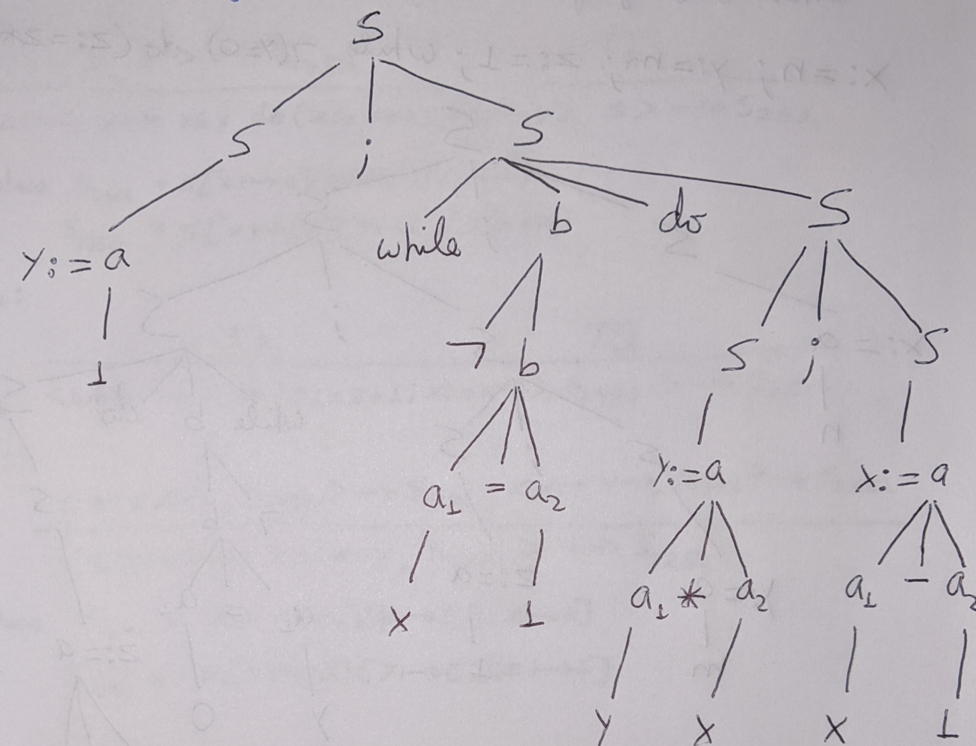
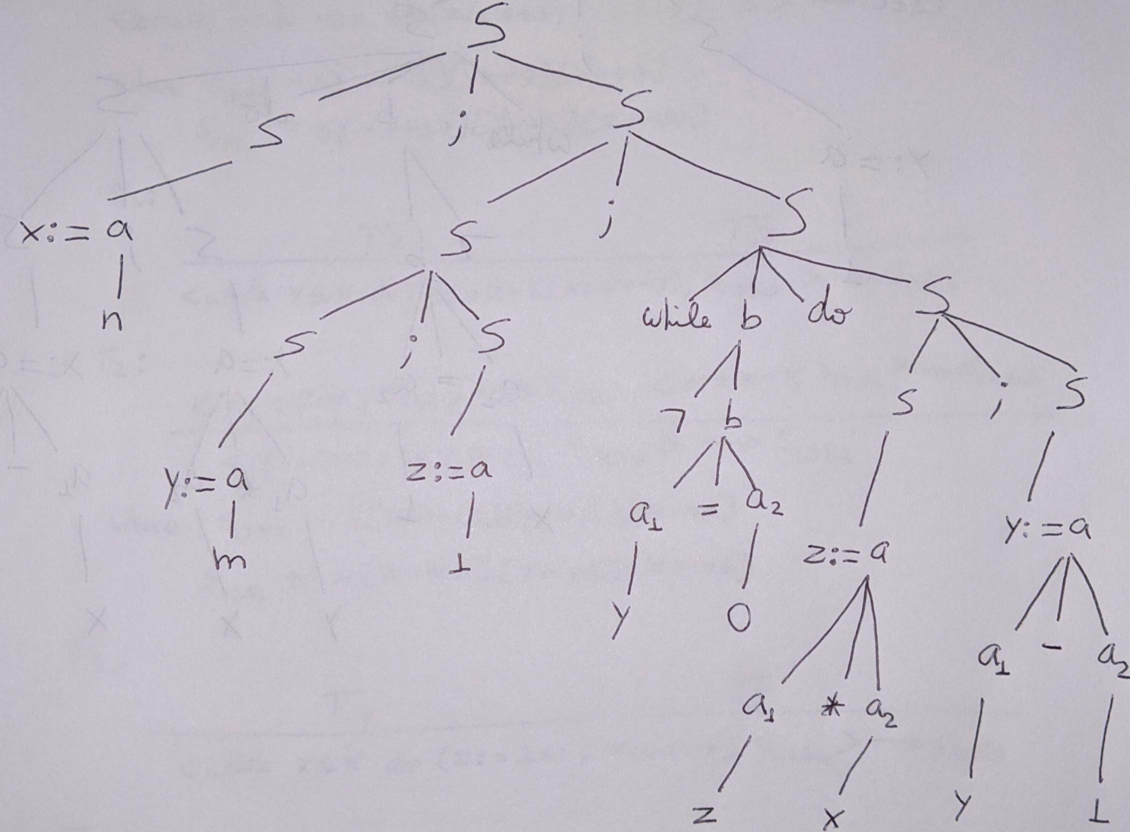
1.1 $y := 1;\ \text{while } \neg(x=1) \text{ do } (y := y * x;\ x := x - 1)$

Abstract syntax tree:



1.2 Assume x: = n; y: = m;
Write a statement that assigns z the value of n to the power of m
Linear and graphical representation:

x: = n; y: = m; z: = 1; while ¬(y = 0) do (z: = z * x; y: = y - 1)



(1.8) Assume that $s\,x = 3$ and determine $\mathcal{B}[\![\neg(x=1)]\!]s$

$$\mathcal{B}[\![\neg(x=1)]\!]s = \quad \mathcal{B}[\![x=1]\!]s = ?$$

$$(\mathcal{A}[\![x]\!]s \; ? \; \mathcal{A}[\![1]\!]s) = ?$$

$$(\; s\,x \; ? \; \mathcal{N}[\![1]\!]\;) = ?$$

$$(\; 3 \neq 1 \;) = ?$$

$$(3 \neq 1) = ff$$

$$\Rightarrow \quad \mathcal{B}[\![\neg(x=1)]\!]s = tt$$

2.3 $z:=0;\ \text{while } y \le x \text{ do } (z:=z+1;\ x:=x-y)$

Assume $s\,x = 17$, $s\,y = 5$

Derivation tree:

$$\frac{\langle z:=0, s\rangle \rightarrow s_{1750} \qquad T_1}{\langle z:=0;\ \text{while } y\le x \text{ do } (z:=z+1;\ x:=x-y), s\rangle \rightarrow s_{253}}$$

where $s_{253} = s[x\mapsto 2][y\mapsto 5][z\mapsto 3]$

$s_{1750} = s[x\mapsto 17][y\mapsto 5][z\mapsto 0]$

T_1:

$$\frac{T_2 \qquad T_3}{\langle \text{while } y\le x \text{ do } (z:=z+1;\ x:=x-y), s_{1750}\rangle \rightarrow s_{253}}$$

T_2:

$$\frac{\langle z:=z+1, s_{1750}\rangle \rightarrow s_{1751}, \quad \langle x:=x-y, s_{1751}\rangle \rightarrow s_{1251}}{\langle z:=z+1;\ x:=x-y, s_{1750}\rangle \rightarrow s_{1251}}$$

where $s_{1751} = s[x\mapsto 17][y\mapsto 5][z\mapsto 1]$

$s_{1251} = s[x\mapsto 12][y\mapsto 5][z\mapsto 1]$

T_3:

$$\frac{T_4 \qquad T_5}{\langle \text{while } y\le x \text{ do } (z:=z+1;\ x:=x-y), s_{1251}\rangle \rightarrow s_{253}}$$

T_4:

$$\frac{\langle z:=z+1, s_{1251}\rangle \rightarrow s_{1252}, \quad \langle x:=x-y, s_{1252}\rangle \rightarrow s_{752}}{\langle z:=z+1;\ x:=x-y, s_{1251}\rangle \rightarrow s_{752}}$$

where $s_{1252} = s[x\mapsto 12][y\mapsto 5][z\mapsto 2]$

$s_{752} = s[x\mapsto 7][y\mapsto 5][z\mapsto 2]$

T_5:

$$\frac{\dfrac{\langle z := z+1, s_{752} \rangle \to s_{753} \quad , \quad \langle x := x-y, s_{753} \rangle \to s_{253}}{\langle z := z+1;\ x := x-y, s_{752} \rangle \to s_{253}} \quad , \quad \langle \text{while } y \le x \text{ do} (z := z+1; x := x-y), s_{253} \rangle \to s_{253}}{\langle \text{while } y \le x \text{ do} (z := z+1;\ x := x-y), s_{752} \rangle \to s_{253}}$$

where $s_{753} = s[x \mapsto 7][y \mapsto 5][z \mapsto 3]$

(2.4) For each statement determine whether or not it always terminates and whether or not it always loops.

- while $\neg(x=1)$ do $(y:=y*x;\ x:=x-1)$

↳ $\boxed{\text{Terminates}}$

for instance, if s $x:=3$; $y:=1$

$$\frac{\langle y:=y*x;\ x:=x-1, s_{13}\rangle \to s_{32},\quad \langle \text{while } \neg(x=1) \text{ do } (y:=y*x; x:=x-1), s_{32}\rangle \to s_{61}}{\langle \text{while } \neg(x=1) \text{ do } (y:=y*x;\ x:=x-1), s_{13}\rangle \to s_{61}}$$

↳ x will get to +1 and it will terminate

- while $1 \leq x$ do $(y:=y*x;\ x:=x-1)$

Same as before, x will be less than 1 and will terminate

↳ $\boxed{\text{terminates}}$

for instance, s $x:=3$; $y:=1$

$$\frac{\langle y:=y*x;\ x:=x-1, s_{13}\rangle \to s_{32},\quad \langle \text{while } 1\leq x \text{ do } (\cdots), s_{32}\rangle \to s_{61}}{\langle \text{while } 1\leq x \text{ do } (y:=y*x;\ x:=x-1), s_{13}\rangle \to s_{61}}$$

- while true do skip

↳ $\boxed{\text{Always loops}}$

$$\frac{\dfrac{\langle \text{skip}, s\rangle \to s,\quad \langle \text{while true do skip}, s\rangle \to s''}{\langle \text{skip}, s\rangle \to s,\quad \langle \text{while true do skip}, s\rangle \to s''}}{\langle \text{while true do skip}, s\rangle \to s''}$$

↳ we keep looping

(2.6) Construct a statement showing that $S_1 ; S_2$ is not, in general, semantically equivalent to $S_2 ; S_1$

Suppose $y := 0$
$\quad\quad\quad\; s\ x = 2$

$$\begin{cases} S_1 \longrightarrow y := 1 \\ S_2 \longrightarrow \text{while } \neg(x=1) \text{ do } (y := y * x;\ x := x-1) \end{cases}$$

$$\dfrac{\langle y:=1, s\rangle \to s_{21} \qquad \dfrac{\dfrac{\langle y:=y*x, s_{21}\rangle \to s_{22}, \quad \langle x:=x-1, s_{22}\rangle \to s_{12}}{\langle y:=y*x;\ x:=x-1, s_{21}\rangle \to s_{12}} \qquad \langle \text{while } \neg(x=1) \text{ do}(\cdots), s_{12}\rangle \to s_{12}}{\langle \text{while } \neg(x=1) \text{ do } (y:=y*x;\ x:=x-1), s_{21}\rangle \to s_{12}}}{\langle y:=1;\ \text{while } \neg(x=1) \text{ do } (y:=y*x;\ x:=x-1), s\rangle \to s_{12}}$$

where $s_{12} = s[x \mapsto 1][y \mapsto 2]$

$$\begin{cases} S_1 \longrightarrow \text{while } \neg(x=1) \text{ do } (y := y * x;\ x := x-1) \\ S_2 \longrightarrow y := 1 \end{cases}$$

$$\dfrac{\dfrac{\dfrac{\langle y:=y*x, s\rangle \to s_{20}, \quad \langle x:=x-1, s_{20}\rangle \to s_{10}}{\langle y:=y*x;\ x:=x-1, s\rangle \to s_{10}} \qquad \langle \text{while } \neg(x=1) \text{ do}(\cdots), s_{10}\rangle \to s_{10}}{\langle \text{while } \neg(x=1) \text{ do } (y:=y*x;\ x:=x-1), s\rangle \to s_{10}} \qquad \langle y:=1, s_{10}\rangle \to s_{11}}{\langle \text{while } \neg(x=1) \text{ do } (y:=y*x;\ x:=x-1);\ y:=1, s\rangle \to s_{11}}$$

where $s_{11} = s[x \mapsto 1][y \mapsto 1]$

$\Rightarrow S_1 ; S_2$ is not semantically equivalent to $S_2 ; S_1$ in this example

In other terms,

$$\langle S_1, s\rangle \to s_{12} \quad \text{and} \quad \langle S_2, s\rangle \to s_{11}$$

are not the same

(2.11) Transition system for the natural semantics for the arithmetic expressions:

$$\langle n, s \rangle \to_{Aexp} \mathcal{N}[\![n]\!]$$

$$\langle x, s \rangle \to_{Aexp} s\,x$$

$$\frac{\langle a_1, s \rangle \to_{Aexp} z_1, \quad \langle a_2, s \rangle \to_{Aexp} z_2}{\langle a_1 + a_2, s \rangle \to_{Aexp} z} \quad \text{where } z = z_1 + z_2$$

$$\frac{\langle a_1, s \rangle \to_{Aexp} z_1, \quad \langle a_2, s \rangle \to_{Aexp} z_2}{\langle a_1 * a_2, s \rangle \to_{Aexp} z} \quad \text{where } z = z_1 * z_2$$

$$\frac{\langle a_1, s \rangle \to_{Aexp} z_1, \quad \langle a_2, s \rangle \to_{Aexp} z_2}{\langle a_1 - a_2, s \rangle \to_{Aexp} z} \quad \text{where } z = z_1 - z_2$$

2.17 Extend **While** with the construct repeat S until b

structural operational semantics (sos):

$[\text{repeat}_{sos}]$ $\quad \langle \text{repeat } S \text{ until } b,\ s \rangle \Rightarrow$
$\langle S; \text{if } b \text{ then skip else (repeat } S \text{ until } b),\ s \rangle$

natural semantics (ns):

$[\text{repeat}^{tt}_{ns}]$ $\quad \langle S, s \rangle \to s'$ if $\mathcal{B}[\![b]\!]s = tt$

$$[\text{repeat}^{ff}_{ns}] \quad \frac{\langle S, s \rangle \to s', \quad \langle \text{repeat } S \text{ until } b,\ s' \rangle \to s''}{\langle \text{repeat } S \text{ until } b,\ s \rangle \to s''}$$

if $\mathcal{B}[\![b]\!] = ff$

(2.33) $x := -1;\ \text{while } x \leq 0 \text{ do } (x := x-1 \text{ or } x := (-1)*x)$

Given a state s describe the set of final states that may result according to the n.S. and the set of derivation sequences specified by S.O.S.

n.s.:

$$\dfrac{\langle x := -1, s\rangle \to s_{-1} \quad , \quad \dfrac{\dfrac{\langle x := (-1)*x, s_{-1}\rangle \to s_1}{\langle x := x-1 \text{ or } x := (-1)*x, s_{-1}\rangle \to s_1} \quad , \quad \langle \text{while } x \leq 0 \text{ do}(\dots), s_1\rangle \to s_1}{\langle \text{while } x \leq 0 \text{ do } (x := x-1 \text{ or } x := (-1)*x), s_{-1}\rangle \to s_1}}{\langle x := -1;\ \text{while } x \leq 0 \text{ do } (x := x-1 \text{ or } x := (-1)*x), s\rangle \to s_1}$$

where $s_{-1} = s[x \mapsto -1]$

and the final state $s_1 = s[x \mapsto 1]$

↳ This is the case when we decide to execute S_2 in S_1 or S_2 one time.

However, there other final states we ~~can get~~ decide to choose from 1 to many times S_1 and then choose S_2 to get out of the while loop.

↳ $s[x \mapsto 2]$

$s[x \mapsto 3]$

$s[x \mapsto 4]$

$s[x \mapsto 5]$

$\vdots$

S.O.S.:

$\langle x := -1;\ \text{while } x \leq 0 \text{ do } (x := x-1 \text{ or } x := (-1) * x),\ s \rangle$

$\Rightarrow \langle \text{while } x \leq 0 \text{ do } (x := x-1 \text{ or } x := (-1) * x),\ s[x \mapsto -1] \rangle$

$\Rightarrow \langle \text{if } x \leq 0 \text{ then } ((x := x-1 \text{ or } x := (-1) * x);$
$\quad \text{while } x \leq 0 \text{ do } (x := x-1 \text{ or } x := (-1) * x))$
$\quad \text{else skip},\ s[x \mapsto -1] \rangle$

$\Rightarrow \langle x := x-1 \text{ or } x := (-1) * x;\ \text{while } x \leq 0 \text{ do } (\cdots),\ s[x \mapsto 1] \rangle$

$\Rightarrow \langle \text{while } x \leq 0 \text{ do } (x := x-1 \text{ or } x := (-1) * x),\ s[x \mapsto 1] \rangle$

$\Rightarrow \langle \text{if } x \leq 0 \text{ then } ((x := x-1 \text{ or } x := (-1) * x);$
$\quad \text{while } x \leq 0 \text{ do } (x := x-1 \text{ or } x := (-1) * x))$
$\quad \text{else skip},\ s[x \mapsto 1] \rangle$

$\Rightarrow \langle \text{skip},\ s[x \mapsto 1] \rangle$

$\Rightarrow s[x \mapsto 1]$

Other derivation sequence we could get:

$\Rightarrow^* s[x \mapsto 2]$

$\Rightarrow^* s[x \mapsto 3]$

$\Rightarrow^* s[x \mapsto 4]$

$\Rightarrow^* s[x \mapsto 5]$

$\vdots$