Try it out 2.24:

$$fv(x * y) = fv(x) \cup fv(y) = \{x, y\}$$

$$fv(A[i] \underset{op_b}{=} x) = fv(A[i]) \cup fv(x) = \{A\} \cup fv(i) \cup \{x\} = \{A, i, x\}$$

Try it out 2.26:

(a) $q_3, \overset{a_1}{\mathtt{r} < \mathtt{y}}, q_6, \overset{a_2}{\mathtt{out!r}}, q_◂$

(b) $q_3, \underset{a_1}{\mathtt{r} \geq \mathtt{y}}, q_4, \underset{a_2}{\mathtt{r} := \mathtt{r} - \mathtt{y}}, q_5, \underset{a_3}{\mathtt{q} := \mathtt{q} + 1}, q_3, \underset{a_4}{\mathtt{r} < \mathtt{y}}, q_6, \underset{a_5}{\mathtt{out!r}}, q_◂$

a) $r \in Use(a_1) \wedge \forall j < 1: r \notin Def(a_j)$

$y \in Use(a_1) \wedge \forall j < 1: y \notin Def(a_j)$

$\Rightarrow Use(\pi) = \{r, y\}$

b) $r \in Use(a_1) \wedge \forall j < 1: r \notin Def(a_j)$

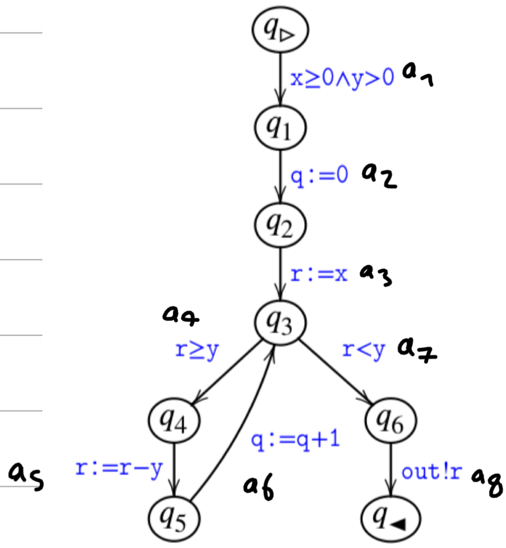
$y \in Use(a_1) \wedge \forall j < 1: y \notin Def(a_j)$

$q \in Use(a_3) \wedge \forall j < 3: q \in Def(a_j)$

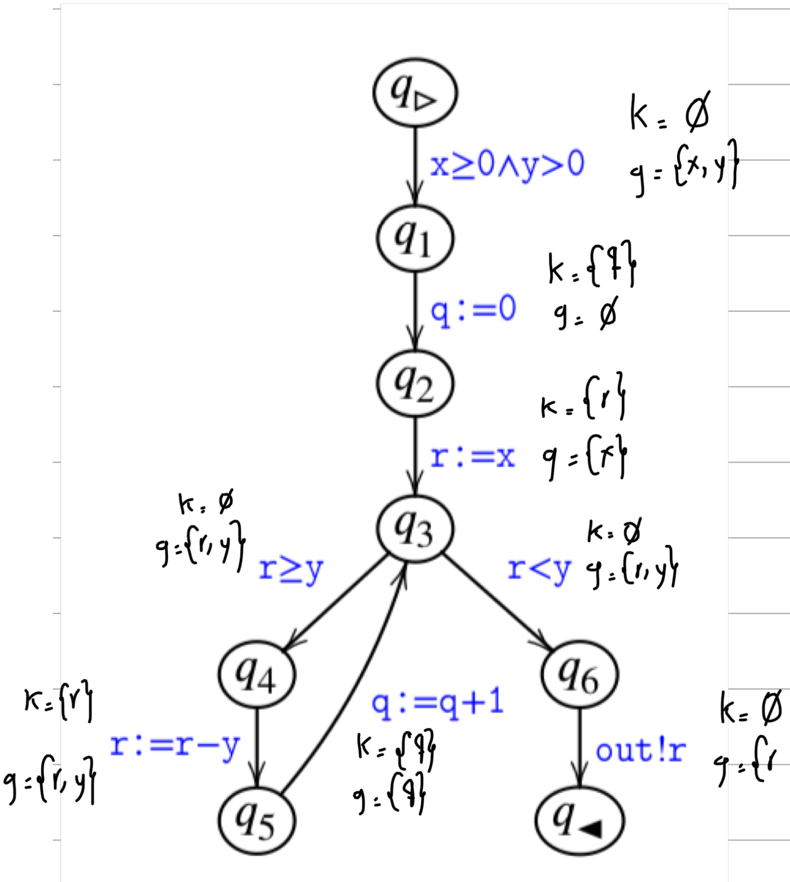
$\Rightarrow Use(\pi) = \{r, y, q\}$

Try it out 2.28:

$\mathsf{LV}(q_\triangleright) \supseteq \{\overset{a_1}{x}, \overset{a_1}{y}\}$		$\mathsf{LV}(q_4) \supseteq \{\overset{a_5}{y}, \overset{a_6}{q}, \overset{a_5}{r}\}$	
$\mathsf{LV}(q_1) \supseteq \{\overset{a_3}{x}, \overset{a_4 \text{ or } a_7}{y}\}$		$\mathsf{LV}(q_5) \supseteq \{\overset{a_7 \text{ or } a_4}{y}, \overset{a_6}{q}, \overset{a_7 \text{ or } a_4}{r}\}$	
$\mathsf{LV}(q_2) \supseteq \{\overset{a_3}{x}, \overset{a_4 \text{ or } a_7}{y}, \overset{a_6}{q}\}$		$\mathsf{LV}(q_6) \supseteq \{r\}\ a_8$	
$\mathsf{LV}(q_3) \supseteq \{\underset{a_4 \text{ or } a_7}{y}, \underset{a_6}{q}, \underset{a_4 \text{ or } a_7}{r}\}$		$\mathsf{LV}(q_\blacktriangleleft) \supseteq \{\ \}$	



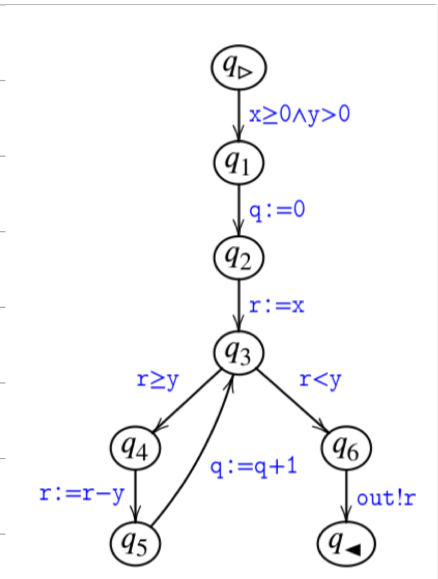
Try it out 2.29:



Exercise 2.31:

Yes, we can see it as a varible. The Use would be the same as for Use(variable).

Try it out 2.32:



$LV(q_6) \supseteq (LV(q_\blacktriangleleft) / \emptyset) \cup \{r\}$

$LV(q_3) \supseteq (LV(q_6) / \emptyset) \cup \{r, y\}$

$LV(q_3) \supseteq (LV(q_4) / \emptyset) \cup \{r, y\}$

$LV(q_4) \supseteq (LV(q_5) / \{r\}) \cup \{r, y\}$

$LV(q_5) \supseteq (LV(q_3) / \{q\}) \cup \{q\}$

$LV(q_2) \supseteq (LV(q_3) / \{r\}) \cup \{x\}$

$LV(q_1) \supseteq (LV(q_2) / \{q\}) \cup \emptyset$

$LV(q_\triangleright) \supseteq (LV(q_1) / \emptyset) \cup \{x, y\}$

Try it out 2.34:

$$
\begin{array}{lcl@{\qquad}lcl}
\mathsf{LV}(q_{\triangleright}) & = & \{\mathtt{x},\mathtt{y}\} & \mathsf{LV}(q_4) & = & \{\mathtt{y},\mathtt{q},\mathtt{r}\} \\
\mathsf{LV}(q_1) & = & \{\mathtt{x},\mathtt{y}\} & \mathsf{LV}(q_5) & = & \{\mathtt{y},\mathtt{q},\mathtt{r}\} \\
\mathsf{LV}(q_2) & = & \{\mathtt{x},\mathtt{y},\mathtt{q}\} & \mathsf{LV}(q_6) & = & \{\mathtt{r}\} \\
\mathsf{LV}(q_3) & = & \{\mathtt{y},\mathtt{q},\mathtt{r}\} & \mathsf{LV}(q_{\blacktriangleleft}) & = & \{\,\}
\end{array}
$$

constraint 1: $\{r\} \supseteq (\{\}/\emptyset) \cup \{r\}$ ✓

constraint 2: $\{y,q,r\} \supseteq (\{r\}/\emptyset) \cup \{r,y\}$ ✓

constraint 3: $\{y,q,r\} \supseteq (\{y,q,r\}/\emptyset) \cup \{r,y\}$ ✓

constraint 4: $\{y,q,r\} \supseteq (\{y,q,r\}/\{r\}) \cup \{r,y\}$ ✓

constraint 5: $\{y,q,r\} \supseteq (\{y,q,r\}/\{q\}) \cup \{q\}$ ✓

constraint 6: $\{x,y,q\} \supseteq (\{y,q,r\}/\{r\}) \cup \{x\}$ ✓

constraint 7: $\{x,y\} \supseteq (\{x,y,q\}/\{q\}) \cup \emptyset$ ✓

constraint 8: $\{x,y\} \supseteq (\{x,y\}/\emptyset) \cup \{x,y\}$ ✓

Exercise 2.35:

$A[a_0]$

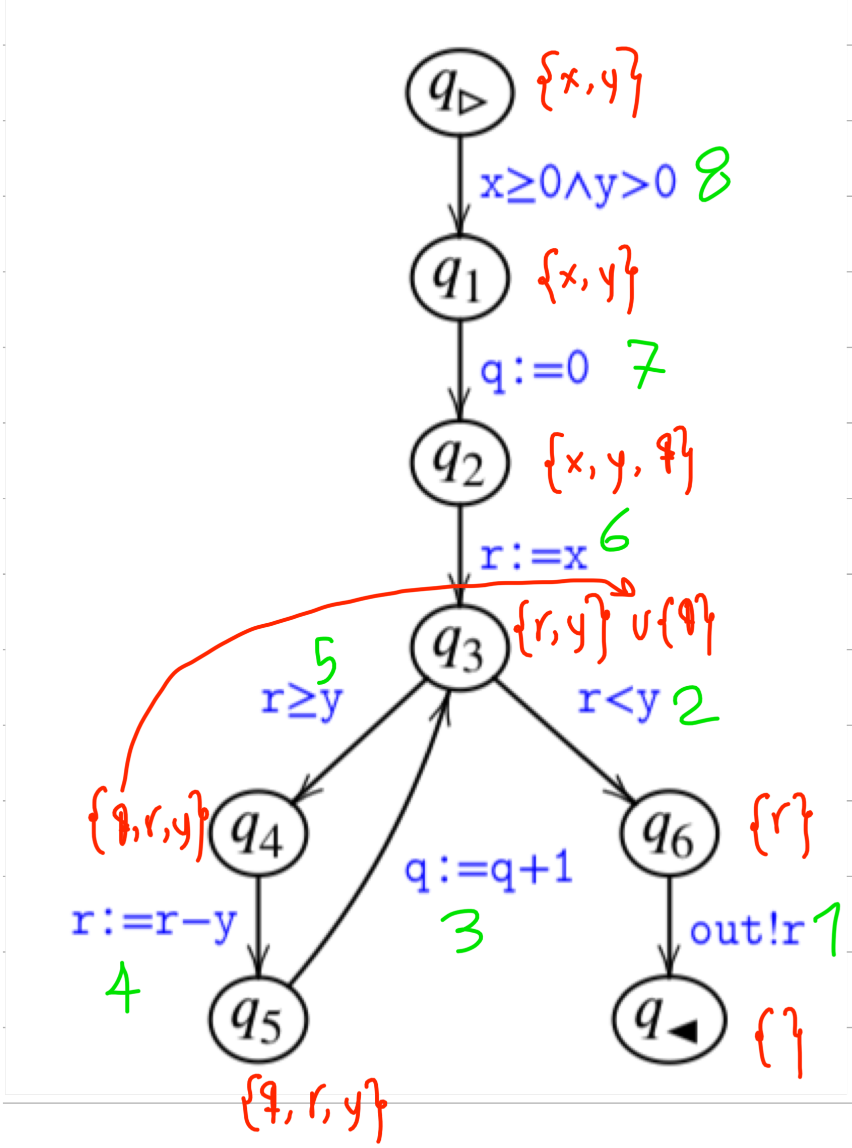
$$
\begin{array}{rcl@{\qquad}rcl}
\mathbf{fv}(n) & = & \{\,\} & \mathbf{fv}(\mathtt{true}) & = & \{\,\} \\
\mathbf{fv}(x) & = & \{x\} & \mathbf{fv}(\mathtt{false}) & = & \{\,\} \\
\mathbf{fv}(A[a_0]) & = & \{A\} \cup \mathbf{fv}(a_0) & \mathbf{fv}(a_1\ op_r\ a_2) & = & \mathbf{fv}(a_1) \cup \mathbf{fv}(a_2) \\
\mathbf{fv}(a_1\ op_a\ a_2) & = & \mathbf{fv}(a_1) \cup \mathbf{fv}(a_2) & \mathbf{fv}(b_1\ op_b\ b_2) & = & \mathbf{fv}(b_1) \cup \mathbf{fv}(b_2) \\
\mathbf{fv}(-a_0) & = & \mathbf{fv}(a_0) & \mathbf{fv}(\neg b_0) & = & \mathbf{fv}(b_0)
\end{array}
$$

The Use will be alright with this change.

α	$\mathsf{kill}_{\mathsf{LV}}(q_\circ,\alpha,q_\bullet)$	$\mathsf{gen}_{\mathsf{LV}}(q_\circ,\alpha,q_\bullet)$
$x := a$	$\{x\}$	$\mathbf{fv}(a)$
$A[a_1] := a_2$	$\{\ \} \to A[a_1]$	$\mathbf{fv}(a_1) \cup \mathbf{fv}(a_2)$
$c?x$	$\{x\}$	$\{\ \}$
$c?A[a]$	$\{\ \} \to A[a]$	$\mathbf{fv}(a)$
$c!a$	$\{\ \}$	$\mathbf{fv}(a)$
b	$\{\ \}$	$\mathbf{fv}(b)$
skip	$\{\ \}$	$\{\ \}$

These changes make the computation more complex and we must keep eyes open for all array entries too.

Try it out 2.36:



Exercise 2.40:

Same as the proof for RD

Try it out 2.45:

$ae(x*y*z) = ae(x*y) \cup \cancel{ae(z)} \cup \{x*y*z\}$

$= ae(x) \cup ae(y) \cup \{x*y\} \cup \{x*y*z\} = \{x*y,\ x*y*z\}$

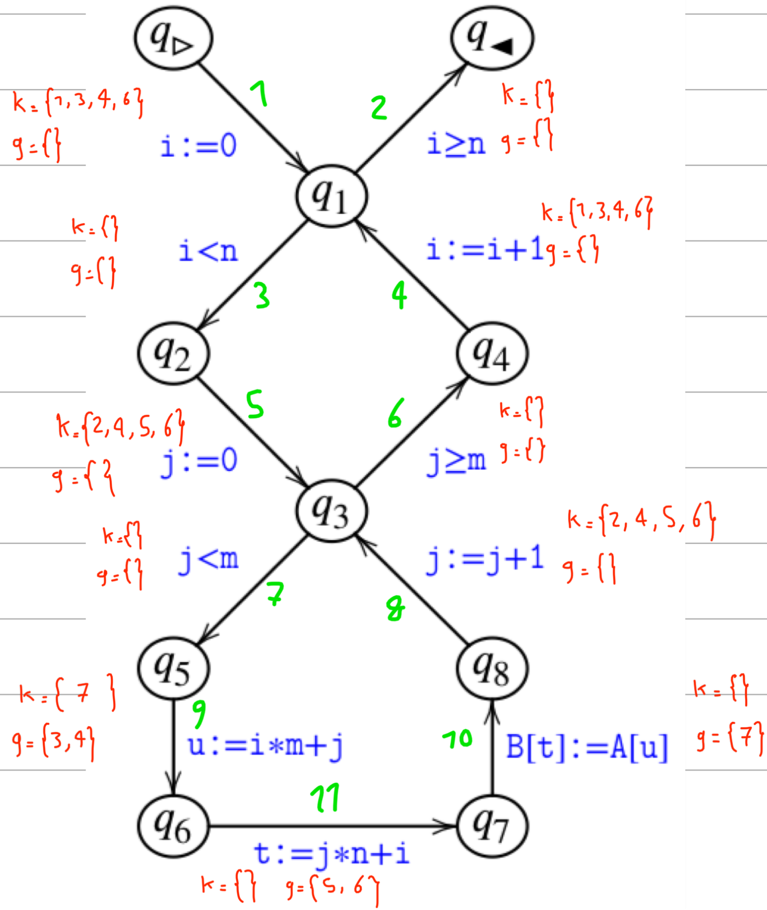
$ae(B[i] + B[i]/10) = ae(B[i]) \cup ae(B[i]/10) \cup \{B[i] + B[i]/10\}$

$= ae(i) \cup \{B[i]\} \cup ae(B[i]) \cup ae(10) \cup \{B[i]/10\} \cup \{B[i] + B[i]/10\}$

$\{B[i],\ B[i]/10,\ B[i] + B[i]/10\}$

Try it out 2.49:

$$\mathrm{AExp} = \{\overset{1}{i+1}, \overset{2}{j+1}, \overset{3}{i*m}, \overset{4}{i*m+j}, \overset{5}{j*n}, \overset{6}{j*n+i}, \overset{7}{A[u]}\}$$



Try it out 2.51:

Constructing all the constraints based on the kill and gen sets written in the try it out 2.49 based on the following rule:

$$(AE(q_\circ) \setminus k) \cup g \supseteq AE(q_\bullet) \quad \text{for each} \quad (q_\circ, a, q_\bullet)$$

Try it out 2.53:

$\mathsf{AE}(q_{\triangleright})$	$=$	$\{\}$	$\mathsf{AE}(q_5)$	$=$	$\{\}$
$\mathsf{AE}(q_1)$	$=$	$\{\}$	$\mathsf{AE}(q_6)$	$=$	$\{\mathtt{i*m}, \mathtt{i*m+j}\}$
$\mathsf{AE}(q_2)$	$=$	$\{\}$	$\mathsf{AE}(q_7)$	$=$	$\{\mathtt{i*m}, \mathtt{i*m+j}, \mathtt{j*n}, \mathtt{j*n+i}\}$
$\mathsf{AE}(q_3)$	$=$	$\{\}$	$\mathsf{AE}(q_8)$	$=$	$\{\mathtt{i*m}, \mathtt{i*m+j}, \mathtt{j*n}, \mathtt{j*n+i}, \mathtt{A[u]}\}$
$\mathsf{AE}(q_4)$	$=$	$\{\}$	$\mathsf{AE}(q_{\blacktriangleleft})$	$=$	$\{\}$

1) $(\{\} \setminus \{1,3,4,6\}) \cup \{\} \supseteq \{\}$ ✓

2) $(\{\} \setminus \{\}) \cup \{\} \supseteq \{\}$ ✓

3) $(\{\} \setminus \{\}) \cup \{\} \supseteq \{\}$ ✓

4) $(\{\} \setminus \{1,3,4,6\}) \cup \{\} \supseteq \{\}$ ✓

5) $(\{\} \setminus \{2,4,5,6\}) \cup \{\} \supseteq \{\}$ ✓

6) $(\{\} \setminus \{\}) \cup \{\} \supseteq \{\}$ ✓

7) $(\{\} \setminus \{\}) \cup \{\} \supseteq \{\}$ ✓

8) $(\{3,4,5,6,7\} / \{2,4,5,6\}) \cup \{\} \supseteq \{\}$ ✓

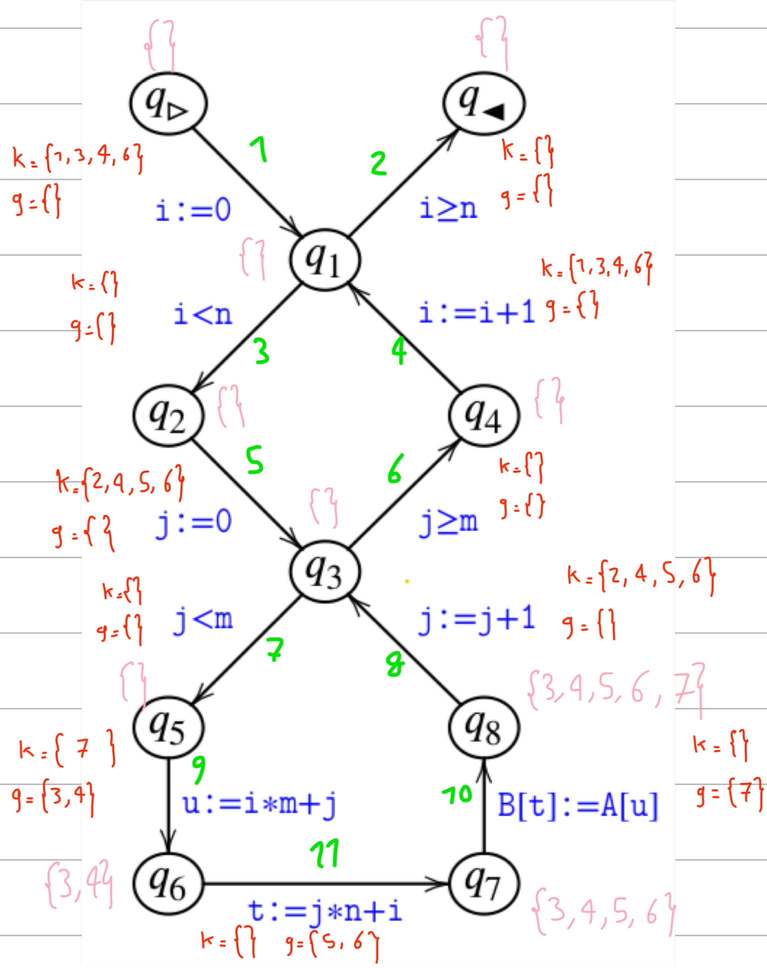
9) $(\{\} / \{7\}) \cup \{3,4\} \supseteq \{3,4\}$ ✓

10) $(\{3,4,5,6\} / \{\}) \cup \{7\} \supseteq \{3,4,5,6,7\}$ ✓

11) $(\{3,4\} / \{\}) \cup \{5,6\} \supseteq \{3,4,5,6\}$ ✓

Try it out 2.54:

AExp = $\{\overset{1}{i+1}, \overset{2}{j+1}, \overset{3}{i*m}, \overset{4}{i*m+j}, \overset{5}{j*n}, \overset{6}{j*n+i}, \overset{7}{A[u]}\}$



Order of selected edges:

1, 2, 3, 5, 6, 4, 7, 9,
11, 10, 8

Exercise 2.58:

Proof by contradiction: $AE(q) \not\supseteq AE'(q)$ for some q

so there are some expressions in AE' which are not in AE. Because

let's call it (expr)

AE' and AE both solves the constraint. For all edges (q_0, a, q)

$$(AE(q_0) / kill(a)) \cup gen(a) \supseteq AE(q)$$

$\Updownarrow$ equal

$$(AE'(q_0) / kill(a)) \cup gen(a) \supseteq AE'(q)$$

so (exp) is in all $(AE'(q_0) / kill(a)) \cup gen(a)$ and because these left sides of constraints are same for AE and AE', so (exp) is in AE too which is contradict with our assumption. ⨳

Teaser 2.61:

The second alternative is more general and it limits the available expressions carried along the edges.

Exercise 3.1:

α	$\mathsf{kill}_{\mathsf{RD}}(q_\circ, \alpha, q_\bullet)$	$\mathsf{gen}_{\mathsf{RD}}(q_\circ, \alpha, q_\bullet)$
$x := a$	$\{x\} \times \mathbf{Q}_? \times \mathbf{Q}$	$\{(x, q_\circ, q_\bullet)\}$
$A[a_1] := a_2$	$\{\ \}$	$\{(A, q_\circ, q_\bullet)\}$
$c?x$	$\{x\} \times \mathbf{Q}_? \times \mathbf{Q}$	$\{(x, q_\circ, q_\bullet)\}$
$c?A[a]$	$\{\ \}$	$\{(A, q_\circ, q_\bullet)\}$
$c!a$	$\{\ \}$	$\{\ \}$
b	$\{\ \}$	$\{\ \}$
$\mathtt{skip}$	$\{\ \}$	$\{\ \}$

α	$\widehat{S}_{\mathsf{RD}'}[\![q_\circ, \alpha, q_\bullet]\!](R')$
$x := a$	$R'[x \mapsto \{(q_\circ, q_\bullet)\}]$
$A[a_1] := a_2$	$R'[A \mapsto R'(A) \cup \{(q_\circ, q_\bullet)\}]$
$c?x$	$R'[x \mapsto \{(q_\circ, q_\bullet)\}]$
$c?A[a]$	$R'[A \mapsto R'(A) \cup \{(q_\circ, q_\bullet)\}]$
$c!a$	R'
b	R'
$\mathtt{skip}$	R'

$R_1 \in \text{Powerset}((\text{Var} \cup \text{Arr}) \times Q_? \times Q)$

$R_1' \in (\text{Var} \cup \text{Arr}) \to \text{Powerset}(Q_? \times Q)$

$x := a$ $\quad R_2 = (R_1 \setminus \text{kill}(q_\circ, x := a, q_\bullet)) \cup \text{gen}(q_\circ, x := a, q_\bullet)$

$= (R_1 \setminus \{x\} \times Q_? \times Q) \cup \{(x, q_\circ, q_\bullet)\}$

isomorphism

$R_2' = R_1'[x \mapsto (q_\circ, q_1)]$

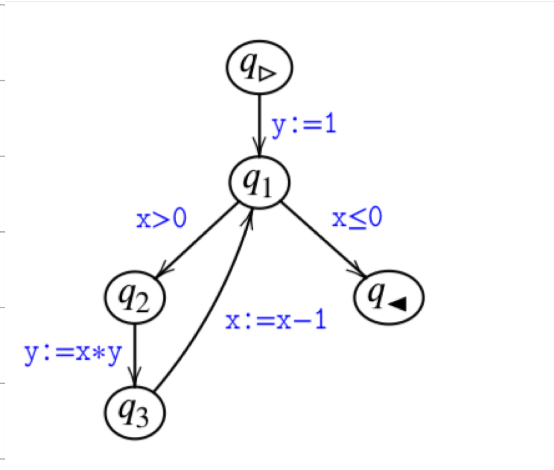
$A[a_1] := a_2$ $\quad R_2 = R_1 \cup \{(A, q_\circ, q_\bullet)\}$

$\|$ isomorphism

$R_2' = R_1'[A \to R_1'(A) \cup \{(q_\circ, q_\bullet)\}]$

Other actions are the same.

Try it out 3.2:



$RD(q_0)[y \mapsto (q_\triangleright, q_1)] \sqsubseteq RD(q_1)$

$RD(q_1) \sqsubseteq RD(q_2)$

$RD(q_2)[y \mapsto (q_2, q_3)] \sqsubseteq RD(q_3)$

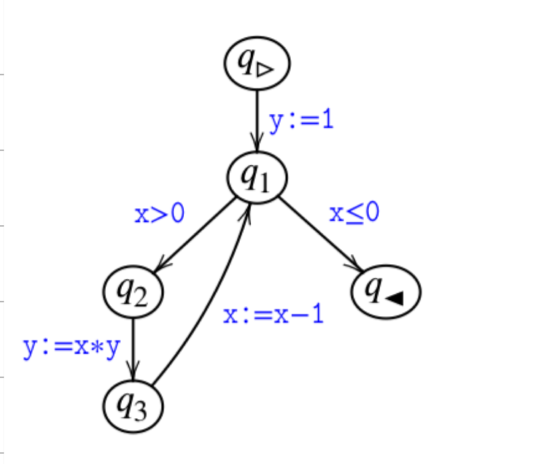
$RD(q_3)[x \mapsto (q_3, q_1)] \sqsubseteq RD(q_1)$

$RD(q_1) \sqsubseteq RD(q_\blacktriangleleft)$

Exercise 3.3:

We've shown that RD' is the map based of RD and $\hat{S}_{RD'}[\![q_\circ, a, q_\bullet]\!](R')$ is the mapping based of $(R \setminus kill(\alpha)) \cup gen(\alpha)$ and $\sqsubseteq$ is mimicking the $\subseteq$ operation for RD's. So there exists an isomorphism between 2 approaches constructing the constraints.

Try it out 3.4:



$$RD(q_\triangleright)[y \mapsto (q_\triangleright, q_1)] \sqsubseteq RD(q_1)$$

$$RD(q_1) \sqsubseteq RD(q_2)$$

$$RD(q_2)[y \mapsto (q_2, q_3)] \sqsubseteq RD(q_3)$$

$$RD(q_3)[x \mapsto (q_3, q_1)] \sqsubseteq RD(q_1)$$

$$RD(q_1) \sqsubseteq RD(q_\blacktriangleleft)$$

Algorithm:

$$RD(q_\triangleright) = \left[\begin{array}{l} x \mapsto \{(?, q_\triangleright)\} \\ y \mapsto \{(?, q_\triangleright)\} \end{array}\right]$$

$$RD(q_1) = \bot \stackrel{1}{\Longrightarrow} \left[\begin{array}{l} y \mapsto \{(q_\triangleright, q_1)\} \\ x \mapsto \{(?, q_\triangleright)\} \end{array}\right] \stackrel{4}{\Longrightarrow} \left[\begin{array}{l} y \mapsto \{(q_\triangleright, q_1), (q_2, q_3)\} \\ x \mapsto \{(?, q_\triangleright), (q_3, q_1)\} \end{array}\right]$$

$$RD(q_2) = \bot \stackrel{2}{\Longrightarrow} \left[\begin{array}{l} y \mapsto \{(q_\triangleright, q_1)\} \\ x \mapsto \{(?, q_\triangleright)\} \end{array}\right] \stackrel{5}{\Longrightarrow} \left[\begin{array}{l} y \mapsto \{(q_\triangleright, q_1), (q_2, q_3)\} \\ x \mapsto \{(?, q_\triangleright), (q_3, q_1)\} \end{array}\right]$$

$$RD(q_3) = \bot \stackrel{3}{\Longrightarrow} \left[\begin{array}{l} y \mapsto \{(q_2, q_3)\} \\ x \mapsto \{(?, q_\triangleright)\} \end{array}\right] \stackrel{6}{\Longrightarrow} \left[\begin{array}{l} y \mapsto \{(q_2, q_3)\} \\ x \mapsto \{(?, q_\triangleright), (q_3, q_1)\} \end{array}\right]$$

$$RD(q_\blacktriangleleft) = \bot \stackrel{7}{\Longrightarrow} \left[\begin{array}{l} y \mapsto \{(q_\triangleright, q_1), (q_2, q_3)\} \\ x \mapsto \{(?, q_\triangleright), (q_3, q_1)\} \end{array}\right]$$

Exercise 3.5:

The $\sqcup$ operator is the map based $\cup$ operator and $\bot$ is the map based $\emptyset$. The other relations have been discussed earlier, so we can conclude that the algorithms are the same.

Try it out 3.6:

$\hat{D} = \text{Powerset}((Var \cup Arr) \times Q_? \times Q)$, $\subseteq$

Reflexive: For each $\hat{d} \in \hat{D}$, $\hat{d}$ is a set of triples (or $\emptyset$) and we know that each set is a subset of itself

Transivity and anti-symmetric also holds trivially for it.

$\hat{D}' = (Var \cup Arr) \rightarrow \text{Powerset}(Q_? \times Q)$, $\sqsubseteq$

The $\sqsubseteq$ is somehow $\subseteq$ for each (input $\times$ output) point in this domain, so the reflexivity, transivity and anti-symmetric will be available.

Exercise 3.7:

The partially ordered set can be conducted pointwise for the map based domain $\hat{D}$ and reflexivity, transivity and anti-symmetric relations are hold for it.

Teaser 3.8:

def: $(d \sqsubseteq d') \wedge (d \neq d') \Rightarrow d \to d'$ transivity & reflexivity ✓

$d_1 \to d_2 \to d_3$

means that $d_1 \sqsubseteq d_2$, $d_2 \sqsubseteq d_3$, $d_1 \sqsubseteq d_3$ this can be removed

Exercise 3.10:

In general if $(D, \subseteq)$ is a partially ordered set:

$\cup : D \times D \to D$

$\forall d_1, d_2, d \in D : d_1 \subseteq d \wedge d_2 \subseteq d \Leftrightarrow d_1 \cup d_2 \subseteq d$ so $(\cup)$ satisfies the join function condition

$R' : (Var \cup Arr) \to Powerset(Q_? \times Q)$

$\sqcup : R' \times R' \to R'$

$R_1' \sqcup R_2' = \{ R' \mid \forall x \in Arr \cup Var : R'(x) = R_1'(x) \cup R_2'(x) \}$

we can say that

$\forall x \in Arr \cup Var, \forall R_1', R_2', R :$

$$R_1'(x) \subseteq R(x) \land R_2'(x) \subseteq R(x) \Leftrightarrow R_1'(x) \cup R_2'(x) \subseteq R$$

mapping based $\Downarrow$

$$R_1' \sqsubseteq R \land R_2' \sqsubseteq R \Leftrightarrow R_1' \sqcup R_2' \sqsubseteq R \quad \checkmark$$

Try it out 3.11:

$\hat{d} \sqcup \hat{d} = \hat{d}$ from $\sqcup$ definition:

$\forall \hat{d}, d : \quad \hat{d} \sqsubseteq d \land \hat{d} \sqsubseteq d \Leftrightarrow \hat{d} \sqcup \hat{d} \sqsubseteq d$

if $d = \hat{d} \Rightarrow \hat{d} \sqcup \hat{d} \sqsubseteq \hat{d}$ and trivially $\hat{d} \sqsubseteq \hat{d} \sqcup \hat{d}$

so from anti-symmetric relation: $\hat{d} \sqcup \hat{d} = \hat{d}$

$\hat{d} \sqcup \bot = \hat{d}$

$\hat{d} \sqsubseteq \hat{d} \wedge \bot \sqsubseteq \hat{d} \iff \hat{d} \sqcup \bot \sqsubseteq \hat{d}$

and trivially $\hat{d} \sqsubseteq \hat{d} \sqcup \bot$ so $\hat{d} \sqcup \bot = \hat{d}$

$\hat{d}_1 \sqcup \hat{d}_2 = \hat{d}_2 \sqcup \hat{d}_1$

$\forall \hat{d}_1, \hat{d}_2, \hat{d} : \hat{d}_1 \sqsubseteq \hat{d} \wedge \hat{d}_2 \sqsubseteq \hat{d} \iff \hat{d}_1 \sqcup \hat{d}_2 \sqsubseteq \hat{d}$

this operation is commutative ✓

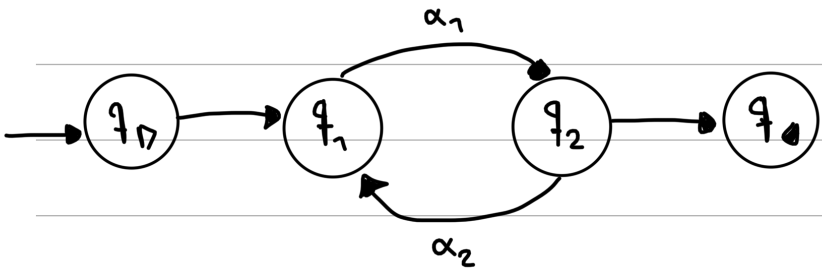
$\hat{d}_1 \sqcup (\hat{d}_2 \sqcup \hat{d}_3) = (\hat{d}_1 \sqcup \hat{d}_2) \sqcup \hat{d}_3$

because $\wedge$ is associative, $(\sqcup)$ is too.

$\hat{d}_1 \sqsubseteq \hat{d}_2 \wedge \hat{d}_2 \sqsubseteq \hat{d}_2 \iff \hat{d}_1 \sqcup \hat{d}_2 \sqsubseteq \hat{d}_2$

and trivially $\hat{d}_2 \sqsubseteq \hat{d}_1 \sqcup \hat{d}_2$ so : $\hat{d}_1 \sqcup \hat{d}_2 = \hat{d}_2$

Exercise 3.13:



S_1 $\hat{S}[\![q_1, \alpha_1, q_2]\!] AA(q_1) \sqsubseteq AA(q_2)$

S_2 $\hat{S}[\![q_2, \alpha_2, q_1]\!] AA(q_2) \sqsubseteq AA(q_1)$

if S_1, S_2 are like this:

$S_1(x) = S_2(x) = \{ x \cup \{ \text{max\_element}(x) + 1 \} \}$

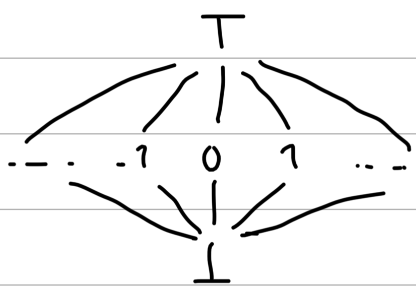
with this setting, the algorithm never terminates.

Teaser 3.14:

If the initialization is not the least element of the domain, then the final result could be the least solution or maybe not.

Exercise 3.15:

Hasse diagram:



As we can see, the height is 2 at most and this poset satisfies ascending chain condition

Exercise 3.17:

$$\forall y \in Y: \quad y \sqsubseteq l \iff \bigsqcup Y \sqsubseteq l$$

The above is the general idea for join operation. Pointed semi-lattice is a poset with a defined join operation which is described as above.

Try it out 3.21:

elements of R_2 which are not in R_1

$R_1 \subseteq R_2$ so $R_2 = R_1 \cup X$

$$f(R_2) = (R_2 \setminus kill) \cup gen = ((R_1 \cup X) \setminus kill) \cup gen$$

$$= \underbrace{[(R_1 \setminus kill) \cup gen]} \cup [(X \setminus kill) \cup gen]$$

$$= f(R_1) \cup f(x)$$

so $f(R_1) \subseteq f(R_2)$

same for mapping-based because there exists an isomorphism.

Try it out 3.24:

At q_D, all variables and arrays are dangerous and then with $\hat{s}$ they will update during the paths on the PG. A variable or an array can become indagerous if a value is assigned to it from a non dangerous variable or array; the logic of $\hat{S}$ is based on that.

Try it out 3.25.

Constraints:

$\{x, y\} \sqsubseteq DV(q_D)$

$DV(q_D) \setminus \{y\} \sqsubseteq DV(q_1)$

$DV(q_1) \sqsubseteq DV(q_2)$

if $\{x, y\} \cap DV(q_2) = \{\}$ then $DV(q_2) \setminus \{y\} \sqsubseteq DV(q_3)$

else $DV(q_2) \cup \{y\} \sqsubseteq DV(q_3)$

if $\{x\} \cap DV(q_3) = \{\}$ then $DV(q_3) \setminus \{x\} \sqsubseteq DV(q_1)$

else $DV(q_3) \cup \{x\} \sqsubseteq DV(q_1)$

$DV(q_1) \sqsubseteq DV(q_{\bullet})$

$DV(q_D) = \{x, y\}$

$DV(q_1) = \{x\} \rightarrow \{x, y\}$

$DV(q_2) = \{x\} \to \{x, y\}$

$DV(q_3) = \{x, y\}$

$DV(q_4) = \{x, y\}$

Exercise 3.27:

Same as try it out 3.21.

Try it out 3.32:

Suppose that $d \sqsubseteq d'$:

$f(d) \overset{?}{\sqsubseteq} f(d')$

$f(\underbrace{d \sqcup d'}_{d'}) = f(d) \sqcup f(d') \Rightarrow f(d') = f(d) \sqcup f(d') \Rightarrow$

$f(d) \sqsubseteq f(d')$ ✓