→ Lisp began as a pure functional language

Function Programming (Ch.16) [Compatibility Mode] - PowerPoint

Click to add text

John Backus proposed the language *Functional Programming* (FP). But did not succeed.

The languages ML, Haskell, F# gained some popularity.

One fundamental characteristic of programs written in imperative languages is that the programs have a *state* represented by the values assigned to its variables.

The program state keeps changing through the execution of the program.

All readers of the program understand the use of its variables and how the program's state changes through execution. For larger programs this becomes a very difficult task. But Functional programming does not have this issue since they don't have variables concept.

Prof.R.Gururaj CSF301 PPL BITS Pilani, Hyderabad Campus

Click to add notes

Slide 4 of 17

Mathematical functions

A mathematical function is a mapping of members of one set called domain set to another set called range set.

A function definition specifies the domain and range sets, either explicitly, along with the mapping.

The mapping is described by an expression or in some cases by a table.

A mathematical function takes an element from domain set as a parameter and yields an element of range set.

A function maps its parameter to a value or values, rather than specifying the sequence of operations.

Ex: *cube(x) ≡ x*x*x*

Here in this function definition the domain and range both are real numbers.

Prof.R.Gururaj CSF301 PPL BITS Pilani, Hyderabad Campus

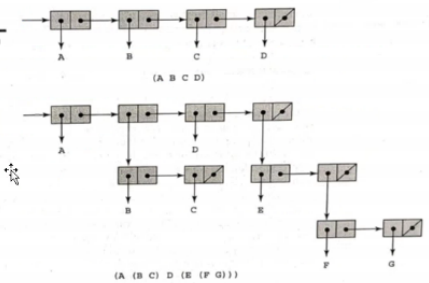
Fundamentals of Functional programming languages.

1. The objective of the design of a functional programming languages is to mimic mathematical functions to the extent possible.
2. This results in a approach to problem solving different than approach used by imperative languages.
3. A purely functional language does not use variables or assignment statements, thus freeing the programmer from the concerns related to memory cell, or state of the program.
4. Without variables iterations not possible. But can be specified with recursion.
5. The execution of a function always gives same results when the parameter is same. This is called referential transparency.

Prof.R.Gururaj CSF301 PPL BITS Pilani, Hyderabad Campus

Click to add text

Figure 1
Internal representation of two LISP lists



(+ 5 7) → void r(5, 7)

Other Functional languages

Scheme is a dialect of LISP. → main use in uni's. other use too.

A scheme program is collection of function definitions.

Lambda expressions are used.

Common LISP is an amalgam of LISP dialects – 1980.

It allows both astatic and dynamic typed variables and includes many imperative features.

ML is static scoped and strongly typed functional language., used syntax close to imperative languages. Includes exception handling, variety of data structures and abstract data types.

Haskell is similar to ML. Unlike Scheme and ML, Haskell is pure FL.

F# is a .NET programming language that supports functional, imperative, an Object Oriented programming.

It can interoperate with other .NET languages and has access to the .NET class library