

Sixteen Building-Block Primitive Recursive Functions

ECE664
Lecture 6
Avi KAK

① $f(x, y) = x + y$	$f(x, 0) = x$ $f(x, y+1) = f(x, y) + 1$	$f(x, 0) = u_1^1(x)$ $f(x, y+1) = s(u_2^3(y, f(x, y), x))$
② $h(x, y) = x \cdot y$	$h(x, 0) = 0$ $h(x, y+1) = h(x, y) + x$	$h(x, 0) = n(x)$ $h(x, y+1) = f(u_2^3(y, h(x, y), x), u_3^3(y, h(x, y), x))$
③ $g(x) = x!$	$g(0) = 1$ $g(x+1) = (x+1) g(x)$	$g(0) = s(0)$ $g(x+1) = h(s(u_1^2(x, g(x))), u_2^2(x, g(x)))$
④ x^y		
⑤ $p(x) = \begin{cases} x-1 & \text{if } x \neq 0 \\ 0 & \text{if } x=0 \end{cases}$ <i>(predecessor function)</i>	$p(0) = 0$ $p(t+1) = t$	$p(0) = n(0)$ $p(t+1) = u_1^2(t, p(t))$
⑥ $x \ominus y = \begin{cases} x-y & \text{if } x \geq y \\ 0 & \text{if } x < y \end{cases}$	$x \ominus 0 = x$ $x \ominus (t+1) = p(x \ominus t)$	$h(x, 0) = u_1^1(x)$ $h(x, y+1) = p(u_2^3(y, h(x, y), x))$
⑦ $ x - y $ $= (x \ominus y) + (y \ominus x)$		Note that $x \ominus y$ is defined everywhere unlike the $x - y$ for which we wrote a program in S at the end of Lecture 3
⑧ $\alpha(x) = \begin{cases} 1 & x=0 \\ 0 & x \neq 0 \end{cases}$	$\alpha(x) = 1 \ominus x$	← Plays an important role in defining Primitive Recursive Predicates

PRIMITIVE RECURSIVE PREDICATES

- A primitive recursive predicate would obviously be a computable predicate. That is, there is guaranteed to exist a program in S that computes the predicate.
- If P and Q are primitive recursive predicates, so are

$$\begin{aligned} \neg P &\iff \alpha(P) \\ P \& Q &\iff P \circ Q \\ P \vee Q &\iff \neg(\neg P \& \neg Q) \end{aligned}$$

⑨ $x = y$	$d(x, y) = \begin{cases} 1 & x=y \\ 0 & x \neq y \end{cases}$	$d(x, y) = \alpha(x-y)$
⑩ $x \leq y$		$\alpha(x \ominus y)$
⑪ $x < y$		$x \leq y \& \neg(x = y)$

• Say $f(x_1, \dots, x_n) = \begin{cases} g(x_1, \dots, x_n) & \text{if } P(x_1, \dots, x_n) \\ h(x_1, \dots, x_n) & \text{otherwise} \end{cases}$

If g and h are primitive recursive, so is f , since

$$\underline{f = g \cdot P + h \cdot \alpha(P)}$$

• If $f(t, x_1, \dots, x_n)$ is primitive recursive, so are

$$g(y, x_1, \dots, x_n) = \sum_{t=0}^y f(t, x_1, \dots, x_n)$$

$$h(y, x_1, \dots, x_n) = \prod_{t=0}^y f(t, x_1, \dots, x_n)$$

• If the predicate P is primitive recursive, so are the predicates

$$(\forall t)_{\leq y} P(t, x_1, \dots, x_n) \iff \prod_{t=0}^y P(t, x_1, \dots, x_n) = 1$$

$$(\exists t)_{\leq y} P(t, x_1, \dots, x_n) \iff \sum_{t=0}^y P(t, x_1, \dots, x_n) \neq 0$$

<p>(12) $y x$</p> <p>y is a divisor of x</p> <p>$3 12$ is true $3 13$ is false</p>	$y x \iff (\exists t)_{\leq x} (y \cdot t = x)$
---	---

<p>(13) Prime(x)</p> <p>A num is prime if it is greater than 1 and it has only two divisors : 1 and itself.</p>	$\text{Prime}(x) \iff x > 1 \wedge (\forall t)_{\leq x} \left(\begin{array}{l} t=1 \vee \\ t=x \vee \\ \neg(t x) \end{array} \right)$
--	--

MINIMIZATION

(not the same thing as minimization)

Minimalization of a predicate returns the first value of the argument for which the predicate is true. \rightarrow least

The least value of an argument for which a predicate is true can be computed with the help of primitive recursive functions. Consider

$$g(y, x_1, \dots, x_n) = \sum_{u=0}^y \prod_{t=0}^u \alpha(P(t, x_1, \dots, x_n))$$

Under the condition $y \geq t_0$
Try an example with $P(0)=0, P(1)=0, P(2)=1, P(3)=1, \dots$

$$\min_{t \leq y} P(t, x_1, \dots, x_n) = \begin{cases} g(y, x_1, \dots, x_n) & \text{if } (\exists t)_{\leq y} P(t, x_1, \dots, x_n) \\ 0 & \text{otherwise} \end{cases}$$

\leftarrow bounded minimalization

If $f(y, x_1, \dots, x_n) = \min_{t \leq y} P(t, x_1, \dots, x_n)$ then f is primitive recursive

<p>(14) $\lfloor x/y \rfloor$ (integral part of the quotient x/y)</p> <p>$\lfloor 7/2 \rfloor = 3 \quad \lfloor 2/3 \rfloor = 0$</p>	$\lfloor x/y \rfloor = \min_{t \leq x} [(t+1) \cdot y > x]$
---	---

<p>(15) $R(x, y)$ (the remainder when x is divided by y)</p>	$\frac{x}{y} = \lfloor x/y \rfloor + \frac{R(x, y)}{y}$ $R(x, y) = x - (y \cdot \lfloor x/y \rfloor)$
---	--

<p>(16) p_n (for $n > 0$, p_n is the nth prime) We set $p_0 = 0$ to make p_n total</p> <p>$p_0 = 0 \quad p_1 = 2 \quad p_2 = 3$ $p_3 = 5 \quad p_4 = 7 \quad p_5 = 11$</p>	<p>$p_0 = 0$ $p_{n+1} = \min_{t \leq p_n! + 1} [\text{Prime}(t) \wedge t > p_n]$</p> <p>$\frac{p_n! + 1}{p_i} = K + \frac{1}{p_i} \rightarrow$ for $i = 1, 2, \dots, n$</p> <p>\leftarrow The reasoning that justifies this upper bound is cute. Do you get it? A clue</p>
---	--