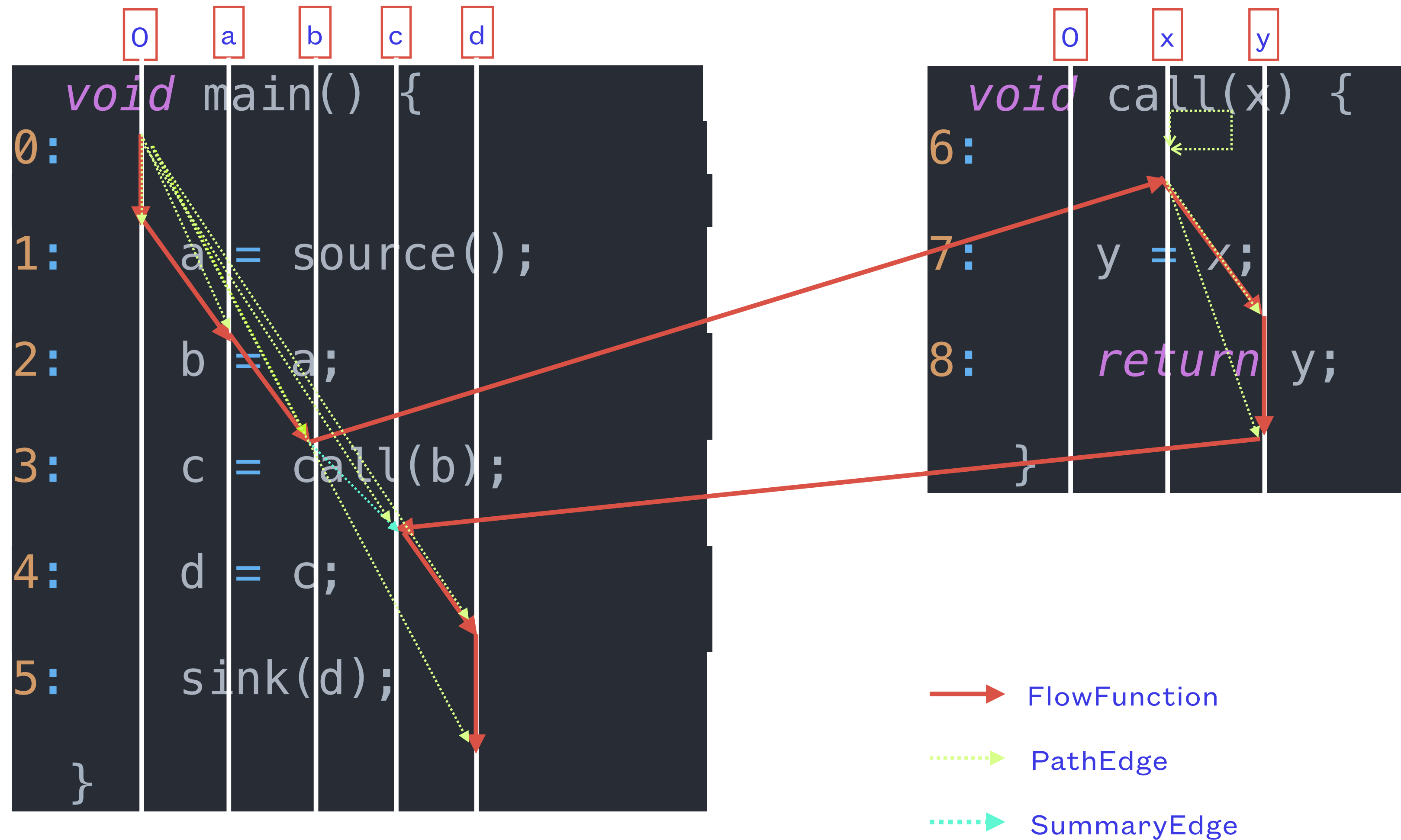

```
void main() {  
0:   
1:   a = source();  
2:   b = a;  
3:   c = call(b);  
4:   d = c;  
5:   sink(d);  
}
```

```
void call(x) {  
6:   
7:   y = x;  
8:   return y;  
}
```



```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs() SLRPs: Same Level Realizable Paths
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  : CallFlowFunction
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  : CallToReturnFlowFunction
[22]        for each  $c \in \text{callers}(p)$  do CallFlowFunction
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do ReturnFlowFunction
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  : NormalFlowFunction
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```

```

void main() {
0:
1:   a = source();
2:   b = a;
3:   c = call(b);
4:   d = c;
5:   sink(d);
}

```

```

void call(x) {
6:
7:   y = x;
8:   return y;
}

```

```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, 0 \rangle \rightarrow \langle s_{main}, 0 \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, 0 \rangle \rightarrow \langle s_{main}, 0 \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{0\}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]            for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]              Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]            od
[29]          fi
[30]        od
[31]      od
[32]    end case
[33]    case  $n \in (N_p - \text{Call}_p - \{e_p\})$  :
[34]      for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]        Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]      od
[37]    end case
[38]  end switch
[39] od
end

```

```

void main() {
0:
1:   a = source();
2:   b = a;
3:   c = call(b);
4:   d = c;
5:   sink(d);
}

```

```

void call(x) {
6:
7:   y = x;
8:   return y;
}

```

PathEdge

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

WorkList

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$


```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^\#$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]            for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]              Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]            od
[29]          fi
[30]        od
[31]      od
[32]    end case
[33]    case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]      for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]        Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]      od
[37]    end case
[38]  end switch
[39] od
end

```

```

void main() {
0:
1:   a = source();
2:   b = a;
3:   c = call(b);
4:   d = c;
5:   sink(d);
}

```

```

void call(x) {
6:
7:   y = x;
8:   return y;
}

```

PathEdge

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

WorkList

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

```
declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, 0 \rangle \rightarrow \langle s_{main}, 0 \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, 0 \rangle \rightarrow \langle s_{main}, 0 \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{0\}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end
procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end
procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{e_p\})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end
```

```
void main() {
0:
1:   a = source();
2:   b = a;
3:   c = call(b);
4:   d = c;
5:   sink(d);
} 0 a b c d
```

```
void call(x) {
6:
7:   y = x;
8:   return y;
} 0 x y
```

PathEdge

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

WorkList

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

$\langle 1, \text{zero} \rangle \rightarrow \langle 2, a \rangle$

传递性: $a \rightarrow b$ 且 $b \rightarrow c \Rightarrow a \rightarrow c$

Propagate($\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$)


```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```

```

void main() {
0:
1:   a = source();
2:   b = a;
3:   c = call(b);
4:   d = c;
5:   sink(d);
}

```

```

void call(x) {
6:
7:   y = x;
8:   return y;
}

```

PathEdge

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$

WorkList

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$

Propagate($\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$)

```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]            for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]              Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]            od
[29]          fi
[30]        od
[31]      od
[32]    end case
[33]    case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]      for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]        Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]      od
[37]    end case
[38]  end switch
[39] od
end

```

```

void main() {
0:
1:   a = source();
2:   b = a;
3:   c = call(b);
4:   d = c;
5:   sink(d);
}

```

```

void call(x) {
6:
7:   y = x;
8:   return y;
}

```

PathEdge

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$

WorkList

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$


```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```

```

void main() {
0:
1:   a = source();
2:   b = a;
3:   c = call(b);
4:   d = c;
5:   sink(d);
}

```

```

void call(x) {
6:
7:   y = x;
8:   return y;
}

```

PathEdge

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$

WorkList

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$

$\langle 2, a \rangle \rightarrow \langle 3, b \rangle$

Propagate($\langle s_{main}, \text{zero} \rangle \rightarrow \langle 3, b \rangle$)

NormalFlowFunction

实际上当前Normal Flow Function会存在另一条数据流 $\langle 2, a \rangle \rightarrow \langle 3, a \rangle$, 导致生成PathEdge $\langle s_{main}, \text{zero} \rangle \rightarrow \langle 3, a \rangle$

这里限于篇幅以及算法演示的时间，对于这类边不予演示；对于最终Source->Sink并不影响

```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```

```

void main() {
0:
1:   a = source();
2:   b = a;
3:   c = call(b);
4:   d = c;
5:   sink(d);
}

```

```

void call(x) {
6:
7:   y = x;
8:   return y;
}

```

PathEdge

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$

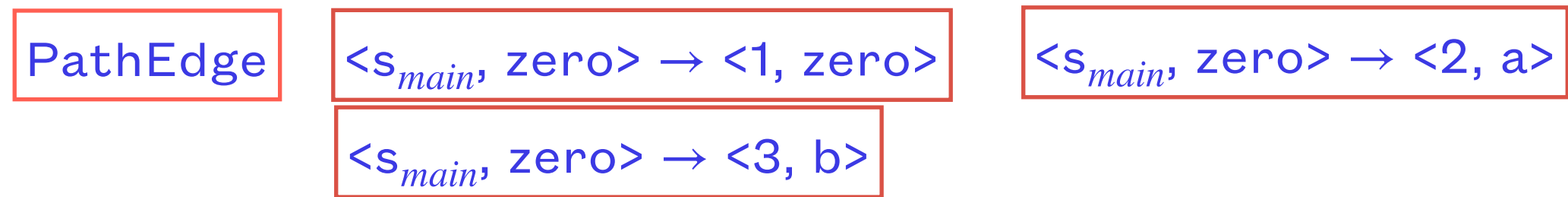
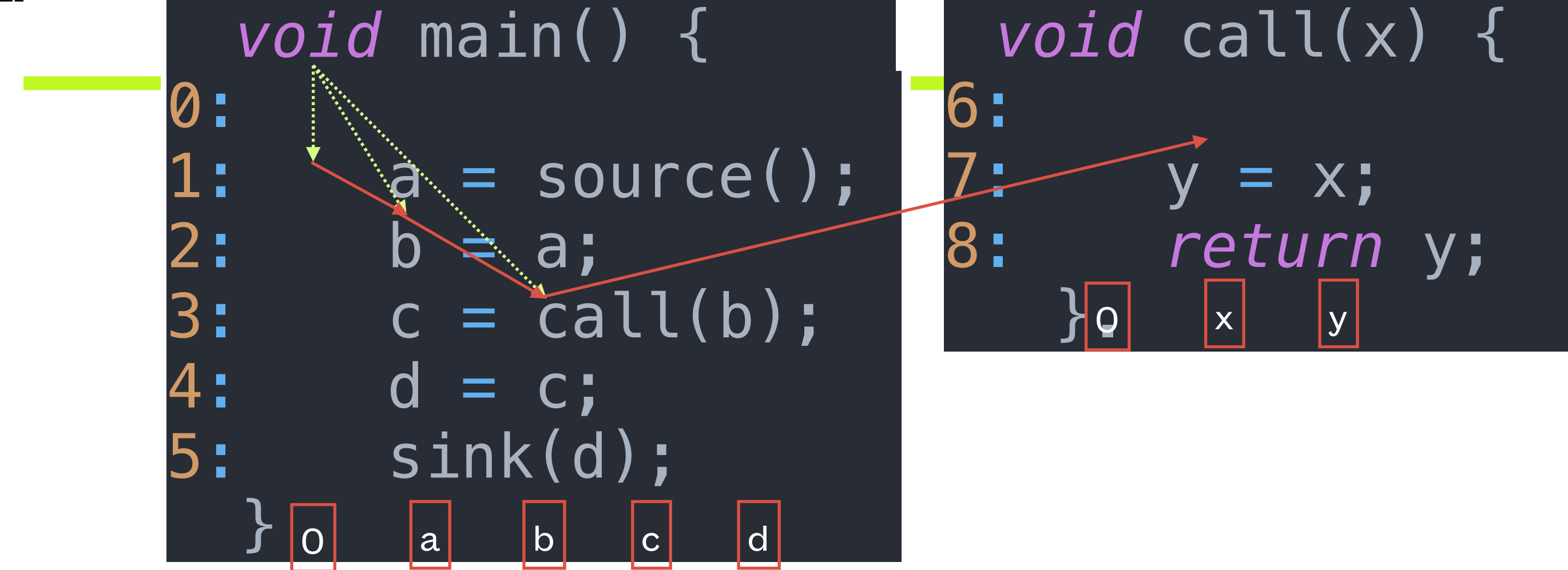
$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 3, b \rangle$

WorkList

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 3, b \rangle$

Propagate($\langle s_{main}, \text{zero} \rangle \rightarrow \langle 3, b \rangle$)


```
declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1] Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2] PathEdge :=  $\{ \langle s_{main}, 0 \rangle \rightarrow \langle s_{main}, 0 \rangle \}$ 
[3] WorkList :=  $\{ \langle s_{main}, 0 \rangle \rightarrow \langle s_{main}, 0 \rangle \}$ 
[4] SummaryEdge :=  $\emptyset$ 
[5] ForwardTabulateSLRPs()
[6] for each  $n \in N^*$  do
[7]    $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{0\}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8] od
end
procedure Propagate( $e$ )
begin
[9] if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end
procedure ForwardTabulateSLRPs()
begin
[10] while WorkList  $\neq \emptyset$  do
[11]   Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]   switch  $n$ 
[13]     case  $n \in \text{Call}_p$  :
[14]       for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]         Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]       od
[17]       for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]         Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]       od
[20]     end case
[21]     case  $n = e_p$  :
[22]       for each  $c \in \text{callers}(p)$  do
[23]         for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]           if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]             Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]             for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]               Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]             od
[29]           fi
[30]         od
[31]       od
[32]     end case
[33]     case  $n \in (N_p - \text{Call}_p - \{e_p\})$  :
[34]       for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]         Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]       od
[37]     end case
[38]   end switch
[39] od
end
```



自环边!!! 导致被调用函数内的数据流传播是上下文无关的

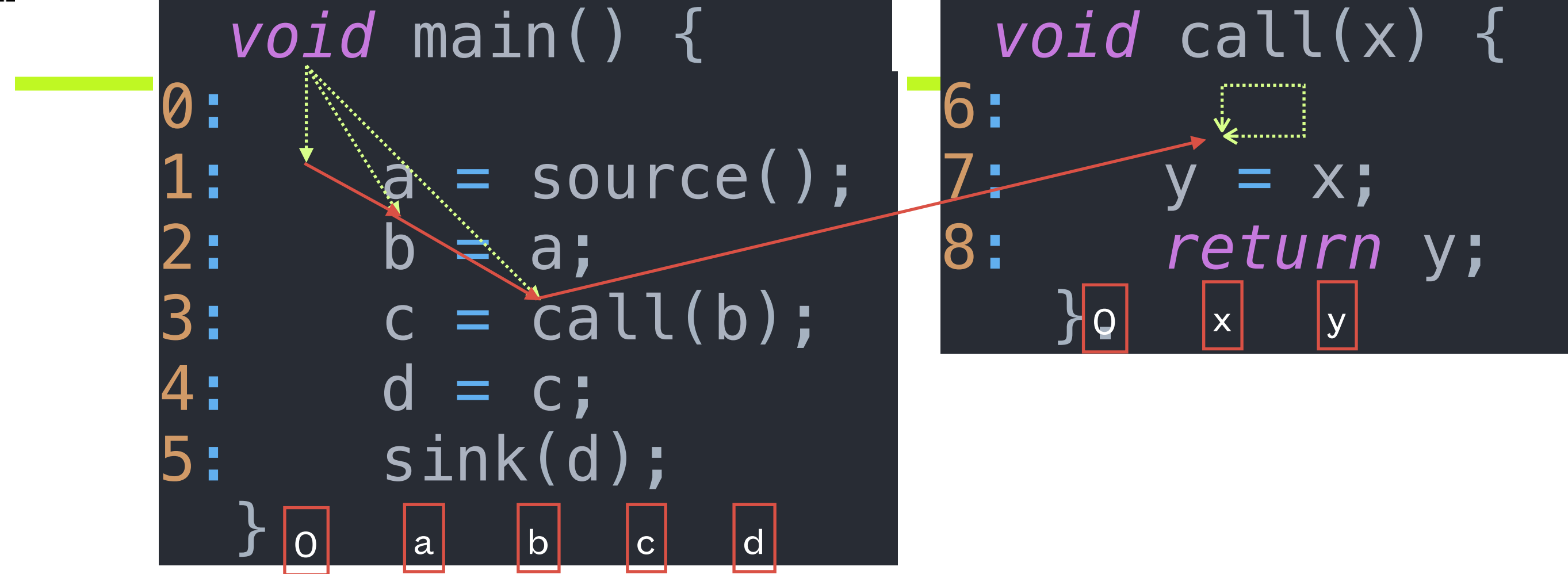

```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```



PathEdge

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 3, b \rangle$

$\langle s_{call}, x \rangle \rightarrow \langle s_{call}, x \rangle$

WorkList

$\langle s_{call}, x \rangle \rightarrow \langle s_{call}, x \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 3, b \rangle$

```

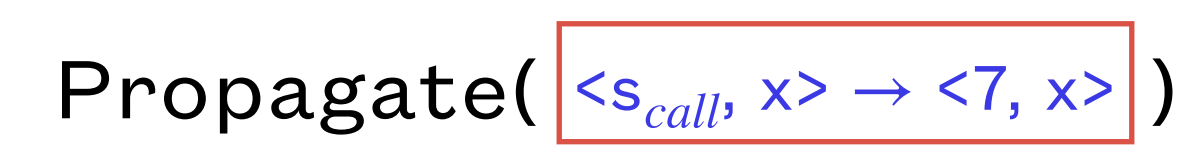
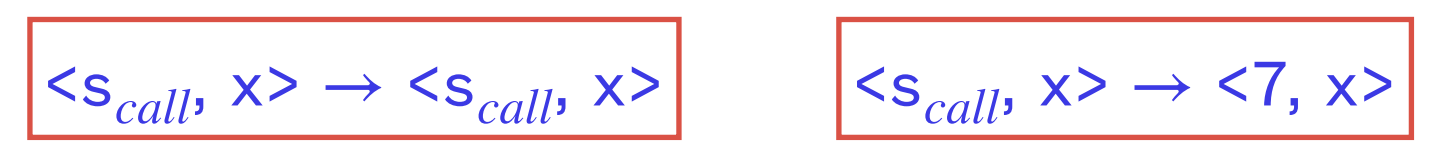
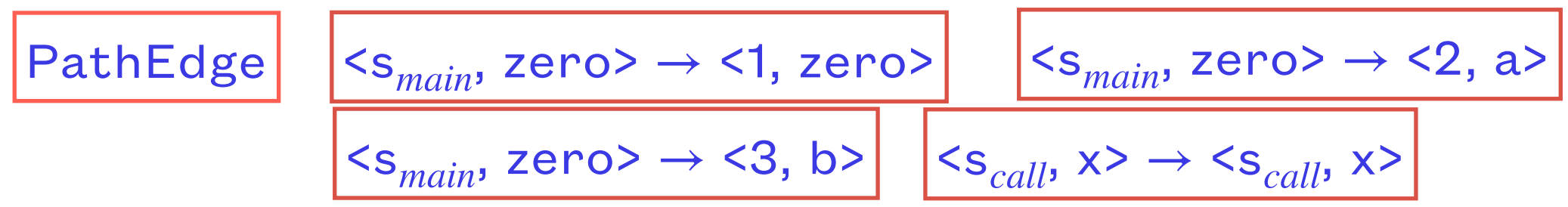
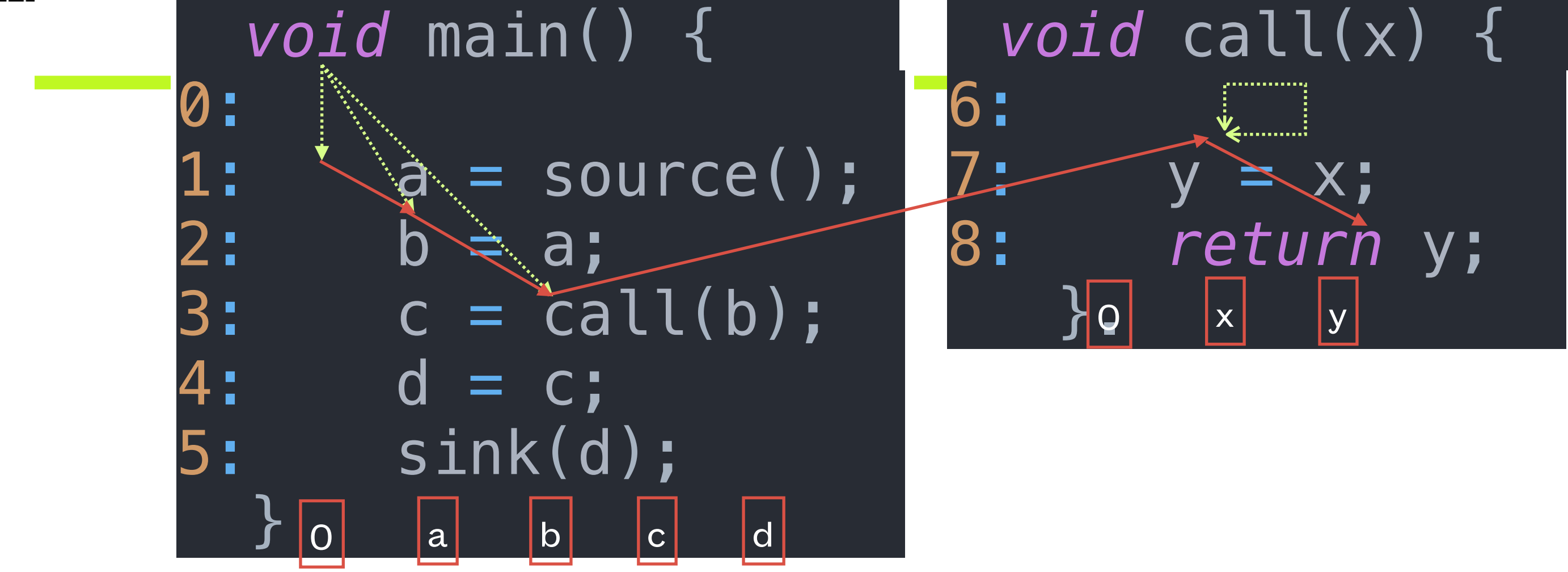
declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]            for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]              Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]            od
[29]          fi
[30]        od
[31]      od
[32]    end case
[33]    case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]      for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]        Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]      od
[37]    end case
[38]  end switch
[39] od
end

```

NormalFlowFunction



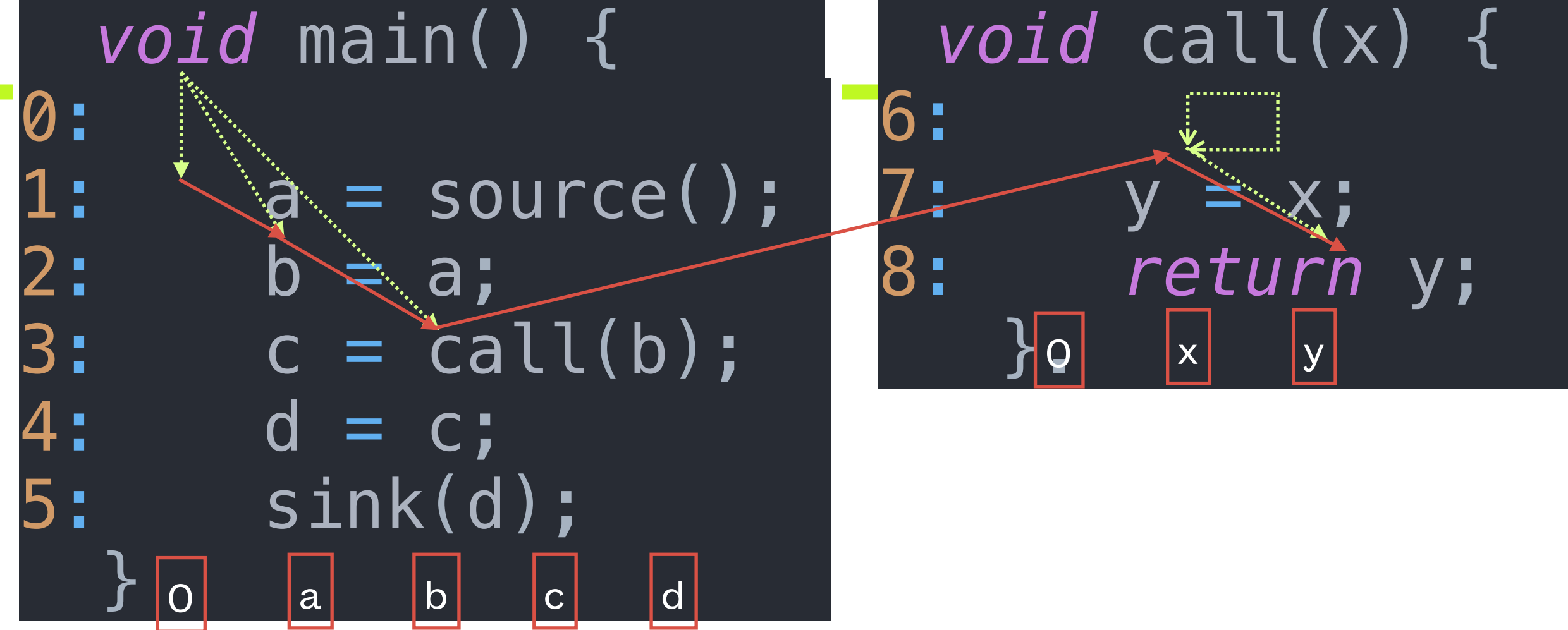

```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```



PathEdge

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 3, b \rangle$

$\langle s_{call}, x \rangle \rightarrow \langle s_{call}, x \rangle$

$\langle s_{call}, x \rangle \rightarrow \langle 7, x \rangle$

WorkList

$\langle s_{call}, x \rangle \rightarrow \langle 7, x \rangle$


```

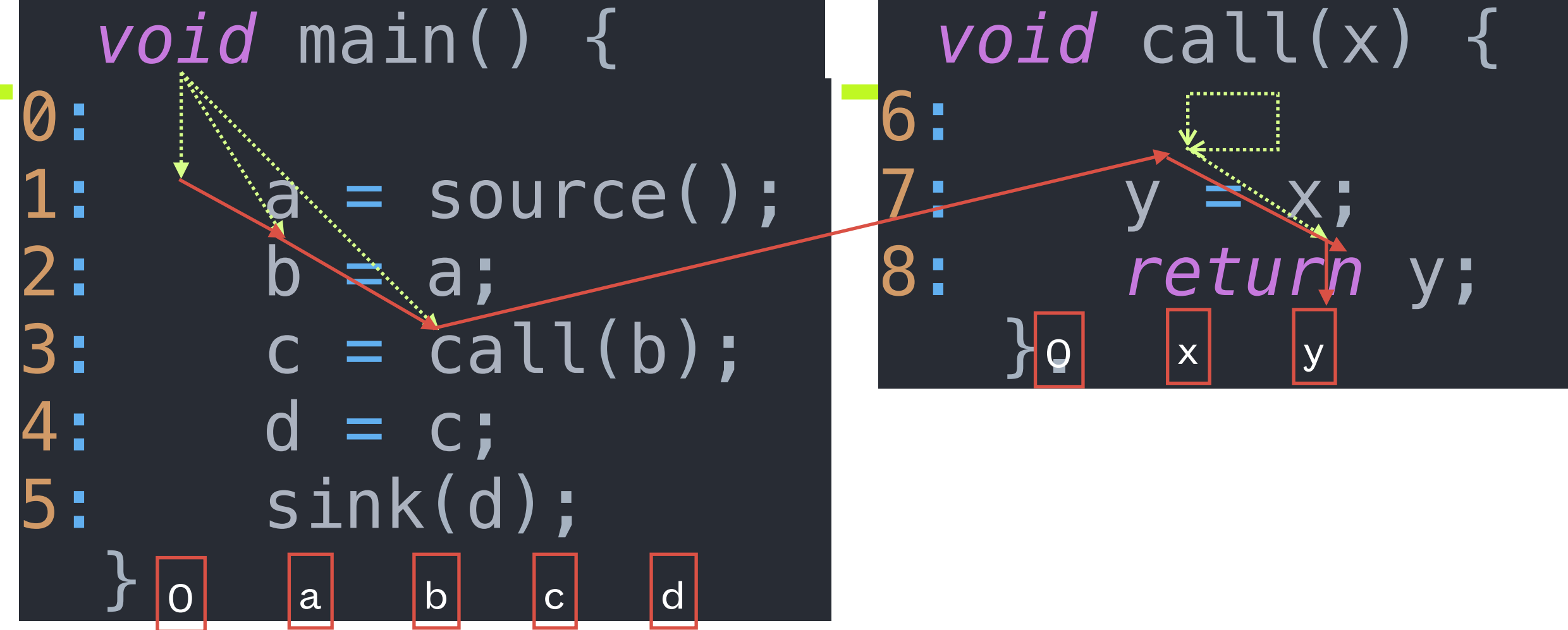
declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]            for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]              Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]            od
[29]          fi
[30]        od
[31]      od
[32]    end case
[33]    case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]      for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]        Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]      od
[37]    end case
[38]  end switch
[39] od
end

```

NormalFlowFunction



PathEdge

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 3, b \rangle$

$\langle s_{call}, x \rangle \rightarrow \langle s_{call}, x \rangle$

$\langle s_{call}, x \rangle \rightarrow \langle 7, x \rangle$

WorkList

$\langle s_{call}, x \rangle \rightarrow \langle 7, x \rangle$

$\langle 7, x \rangle \rightarrow \langle 8, y \rangle$

Propagate($\langle s_{call}, x \rangle \rightarrow \langle 8, y \rangle$)

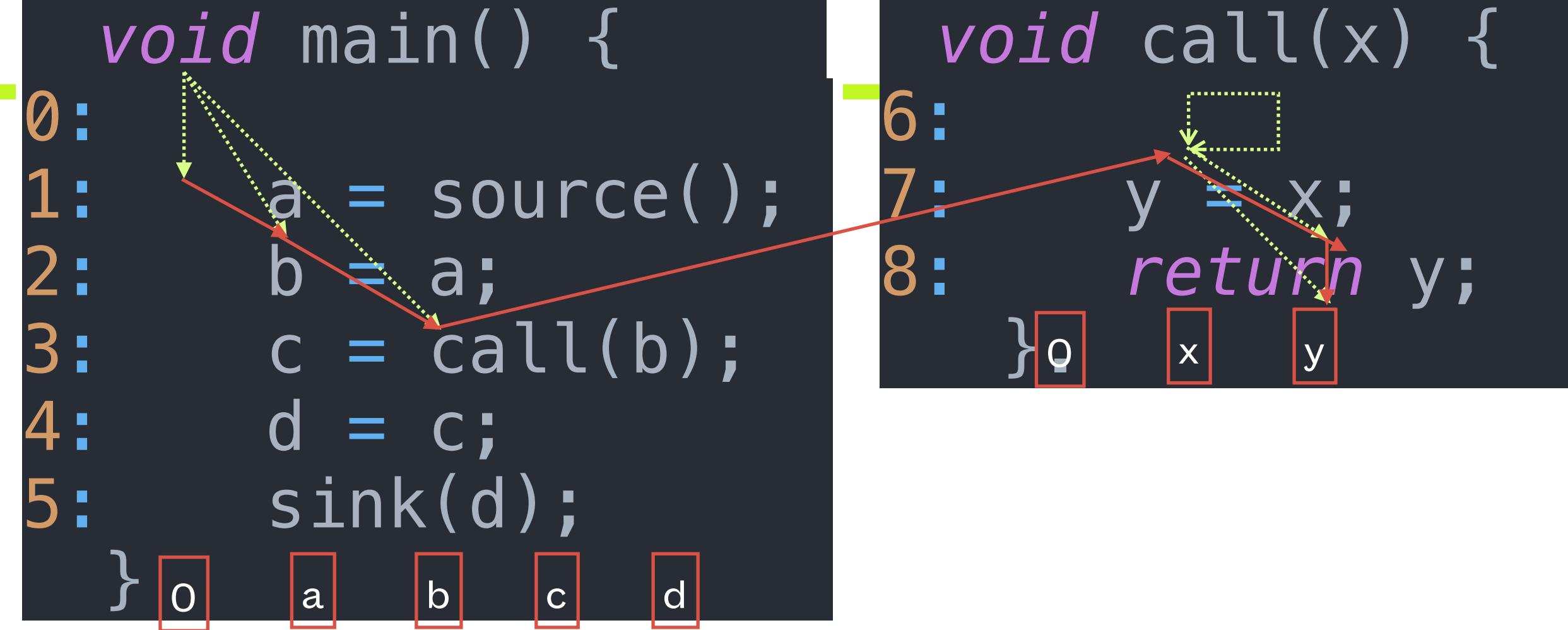
```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```



PathEdge

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 3, b \rangle$

$\langle s_{call}, x \rangle \rightarrow \langle s_{call}, x \rangle$

$\langle s_{call}, x \rangle \rightarrow \langle 7, x \rangle$

$\langle s_{call}, x \rangle \rightarrow \langle 8, y \rangle$

WorkList

$\langle s_{call}, x \rangle \rightarrow \langle 8, y \rangle$

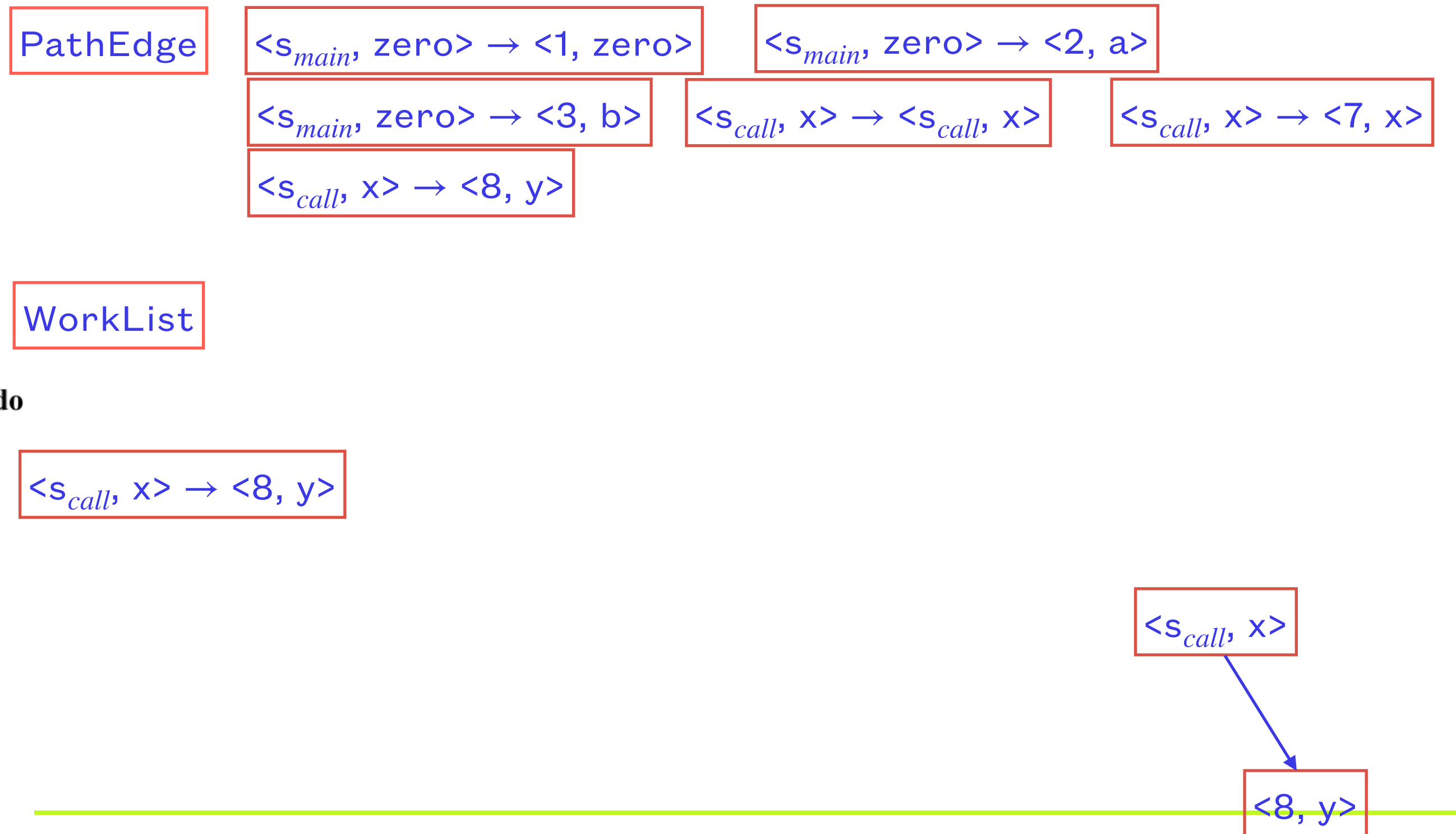
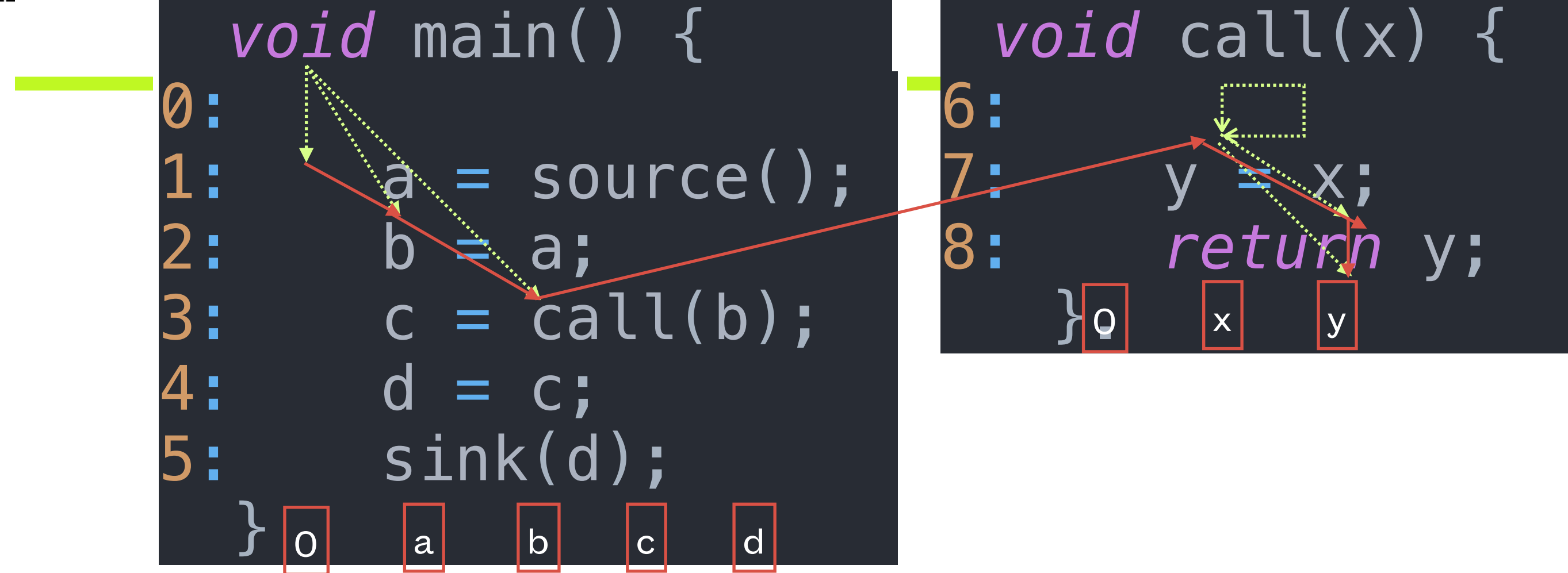

```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```



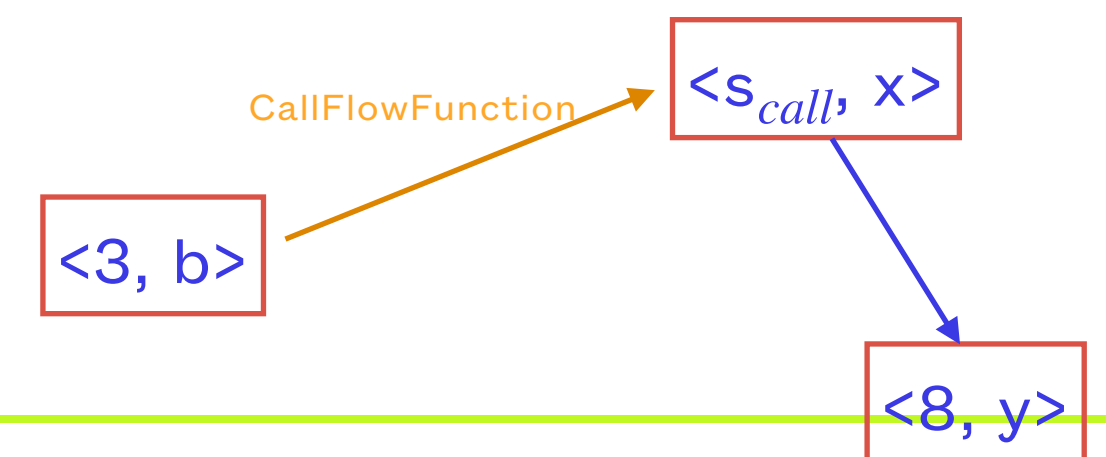
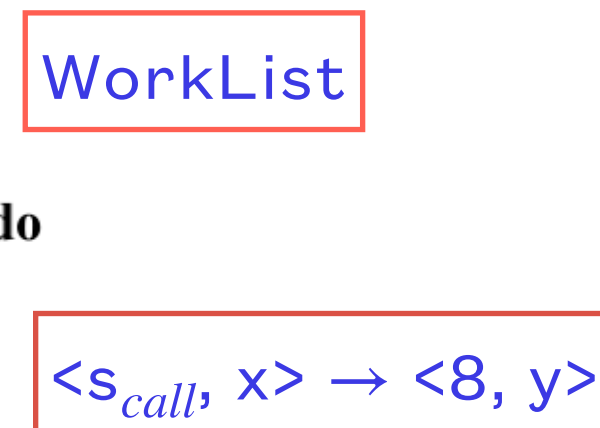
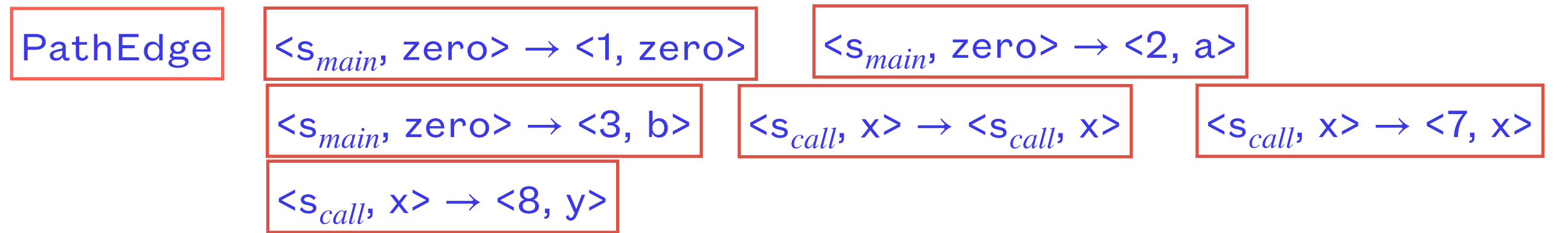
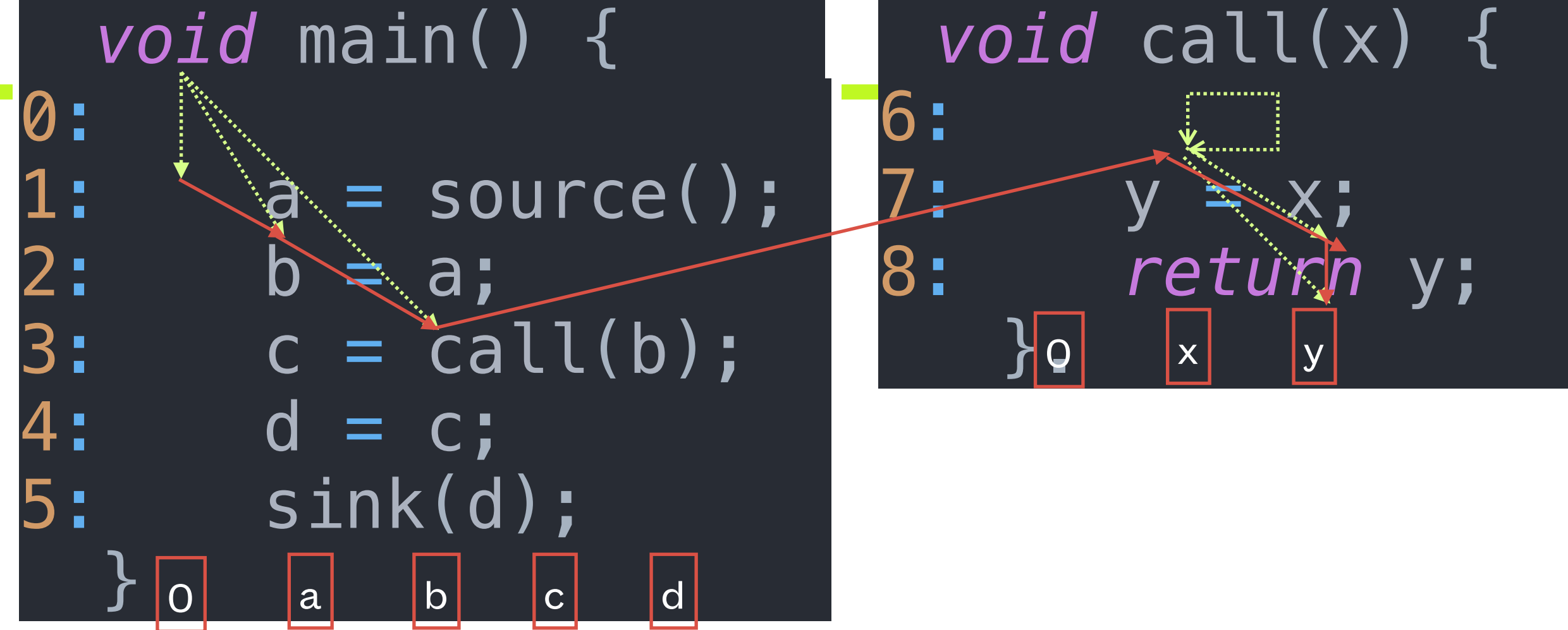

```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```



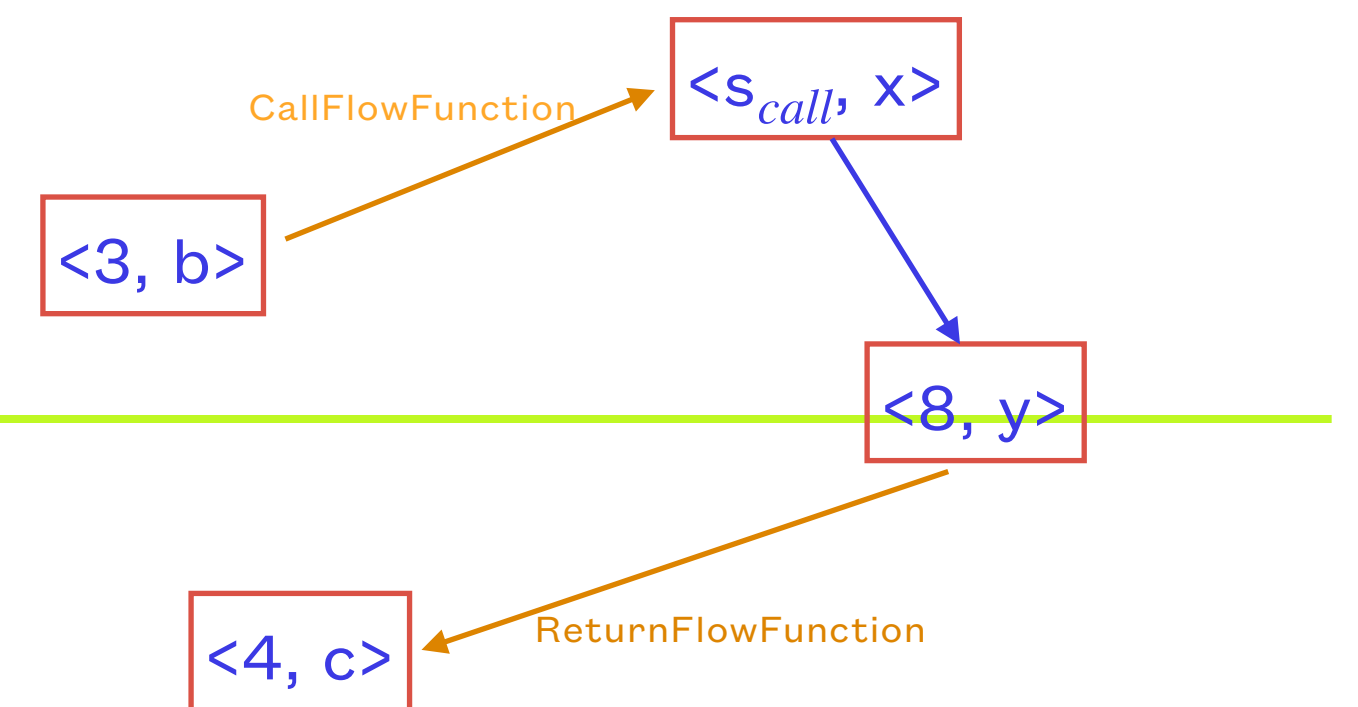
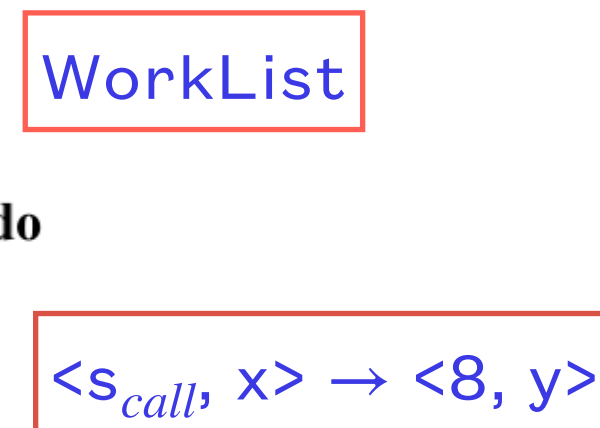
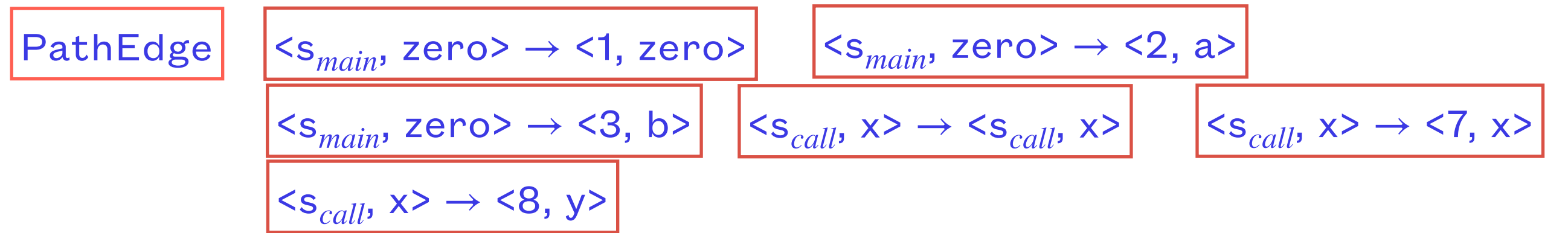
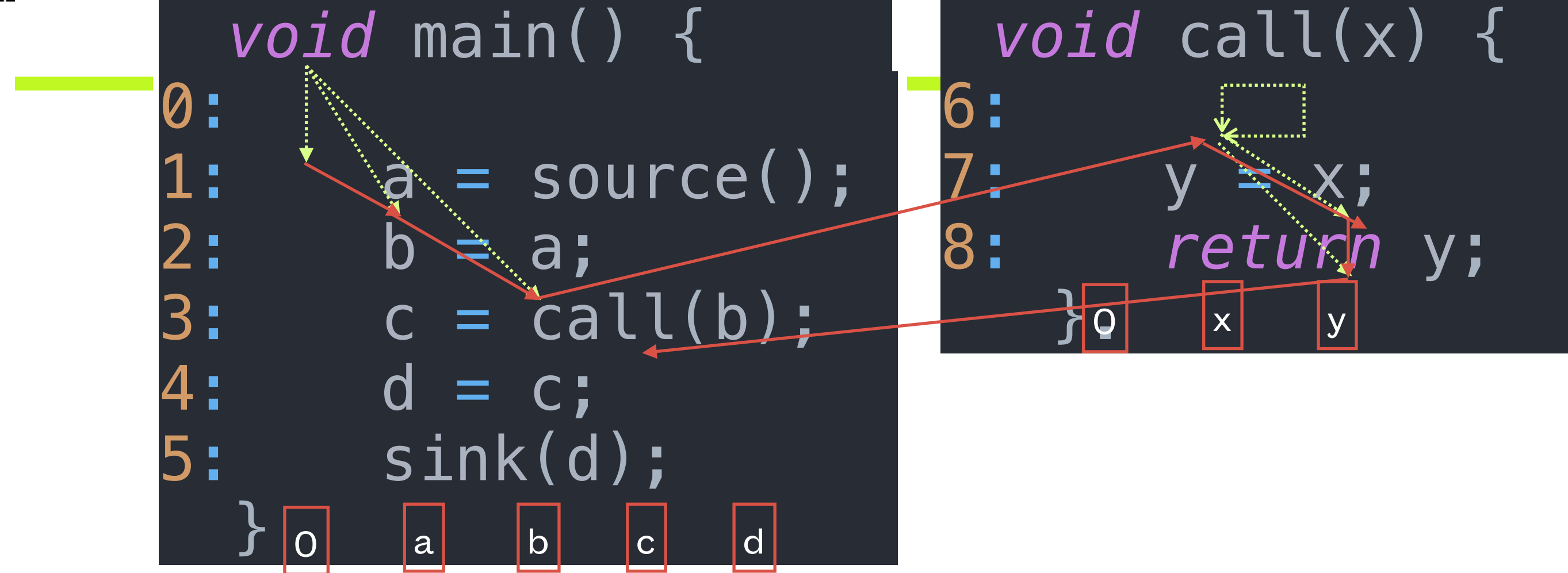
```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```



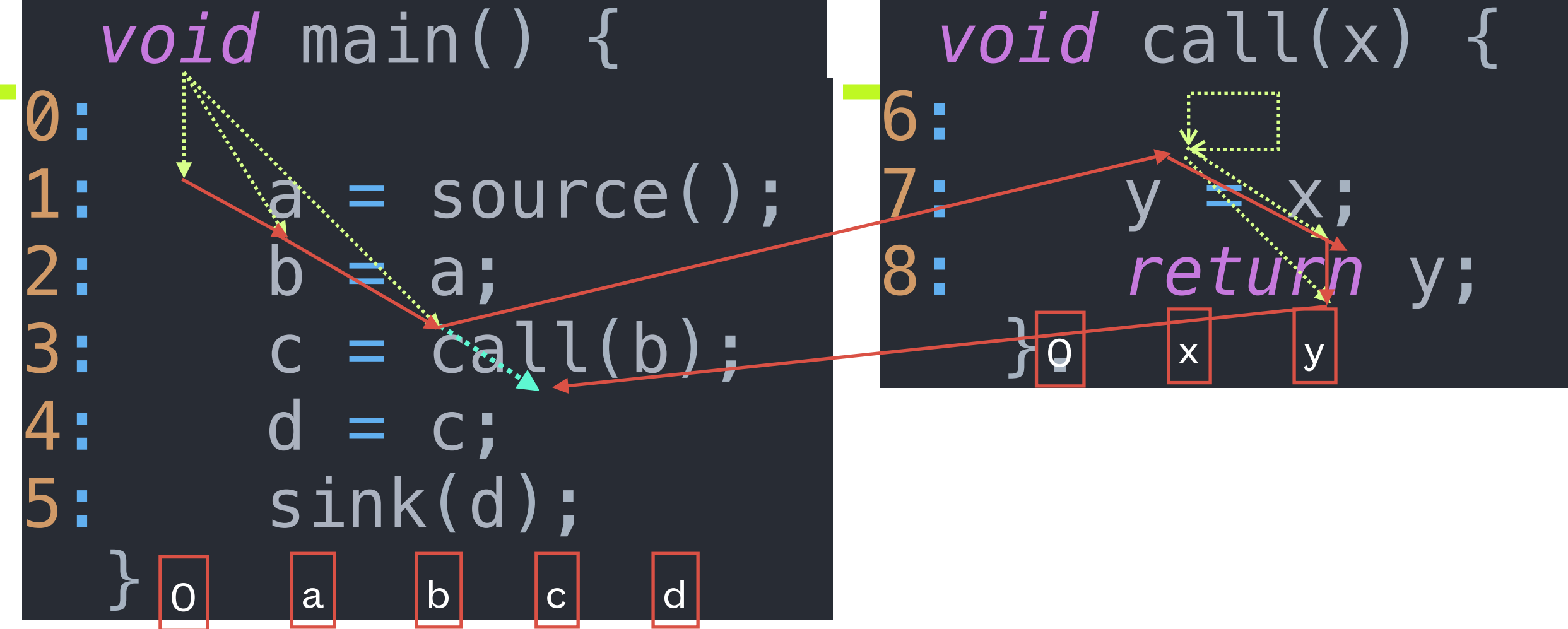

```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]            for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]              Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]            od
[29]          fi
[30]        od
[31]      od
[32]    end case
[33]    case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]      for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]        Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]      od
[37]    end case
[38]  end switch
[39] od
end

```



PathEdge

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 1, \text{zero} \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 2, a \rangle$

$\langle s_{main}, \text{zero} \rangle \rightarrow \langle 3, b \rangle$

$\langle s_{call}, x \rangle \rightarrow \langle s_{call}, x \rangle$

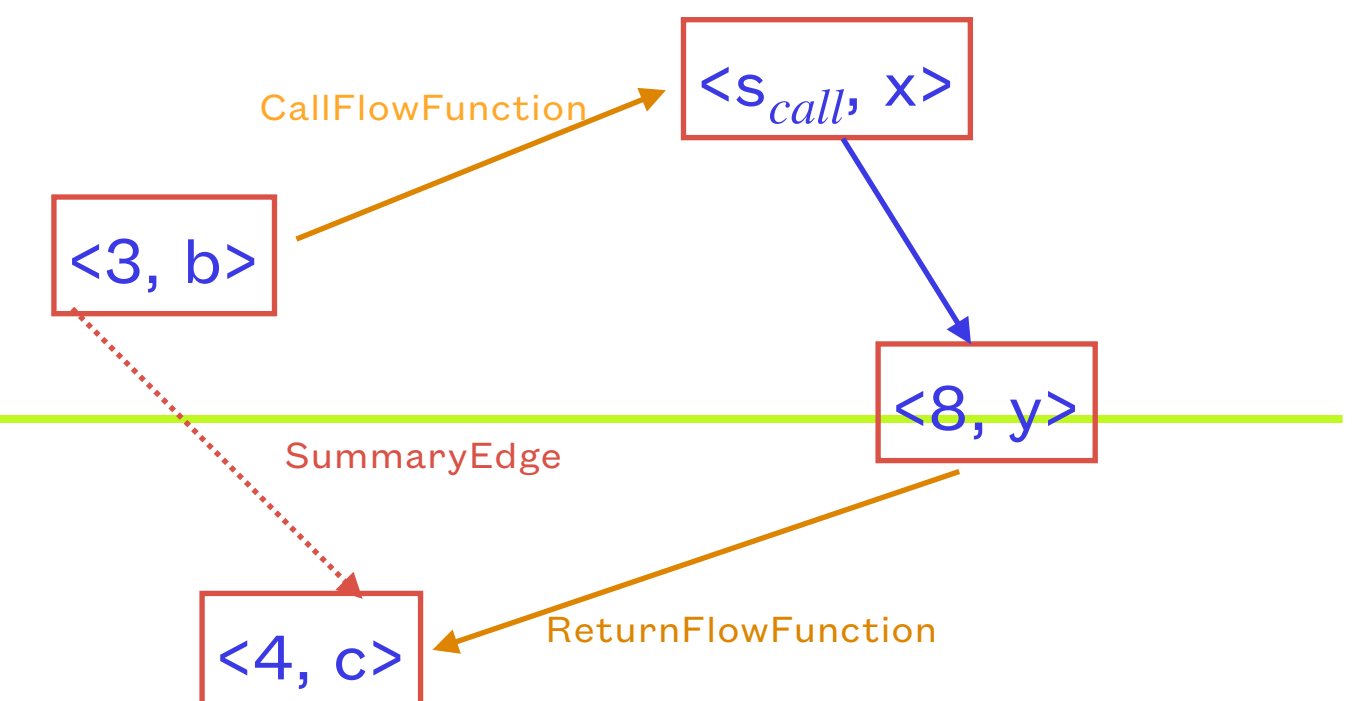
$\langle s_{call}, x \rangle \rightarrow \langle 7, x \rangle$

$\langle s_{call}, x \rangle \rightarrow \langle 8, y \rangle$

WorkList

$\langle s_{call}, x \rangle \rightarrow \langle 8, y \rangle$

SummaryEdge U= $\langle 3, b \rangle \rightarrow \langle 4, c \rangle$



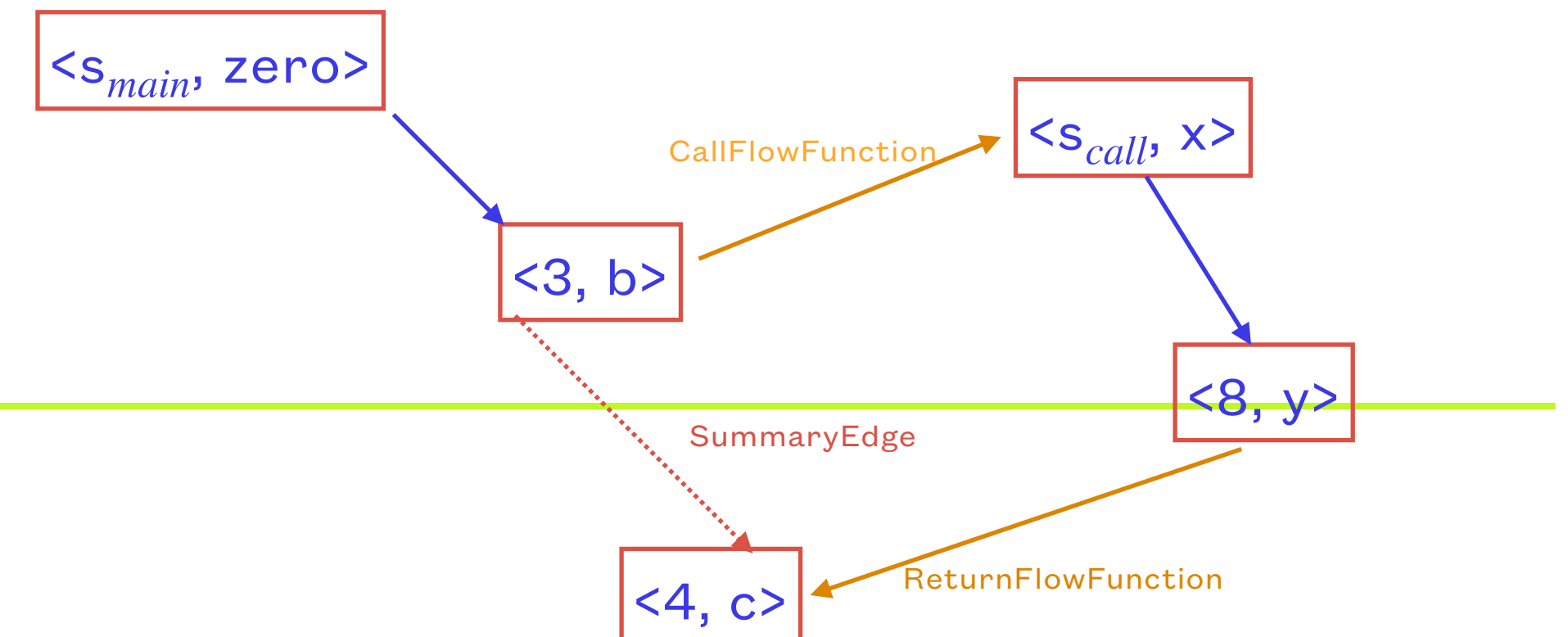
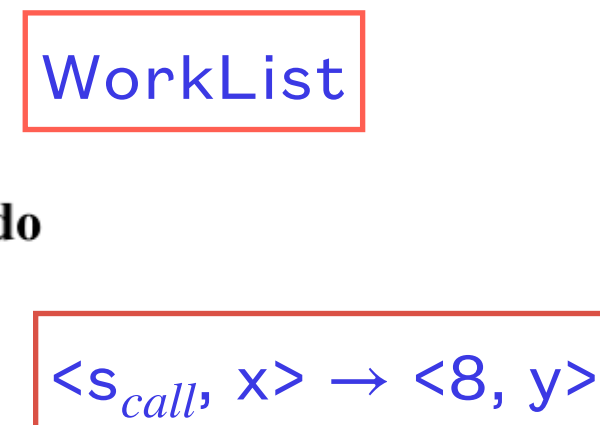
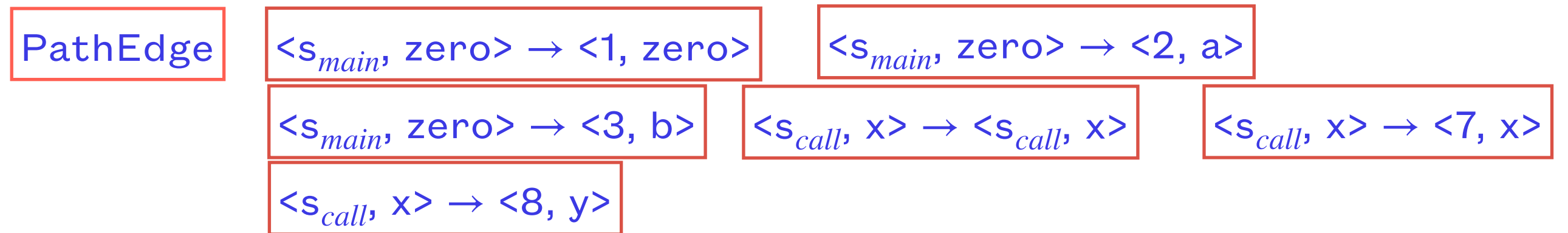
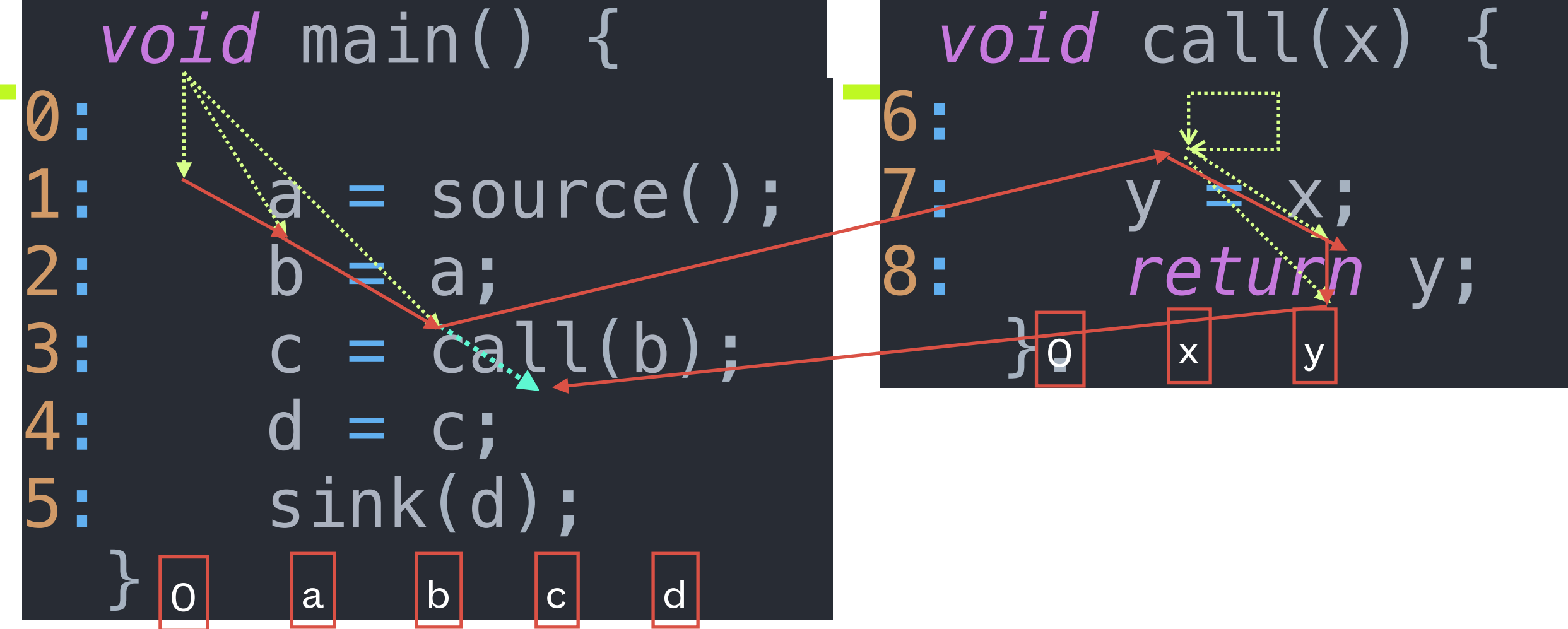

```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]            for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]              Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]            od
[29]          fi
[30]        od
[31]      od
[32]    end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```



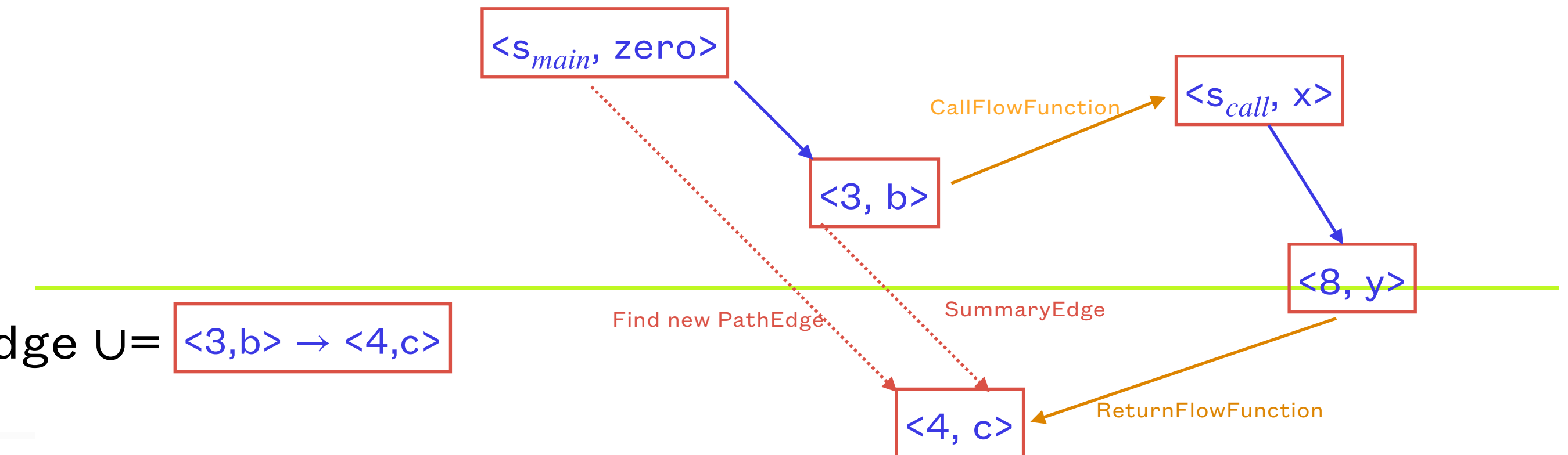
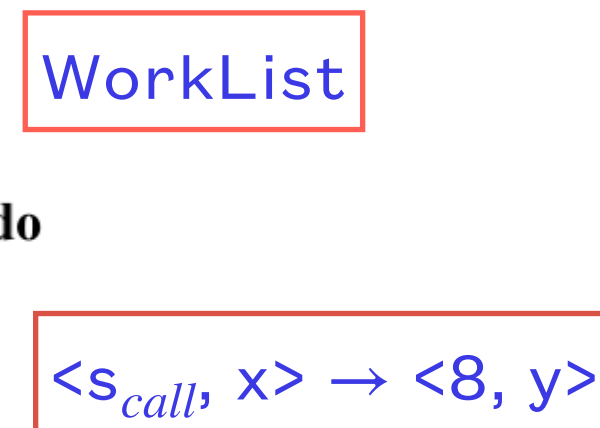
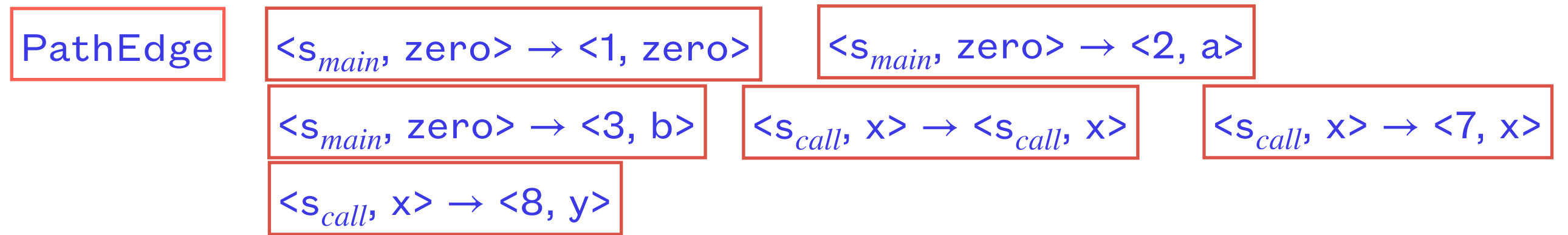
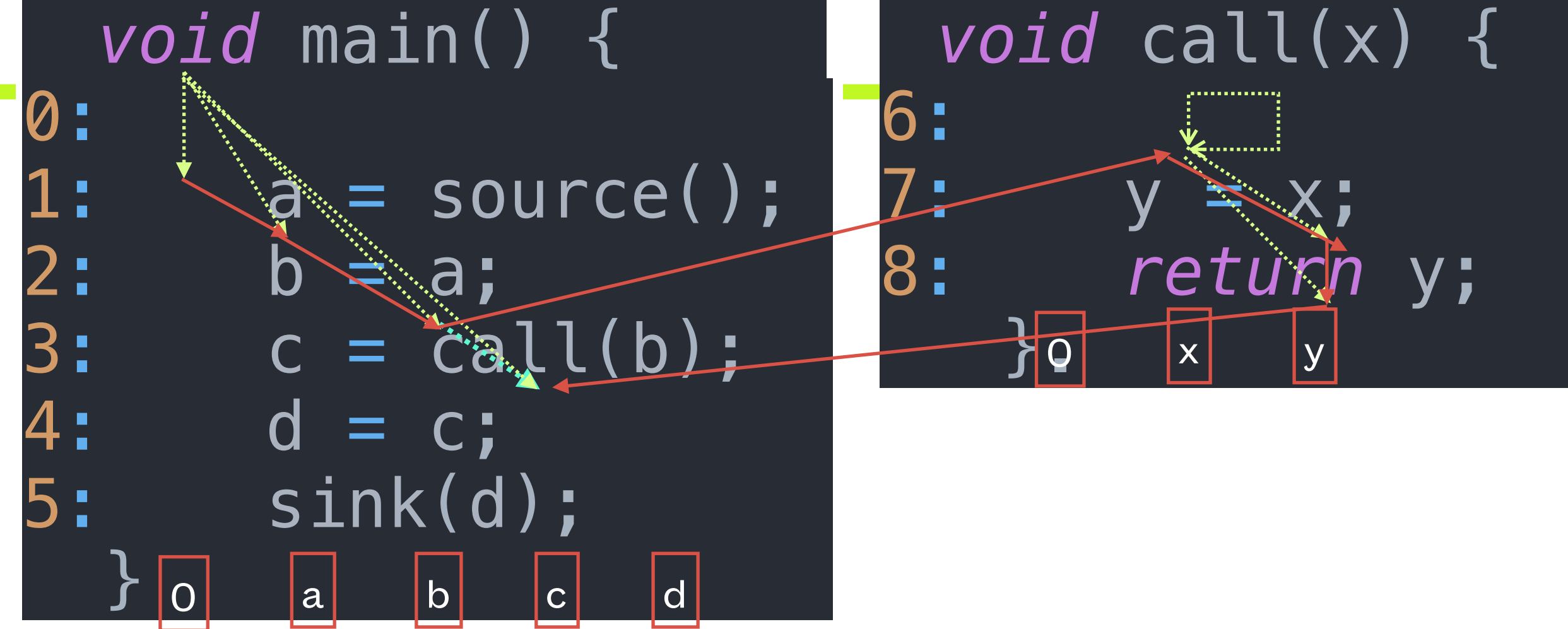
```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

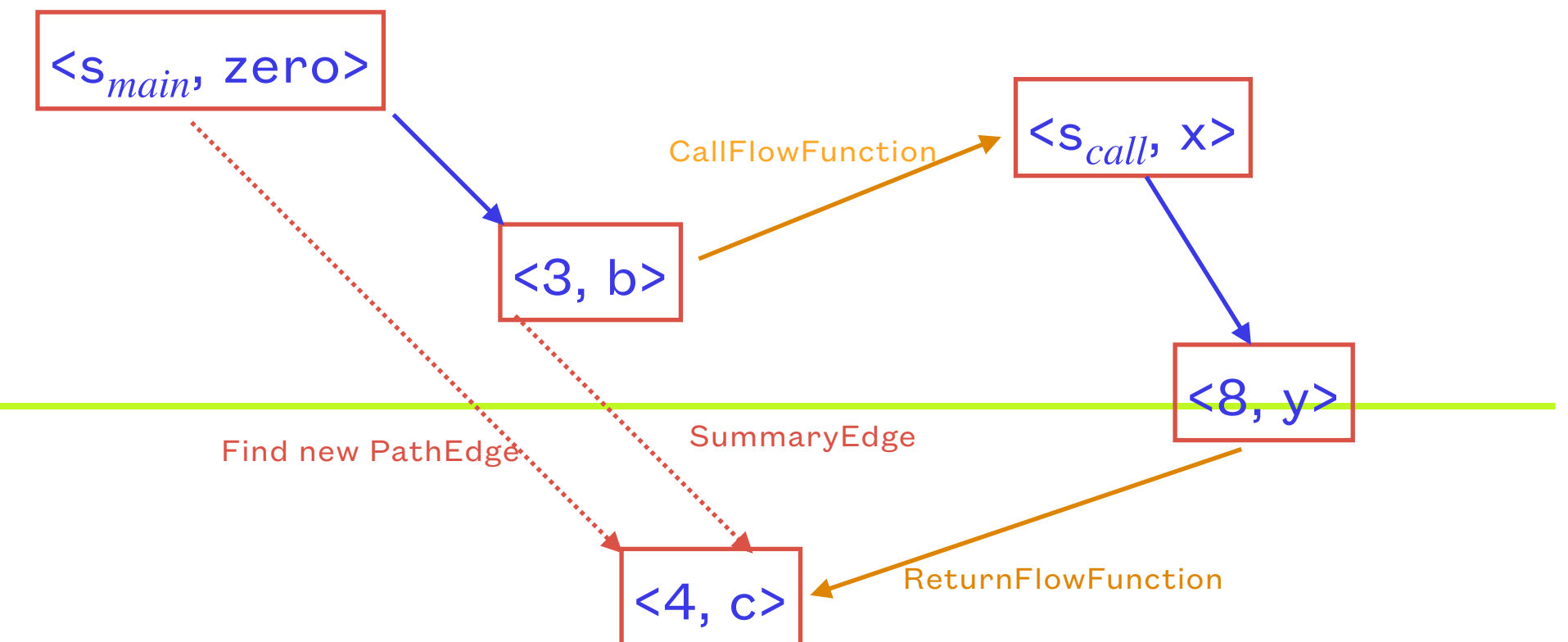
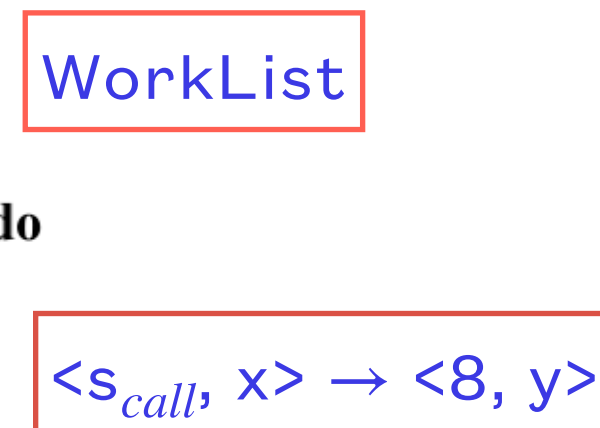
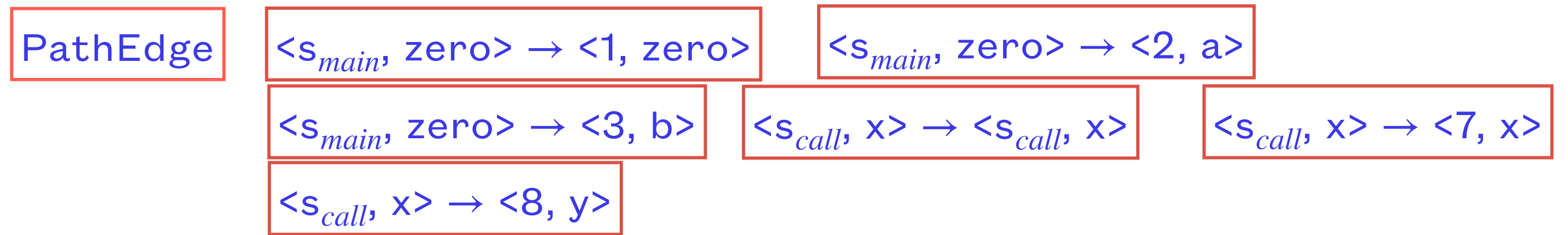
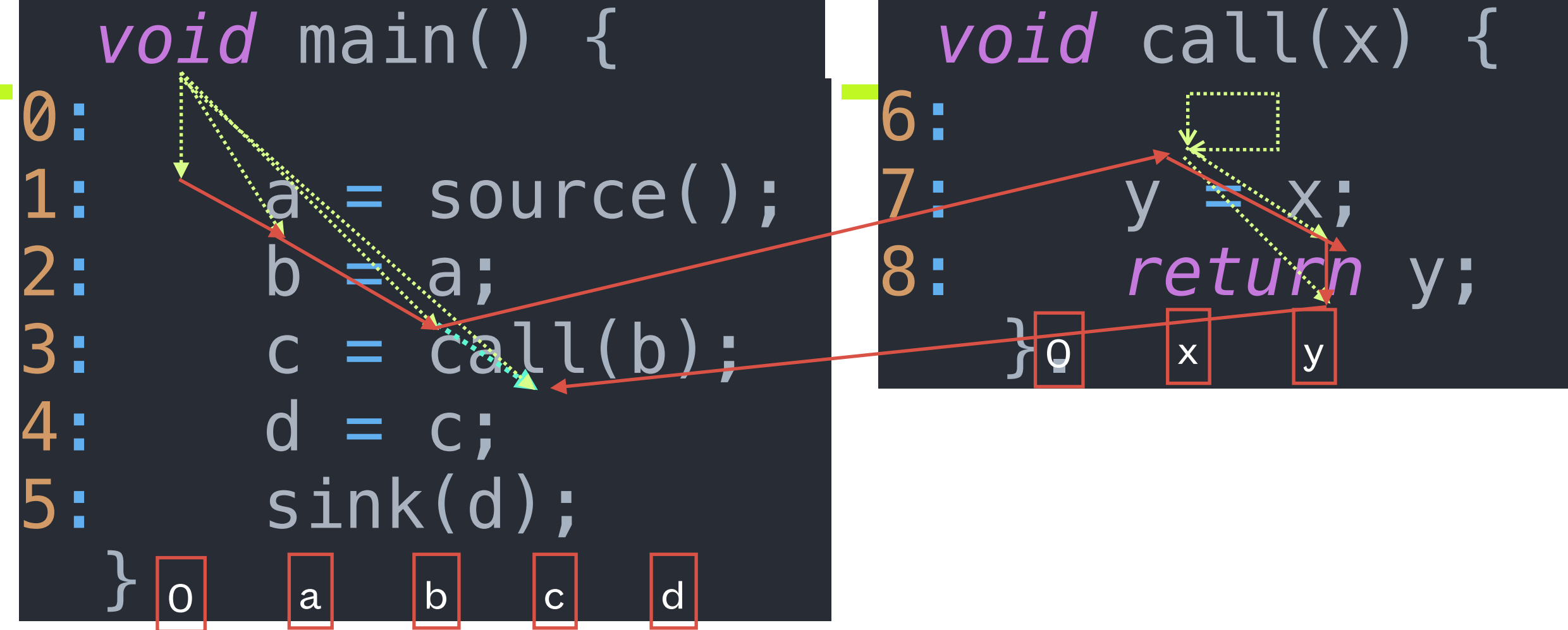
```




```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end
procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end
procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```



SummaryEdge U= $\langle 3, b \rangle \rightarrow \langle 4, c \rangle$

Propagate($\langle s_{main}, \text{zero} \rangle \rightarrow \langle 4, c \rangle$)

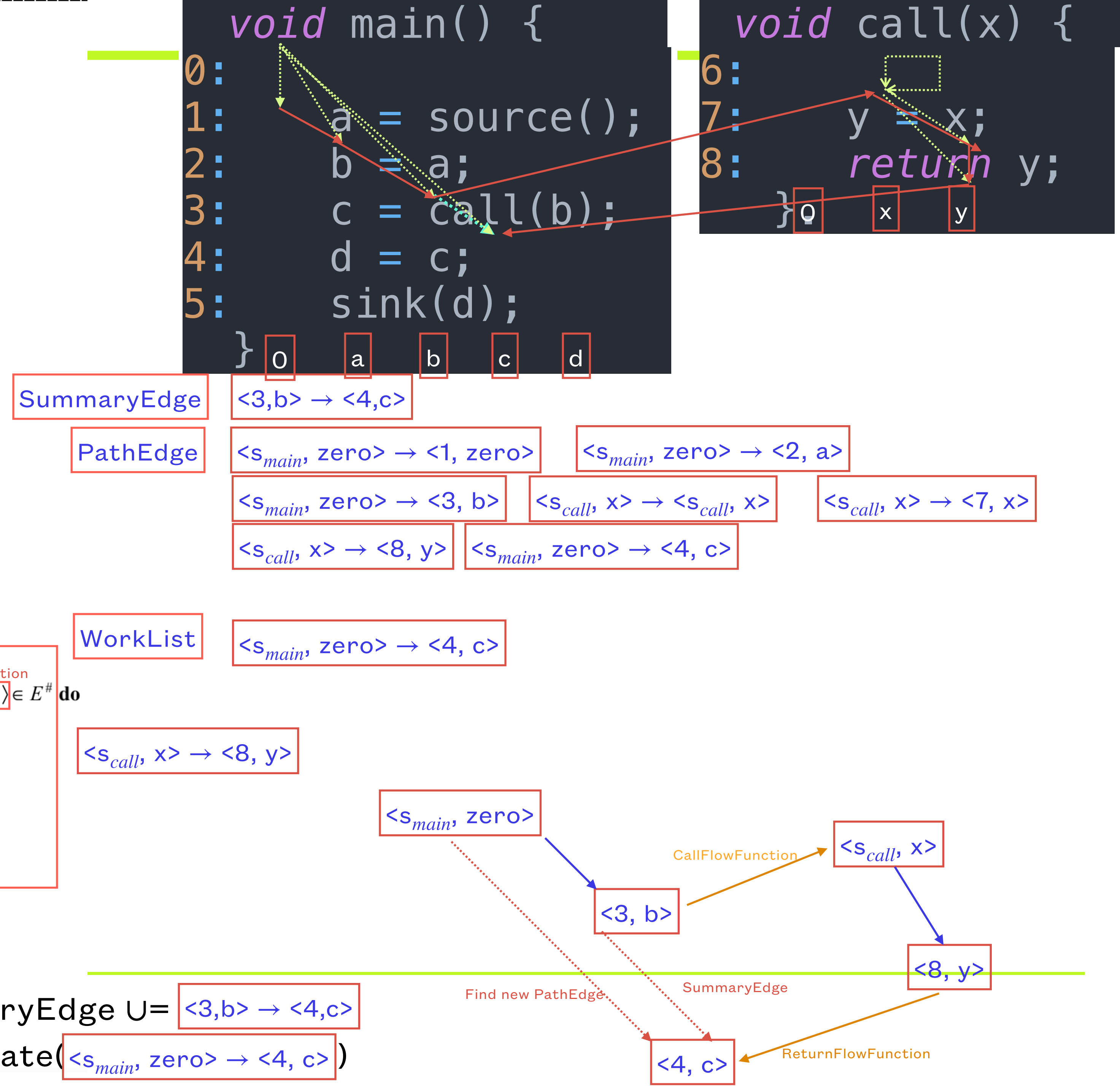
```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```



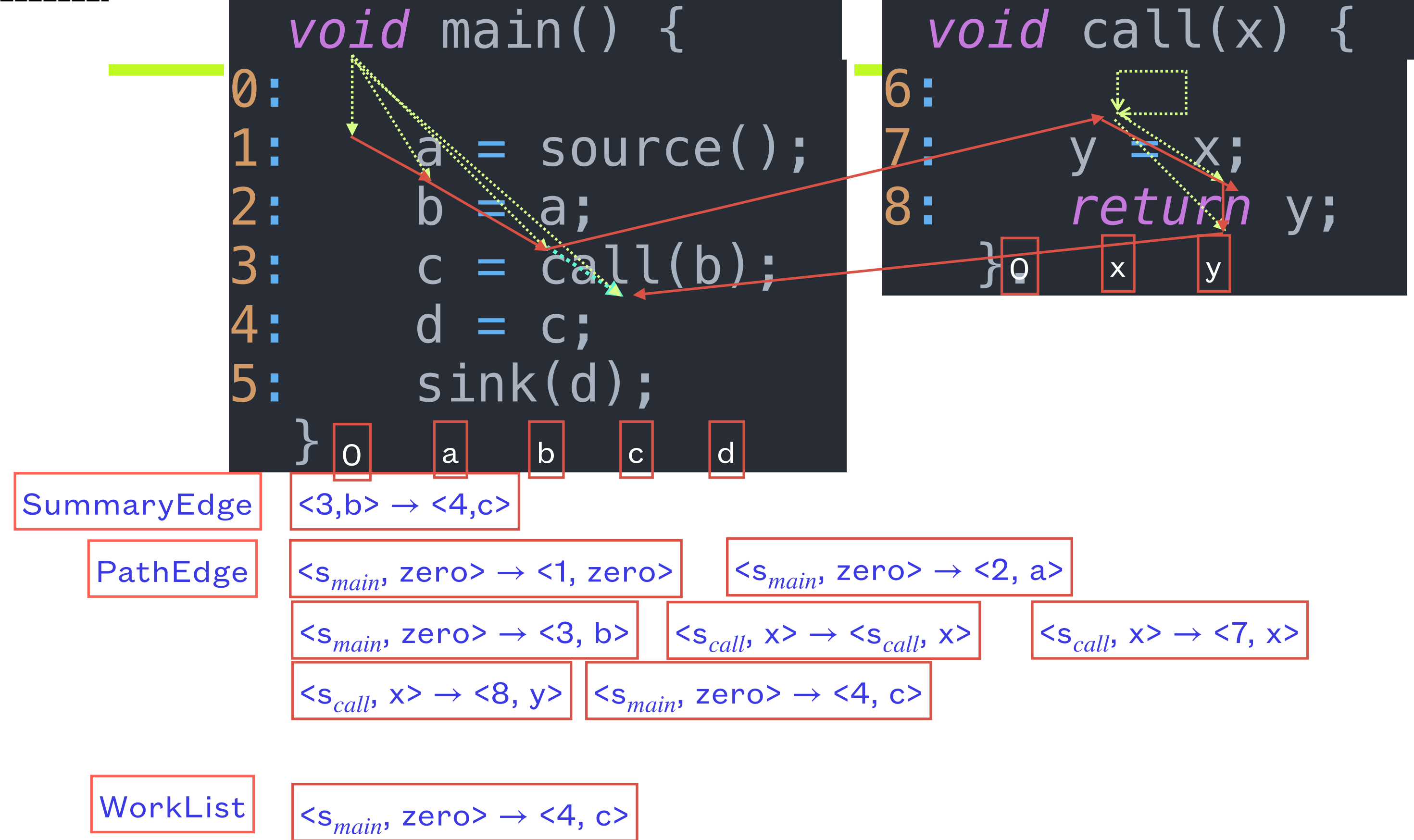

```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```



```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```

NormalFlowFunction



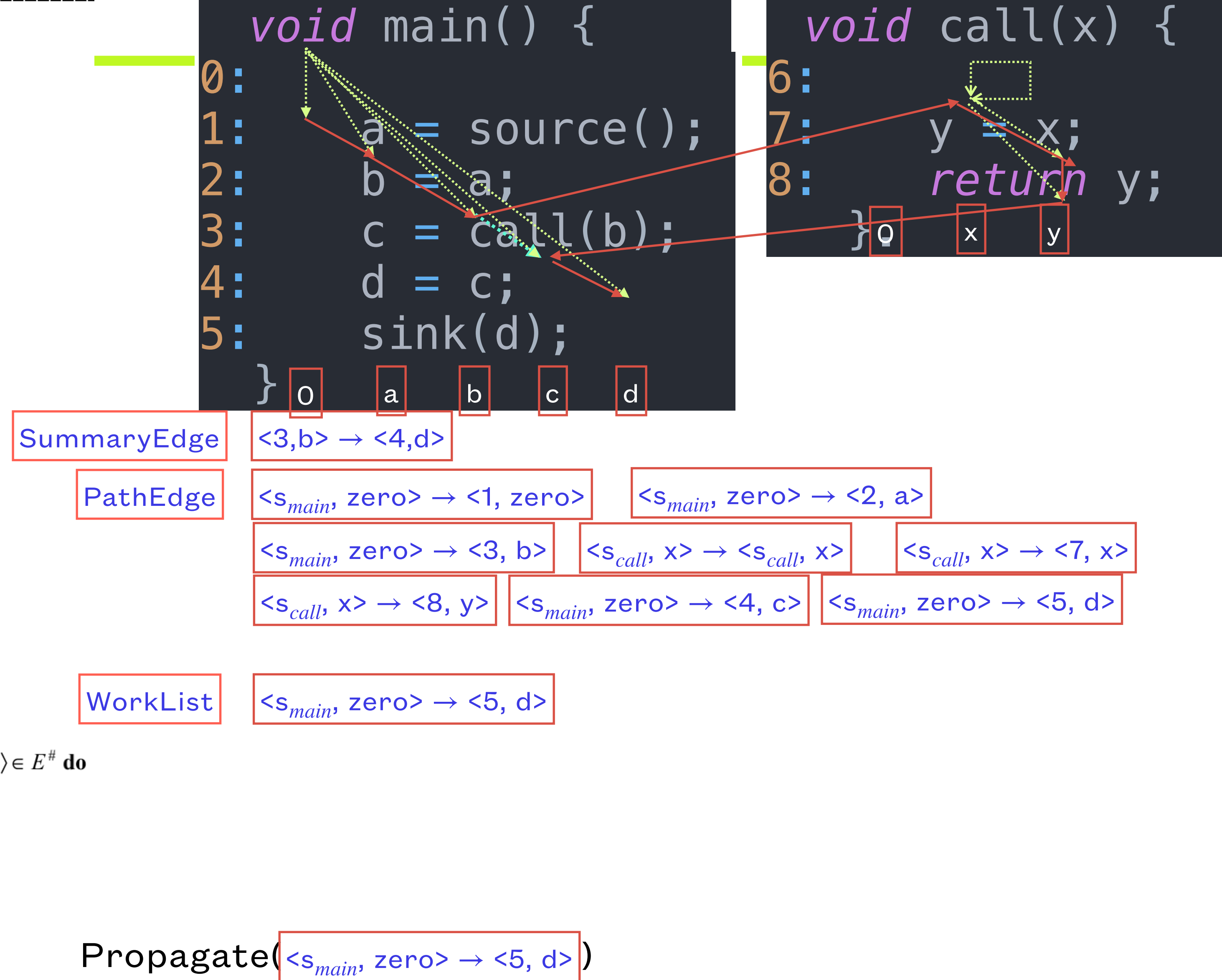

```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```



```

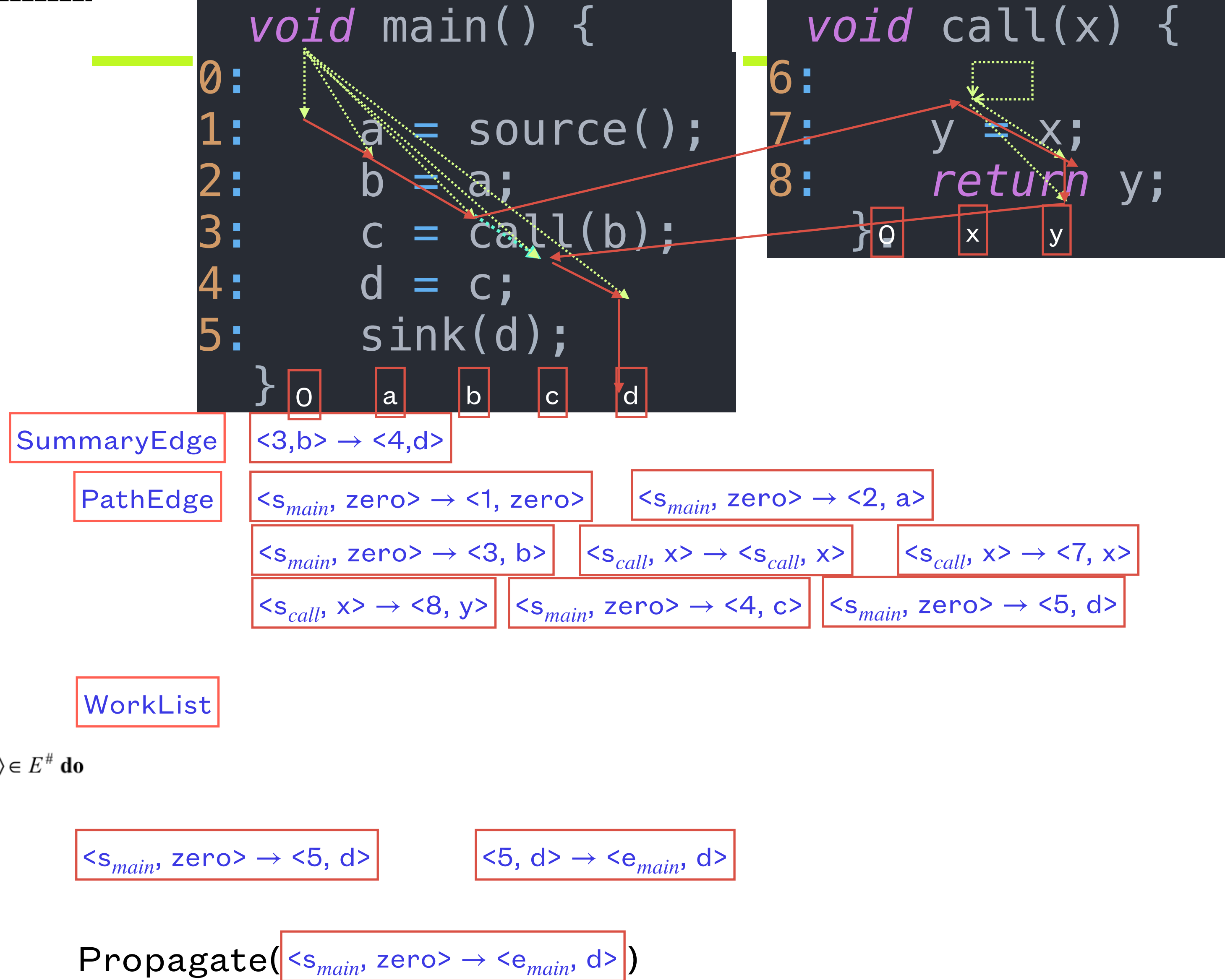
declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]            for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]              Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]            od
[29]          fi
[30]        od
[31]      od
[32]    end case
[33]    case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]      for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]        Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]      od
[37]    end case
[38]  end switch
[39] od
end

```

NormalFlowFunction



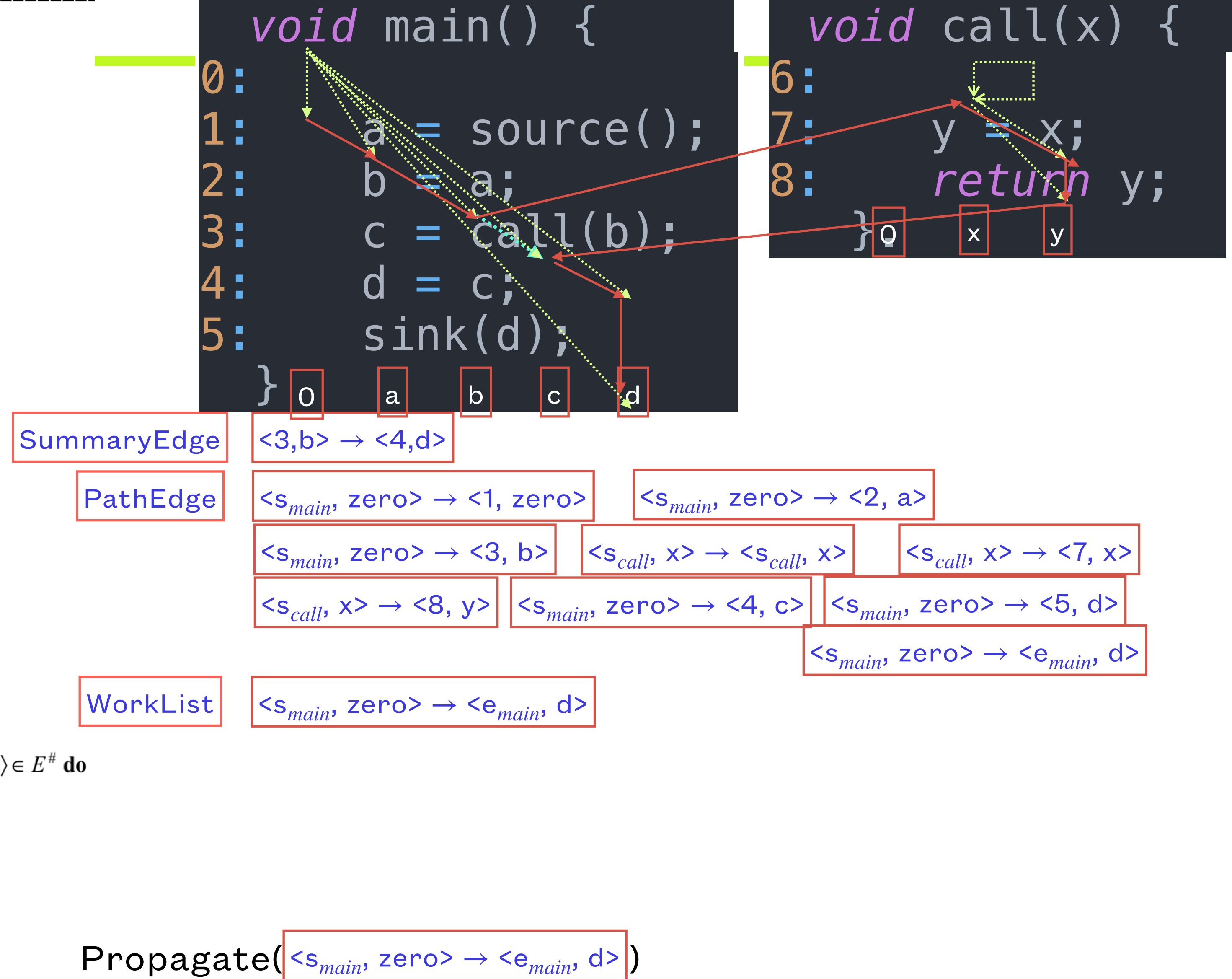

```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```



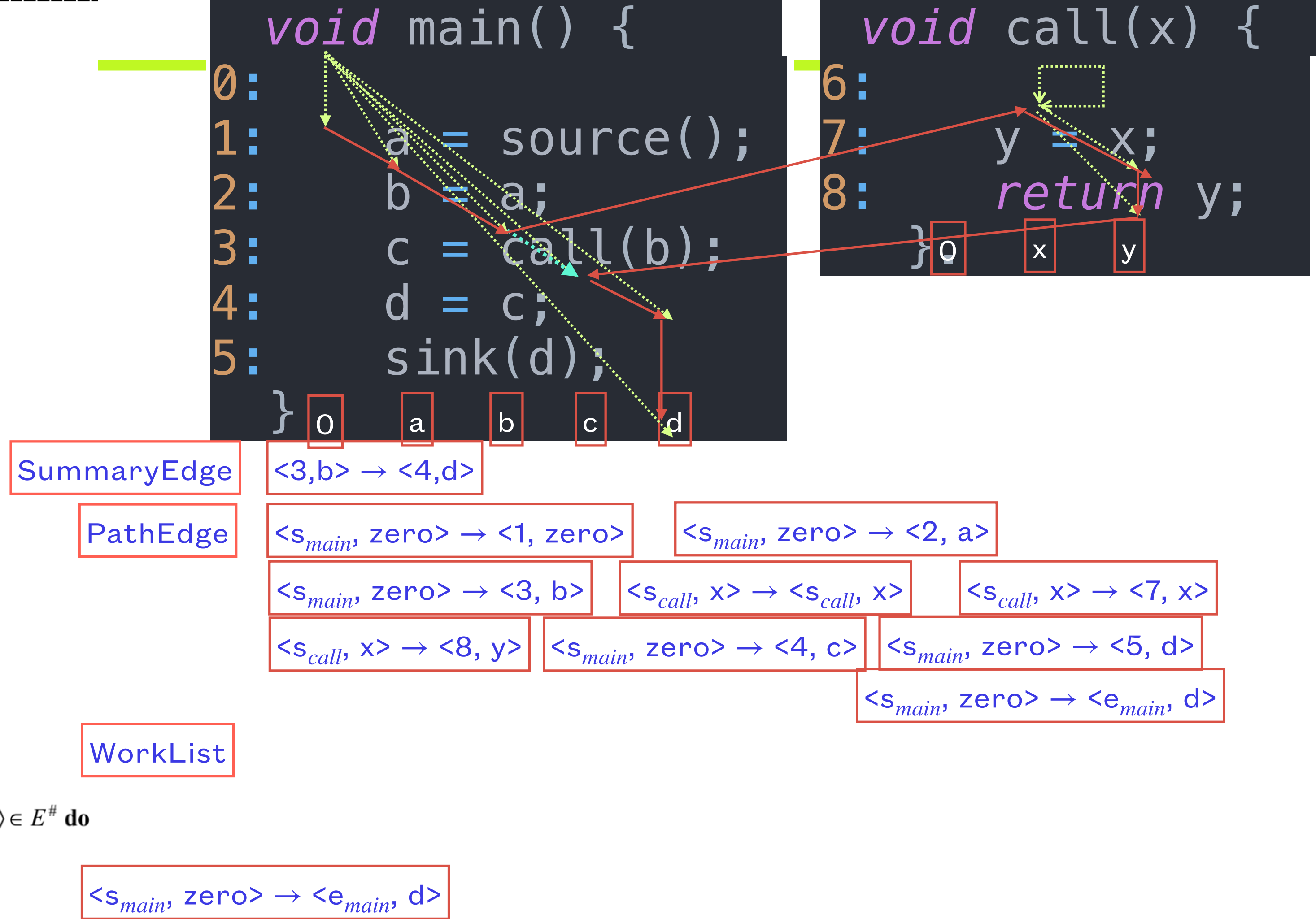
```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^\#$ )
begin
[1]   Let  $(N^\#, E^\#) = G_{IP}^\#$ 
[2]   PathEdge :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[3]   WorkList :=  $\{ \langle s_{main}, \mathbf{0} \rangle \rightarrow \langle s_{main}, \mathbf{0} \rangle \}$ 
[4]   SummaryEdge :=  $\emptyset$ 
[5]   ForwardTabulateSLRPs()
[6]   for each  $n \in N^*$  do
[7]      $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{ \mathbf{0} \}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8]   od
end

procedure Propagate( $e$ )
begin
[9]   if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end

procedure ForwardTabulateSLRPs()
begin
[10]  while WorkList  $\neq \emptyset$  do
[11]    Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]    switch  $n$ 
[13]      case  $n \in \text{Call}_p$  :
[14]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^\#$  do
[15]          Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]        od
[17]        for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^\# \cup \text{SummaryEdge})$  do
[18]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]        od
[20]      end case
[21]      case  $n = e_p$  :
[22]        for each  $c \in \text{callers}(p)$  do
[23]          for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^\#$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^\#$  do
[24]            if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]              Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]              for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]                Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]              od
[29]            fi
[30]          od
[31]        od
[32]      end case
[33]      case  $n \in (N_p - \text{Call}_p - \{ e_p \})$  :
[34]        for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^\#$  do
[35]          Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]        od
[37]      end case
[38]    end switch
[39]  od
end

```



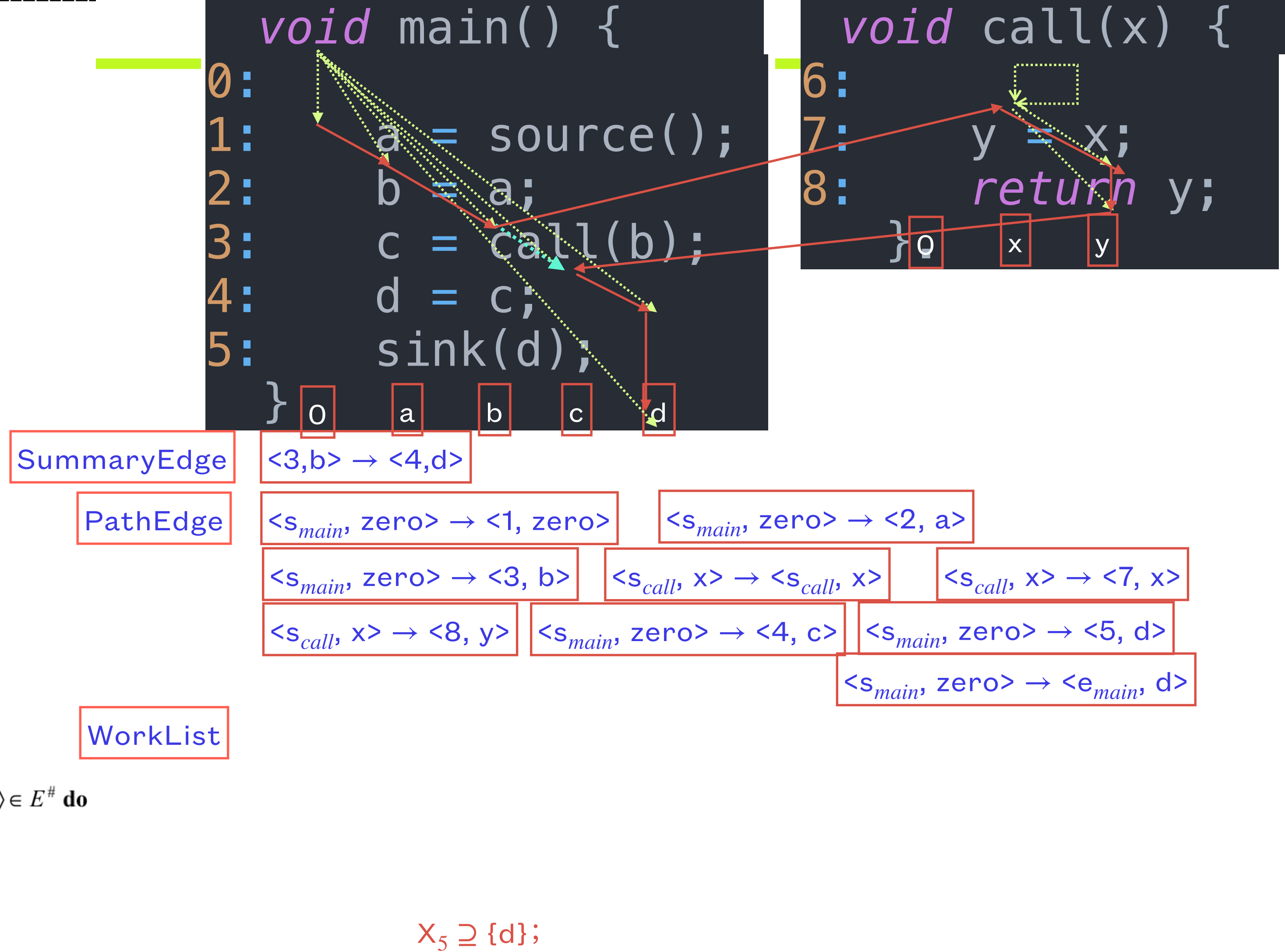
Callers(main) = \emptyset ;

算法终止!


```

declare PathEdge, WorkList, SummaryEdge: global edge set
algorithm Tabulate( $G_{IP}^{\#}$ )
begin
[1] Let  $(N^{\#}, E^{\#}) = G_{IP}^{\#}$ 
[2] PathEdge :=  $\{ \langle s_{main}, 0 \rangle \rightarrow \langle s_{main}, 0 \rangle \}$ 
[3] WorkList :=  $\{ \langle s_{main}, 0 \rangle \rightarrow \langle s_{main}, 0 \rangle \}$ 
[4] SummaryEdge :=  $\emptyset$ 
[5] ForwardTabulateSLRPs()
[6] for each  $n \in N^{\#}$  do
[7]    $X_n := \{ d_2 \in D \mid \exists d_1 \in (D \cup \{0\}) \text{ such that } \langle s_{procOf(n)}, d_1 \rangle \rightarrow \langle n, d_2 \rangle \in \text{PathEdge} \}$ 
[8] od
end
procedure Propagate( $e$ )
begin
[9] if  $e \notin \text{PathEdge}$  then Insert  $e$  into PathEdge; Insert  $e$  into WorkList fi
end
procedure ForwardTabulateSLRPs()
begin
[10] while WorkList  $\neq \emptyset$  do
[11]   Select and remove an edge  $\langle s_p, d_1 \rangle \rightarrow \langle n, d_2 \rangle$  from WorkList
[12]   switch  $n$ 
[13]     case  $n \in \text{Call}_p$  :
[14]       for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle \in E^{\#}$  do
[15]         Propagate( $\langle s_{calledProc(n)}, d_3 \rangle \rightarrow \langle s_{calledProc(n)}, d_3 \rangle$ )
[16]       od
[17]       for each  $d_3$  such that  $\langle n, d_2 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle \in (E^{\#} \cup \text{SummaryEdge})$  do
[18]         Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle \text{returnSite}(n), d_3 \rangle$ )
[19]       od
[20]     end case
[21]     case  $n = e_p$  :
[22]       for each  $c \in \text{callers}(p)$  do
[23]         for each  $d_4, d_5$  such that  $\langle c, d_4 \rangle \rightarrow \langle s_p, d_1 \rangle \in E^{\#}$  and  $\langle e_p, d_2 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \in E^{\#}$  do
[24]           if  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle \notin \text{SummaryEdge}$  then
[25]             Insert  $\langle c, d_4 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$  into SummaryEdge
[26]           for each  $d_3$  such that  $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle c, d_4 \rangle \in \text{PathEdge}$  do
[27]             Propagate( $\langle s_{procOf(c)}, d_3 \rangle \rightarrow \langle \text{returnSite}(c), d_5 \rangle$ )
[28]           od
[29]         fi
[30]       od
[31]     od
[32]   end case
[33]   case  $n \in (N_p - \text{Call}_p - \{e_p\})$  :
[34]     for each  $\langle m, d_3 \rangle$  such that  $\langle n, d_2 \rangle \rightarrow \langle m, d_3 \rangle \in E^{\#}$  do
[35]       Propagate( $\langle s_p, d_1 \rangle \rightarrow \langle m, d_3 \rangle$ )
[36]     od
[37]   end case
[38] end switch
[39] od
end

```



定义sink第0个参数为sink点，则找到污点缺陷