in the set EVEN are those that can be produced from the two rules above

Proof

- Prove that 14 is in the set EVEN
  - The first definition; Divide 14 by 2 and there is no remainder
  - The second definition; 14 = (2)(7)
  - The third definition; recursive
    - Rule 1, 2 is in EVEN
    - Rule 2, 2+2 is also in EVEN
    - Rule 2, 4+2 is also in EVEN
    - Rule 2, 6+2 is also in EVEN
    - Rule 2, 8+2 is also in EVEN
    - Rule 2, 10+2 is also in EVEN
    - Rule 2, 12+2 is also in EVEN, conclude that 14 is in EVEN

Proof

- Another recursive definition for EVEN
  - Rule 1, 2 is in EVEN
  - Rule 2, If x and y are both in EVEN, then so is x+y

- Show that 14 is in EVEN
  - Rule 1; 2 is in EVEN
  - Rule 2; x=2, y=2, 4 is in EVEN
  - Rule 2; x=2, y=4, 6 is in EVEN
  - Rule 2; x=4, y=4, 8 is in EVEN
  - Rule 2; x=6, y=8, 14 is in EVEN

Recursive Definition

- Recursive defiintion is still harder to use (prove EVEN) than two nonrecursive definitions
  - But it has some advantages!
  - Suppose that we want to prove that the sum of two numbers is EVEN is also a number in EVEN
    - Trivial conclusion from the second recursive definition
    - To prove this from the first definition is decidedly harder!
- Why do we want a recursive definition?
  - How easy other possible definitions are to understand
  - What types of theorems we may wish to prove

Recursive Definition – Positive/Negative Integers

- Define positive integers
  - Rule I, 1 is in INTEGERS
  - Rule 2, If x is in INTEGERS,then so is x+1

- Define integers to also include negative numbers and zero
  - Rule 1, 1 is in INTEGERS
  - Rule 2, If both x and y are in INTEGERS, then so are x+y and x-y
  - ✓1-1 = 0 and for all positive x, 0 - x = -x

Recursive Definition – Positive Real Numbers

- Rule 1, x is in POSITIVE
- Rule 2, If x and y are in POSITIVE, then so are x+y and xy
- Problem
  - There is no smallest positive real number x on which to build the rest of the set

Recursive Definition – Positive Real Numbers

- Try
  - Rule 1, If x is in INTEGERS, "." is a decimal point and y is any finite string of digits, even one that starts with some zeros, then x.y is in POSITIVE
  - Two problems:
    - It does not generate all real numbers (π is not included – infinite length)
    - Definition is not recursive. We did not use known elements of POSITIVE to create new elements of POSITIVE. Instead, we used an element of INTEGERS and a string of digits

Recursive Definition – Positive Real Numbers

- Try
  - Rule 1, 1 is in POSITIVE
  - If x and y are in POSITIVE, then so are x+y, x*y, and x/y
  - Problem:
    - It defines a set, but it is not the set of positive real numbers (check problem 17 at the end of the chapter)

Recursive Definition - Polynomials

- A polynomial is a finite sum of terms, each of which is of the form a real number times a power of x (can be $x^0=1$)
- The set POLYNOMIAL
  - Rule 1, Any number is in POLYNOMIAL
  - The variable x is in POLYNOMIAL
  - If p and q are in POLYNOMIAL, then so are p+q, p-q, (p), and pq

  (pq refers to multiplication, not concatenation)

Recursive Definition - Polynomials

- Show that $3x^2+7x-9$ is in POLYNOMIAL
  - Rule 1; 3 is in POLYNOMIAL
  - Rule 2; x is in POLYNOMIAL
  - Rule 3; (3)(x) is in POLYNOMIAL; call it 3x
  - Rule 3; (3x)(x) is in POLYNOMIAL; call it $3x^2$
  - Rule 1; 7 is in POLYNOMIAL
  - Rule 3; (7)(x) is in POLYNOMIAL
  - Rule 3; $3x^2+7x$ is in POLYNOMIAL
  - Rule 1; -9 is in POLYNOMIAL
  - Rule 3; $3x^2+7x + (-9) = 3x^2+7x-9$ is in POLYNOMIAL

  There are several other sequences that could produce this result!

Advantages of Recursive Definition - Polynomials

- Suppose that we just proved that
  - The derivative of the sum of two functions is the sum of the derivatives

and

$$\frac{d}{dx}[f(x)+g(x)] = \frac{d}{dx}f(x) + \frac{d}{dx}g(x)$$

  - The derivative of the product of two functions is

$$\frac{d}{dx}[f(x)\cdot g(x)] = \frac{d}{dx}[f(x)]\cdot g(x) + f(x)\cdot\frac{d}{dx}[g(x)]$$

✓ We automatically show that we can differentiate all polynomials as soon as we prove that the derivative of a number is 0 and that the derivative of x is 1

Advantages of Recursive Definition - Polynomials

- Conclusions
  - We do not know that the derivative of x^n is nx^{n-1}

  ✓But we do know that it can be calculated for every n

  ✓We can prove that it is possible to differentiate all polynomials <u>without giving the best algorithm to do it</u>

  ✓We can <u>prove certain tasks are possible</u> for a computer to do even if <u>we do not know the best algorithms</u> to do them

  ✓For this reason, recursive definitons are important for computer theory!

More Recursive Definitions

- The set of people who are descended from Henry VIII
  - Rule 1, The children of Henry VIII are all elements of DESCENDANTS
  - Rule 2, If x is an element of DESCENDANTS, then so are x's children
- The definition of factorial
  - Rule 1, $0! = 1$
  - Rule 2, $n! = n*(n-1)!$
- The definitions are recursive because one of the rules used to define the set mentions the set itself.
  - In computer languages, when we allow a procedure to call itself, we refer to the program as recursive

Examples of Recursive Definitions

- $L_1 = x^+ = \{x\ xx\ xxx\ \ldots\}$
  - Rule 1, x is in L_1
  - Rule 2, If w is any word in L_1 then xw is also in L_1
- $L_4 = x^* = \{\Lambda\ x\ xx\ xxx\ \ldots\}$
  - Rule 1, Λ is in L4
  - Rule 2, If w is any word in L_4 then xw is also in L_4
- $L_2 = x^{odd} = \{x\ xxx\ xxxxx\ \ldots\}$
  - Rule 1, x is in L_2
  - Rule 2, If w is any word in L_2 then xxw is also in L_2

Examples of Recursive Definitions

- INTEGERS
  - Rule 1, 1 2 3 4 5 6 7 8 9 are in INTEGERS
  - Rule 2, If w is any word in INTEGERS, then w0 w1 w2 w3 w4 w5 w6 w7 w8 w9 are also words in INTEGERS
- Kleene closure
  - Rule 1, If S is a language, then all the words of S are in *S**
  - Rule 2, Λ is in *S**
  - Rule 3, If x and y are in *S**, then so is their concatenation xy

An Important Language: Arithmetic Expressions

- Define a valid arithmetic expression that can be typed on one line, in a form digestible by computers
  - Σ = {0 1 2 3 4 5 6 7 8 9 + - * / ( )}
  - Not valid
    - (3+5)+6) Unbalanced parentheses
    - 2//8+9) Forbidden substring -)
    - (3+(4-)8) Forbidden substring -)
    - 2)-(4 Close parenthesis before opening
  - Are there more rules?
    - // and */
    - Still more?

An Important Language: Arithmetic Expressions

- Define AE by using a recursive definition rather than a long list of forbidden substrings and parenthesis requirements
  - Rule 1, Any number (positive, negative, zero) is in AE
  - Rule 2, If x is in AE, then so are:
    - (i) (x)
    - (ii) -x (provided x does not already start with a minus sign)
  - Rule 3, If x and y are in AE, then so are:
    - (i) x+y (if the first symbol in y is not + or -)
    - (ii) x-y (if the first symbol in y is not + or -)
    - (iii) x*y
    - (iv) x/y
    - (v) x**y (denotes exponentiation)

An Important Language: Arithmetic Expressions

- The most natural definition
  - We use it to recognize arithmetic expressions in real life
- Valid or not? (2+4)*(7*(9-3)/4)/4*(2+8)-1
  - We do not scan over the string looking for forbidden substrings
  - We do not count the parentheses
  - We break down into its components

Operator Hierarchy

- The definition of AE gives us the possiblity of writing
  - 2+3+4
    - Not ambiguous
  - 8/4/2
    - Ambiguous. It could mean 8/(4/2)=4 or (8/4)/2=1
  - 3+4*5
    - Ambiguous. Adopt operator hierarchy and left-to-right execution
    - Apply Rule 2 and put enough parentheses to avoid confusion if desired
- The ambiguity in 8/4/2 is a problem of **meaning**. There is no doubt that the string is a word in AE. There is only doubt about what it means