LOOP programs

- \( P \) is a LOOP program, \( i, j \in \mathbb{N} \)
- These are all LOOP programs:
  - \( x_j := 0 \)
  - \( x_i := x_j + 1 \)
  - \( P; P \)
  - LOOP \( x_j \) DO \( P \) END

- Inputs are \( x_1, x_2, ..., x_k \);
  remaining variables are 0
- Output is \( x_0 \)
- LOOP \( x_j \) DO \( P \) END
  - \( P \) is executed \( x_j \) times
  - \( x_j \) may not be changed in \( P \)
LOOP: Assignment and Decrement

• \( x_j := x_i \)

\[
\begin{align*}
x_j & := 0; \\
& \text{LOOP } x_i \text{ DO} \\
& \quad x_j := x_j + 1 \\
& \text{END}
\end{align*}
\]

• \( x_j := \max\{0, x_i - 1\} \)

\[
\begin{align*}
x_j & := 0; \\
x_k & := 0; \\
& \text{LOOP } x_i \text{ DO} \\
& \quad x_j := x_k; \\
& \quad x_k := x_k + 1 \\
& \text{END}
\end{align*}
\]
LOOP: Addition and Multiplication

• $x_k := x_i + x_j$

\[ x_k := x_i; \]
\[ \text{LOOP } x_j \text{ DO} \]
\[ \quad x_k := x_k + 1 \]
\[ \text{END} \]

• $x_k := x_i \cdot x_j$

\[ x_k := 0; \]
\[ \text{LOOP } x_j \text{ DO} \]
\[ \quad x_k := x_k + x_i \]
\[ \text{END} \]
LOOP: If-zero

IF $x_j \neq 0$ THEN
   $P$
ELSE
   $Q$
END

$x_k := 0$;
LOOP $x_j$ DO
   $x_k := 1$
END;
$x_l := 1$;
LOOP $x_k$ DO
   $P$
   $x_l := 0$
END;
LOOP $x_l$ DO
   $Q$
END
Ackermann function

• $A(0, n) = n + 2$
• $A(m + 1, 0) = 1$
• $A(m + 1, n + 1) = A(m, A(m + 1, n))$

• $A(0, n) = 2 + (n + 1) - 1$
• $A(1, n) = 2(n + 1) - 1$
• $A(2, n) = 2^{n+1} - 1$
• $A(3, n) = 2^{2^{\cdot^{\cdot^{2}}}} - 1$ with $n + 1$ levels

Increasing the first argument by 1 is much larger than any "small" change to the second argument.
Halting problem for LOOP programs is decidable

• Lemma: If LOOP program has input $x_1, x_2, \ldots, x_k$ and the program takes $t$ steps, then the output is at most $t + \max x_i$
  • The only instruction that increases a value is $x_i := x_j + 1$
  • This increases the maximum of all variables by at most 1 per step
  • Output is a variable
Halting problem for LOOP programs is decidable

• Theorem: For any LOOP program $P$, there exists natural $m = m(P)$ such that if given input $x_1, x_2, ..., x_k$, $P$ will halt in $\leq A(m, n)$ steps, where $n = \max\{2, x_1, x_2, ..., x_k\}$
  • $m$ only depends on $P$, not input $x_1, x_2, ..., x_k$

• Corollary: Any LOOP program halts

• Corollary: LOOP program cannot compute $f(n) = A(n, n)$
Halting problem for LOOP programs is decidable

• Claim: take $m = 3 \times \text{number of lines}$
  • $x_i := 0$ and $x_i := x_j + 1$ are each 1 line
  • $P_1; P_2$ has the sum of numbers of lines of $P_1$ and $P_2$
  • LOOP $x_i$ DO $P$ END has 1 more line than $P$

• Structural induction: if $P$ is in the form...

• $x_j := 0$ and $x_j := x_i + 1$: trivially $1 \leq A(3, n)$ step
Halting problem for LOOP programs is decidable

- \( P_1, P_2 \): induct
- \( P_1 \) has \( \ell_1 \) lines, is done in \( \leq A(3\ell_1, n) \) steps
- \( P_2 \) has \( \ell_2 \) lines, has maximum input \( \leq n + A(3\ell_1, n) \)
- \( P_2 \) is done in \( \leq A(3\ell_2, n + A(3\ell_1, n)) \leq A(3\ell_2, A(3\ell_1 + 1, n)) \) steps
- \( m = 3(\ell_1 + \ell_2) \), so \( 3\ell_1, 3\ell_2 \leq m - 3 \)
- \( P_1, P_2 \) is done in \( \leq A(3\ell_1, n) + A(3\ell_2, A(3\ell_1 + 1, n)) \) steps
- \( \leq A(m - 3, n) + A(m - 3, A(m - 2, n)) \)
- \( \leq A(m - 3, n) + A(m - 2, n + 1) \)
- \( \leq 2 A(m - 2, n + 1) \leq A(m, n) \)
Halting problem for LOOP programs is decidable

• LOOP $x_j$ DO $P'$ END

• $P'$ has $\ell'$ lines, is done in $\leq A(3\ell', n)$ steps

• $P'$ is run $x_j$ times
  • First iteration $\leq A(3\ell', n) \leq A(3\ell' + 1, n)$ steps
  • Second iteration $\leq A(3\ell', n + A(3\ell' + 1, n)) \leq A(3\ell' + 1, n + 2)$ steps
  • Third iteration $\leq A(3\ell', n + A(3\ell' + 1, n) + A(3\ell' + 1, n + 2)) \leq A(3\ell' + 1, n + 4)$ steps
  • ...
  • $x_j$-th iteration takes $\leq A(3\ell' + 1, n + 2x_j - 2)$ steps

• Total $\leq \sum_{k=0}^{x_j-1} A(3\ell' + 1, n + 2k)$ steps
Halting problem for LOOP programs is decidable

• LOOP $x_j$ DO $P'$ END, assume $P'$ takes $\leq A(3\ell', n)$ time
• Total $\leq \sum_{k=0}^{x_j-1} A(3\ell' + 1, n + 2k)$ steps
• $x_j \leq n$: total $\leq n \cdot A(3\ell' + 1, 3n)$ steps
• $\leq A(3\ell' + 2, 3n) \leq A(3\ell' + 3, n) = A(m, n)$
Halting problem for LOOP programs is decidable

• Corollary: LOOP program cannot compute \( f(n) = A(n, n) \)
• If there is such program \( P \), it has \( m = m(P) \) so \( P \) halts in \( \leq A(m, n) \) steps
• Give input \( n = m + 2 \)
• \( P \) halts in \( \leq A(m, m + 2) \) steps
• Maximum output \( m + 2 + A(m, m + 2) < A(m + 2, m + 2) \)