Concept of Programming Languages (CS320)
Lecture 5

By Zhiqiang Ren
aren@cs.bu.edu
Content

- Recursive multiplication, division, sqrt
- Game of 24
- Randomization on Binary Tree
- Using Cairo
Recursive multiplication, division, sqrt

- **Multiplication using recursion**
  \[ x_1 = 2x_1' + b_1 \]
  \[ x_1x_2 = 2x_1'x_2 + b_1x_2 \]

- **Division**
  \[ x_1 = 2x_1' + b_1 \]
  \[ x_1' = qx_2 + r \]
  \[ x_1 = 2qx_2 + 2r + b_1 \]
  \[ 2r + b_1 < 2x_2 \]
Recursive multiplication, division, sqrt

- $\sqrt{x_1'} - 1 < rt_1 \leq \sqrt{x_1'}$  
  $\left(2rt_1\right)^2 \leq x_1$  
  $\left(2rt_1 + 1\right)^2 = x_1$  
  $\left(2rt_1 + 2\right)^2 > x_1$
Game of 24

- Number: a, b, c, d
- Opr: +, -, *, /
- \(((a \wedge b) \wedge c) \wedge d\)
- \((a \wedge b) \wedge (c \wedge d)\)
- Using exception or option smartly
Randomization on Binary Tree

- Reference in Functional Language
- Generate a random number between 0 and size – 1
- Find the node recursively
- Return the reference to the node instead of the node itself
- Further Application: Conrado Martinez and Salvador Roura. *Randomized Binary Search Trees*
reference

- ref int

“pointer to the object of type reference”
val ref_a = ref_make_elt<int> (3)

- ref (ltree int)
Using Cairo

- `val (pf | ()) = cairo_save (cr), val () = cairo_restore (pf | cr)
- `cairo_translate (cr, x, y)
- `cairo_rotate (cr, theta)
- `cairo_scale (cr, xalpha, yalpha)
- `cairo_set_line_width (cr, width)
- `cairo_arc (cr, xc, yc, rad, M_PI, 2*M_PI)
- `cairo_rectangle (cr, xc, yc, 7.0, 7.0)
- `cairo_move_to (cr, xc, yc), `cairo_line_to (cr, xc, yc)
- `cairo_close_path (cr)
- `cairo_set_source_rgb
- `cairo_fill (cr)
- `cairo_stroke (cr)