

Problem Sheet 6: Data Flow Analysis

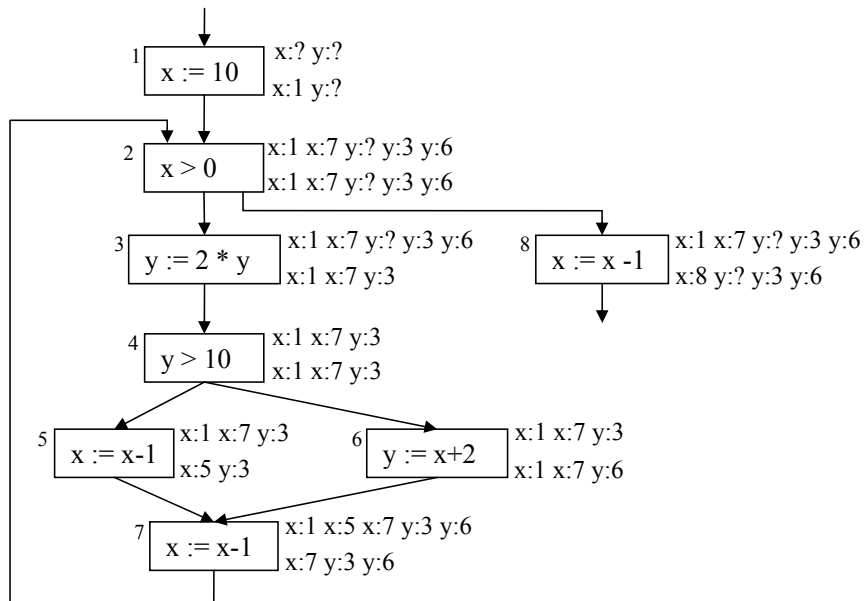
Sample Solutions

Chris Poskitt*
 ETH Zürich

Starred exercises (*) are more challenging than the others.

1 Reaching Definitions Analysis

i-ii. The control flow graph and the results of the reaching definitions analysis are given in the diagram below:



iii. We give the use-definition information for x and y in the table below (you could also annotate the diagram above with additional arrows).

*These sample solutions were adapted from previous iterations of the course when Stephan van Staden was the teaching assistant.

Program Block	x	y
1	\emptyset	\emptyset
2	$\{1, 7\}$	\emptyset
3	\emptyset	$\{?, 3, 6\}$
4	\emptyset	$\{3\}$
5	$\{1, 7\}$	\emptyset
6	$\{1, 7\}$	\emptyset
7	$\{1, 5, 7\}$	\emptyset
8	$\{1, 7\}$	\emptyset

2 Live Variables Analysis

- i. Below we identify the blocks of the program:

```

[x := y]1
[x := x-1]2
[x := 4]3
while [y < x]4 do
    [y := y+x]5
end
[y := 0]6
    
```

- ii. The system of equations for a live variable analysis are as follows:

$$\begin{aligned}
 LV_{\text{entry}}(1) &= (LV_{\text{exit}}(1) - \{x\}) \cup \{y\} \\
 LV_{\text{entry}}(2) &= (LV_{\text{exit}}(2) - \{x\}) \cup \{x\} \\
 LV_{\text{entry}}(3) &= LV_{\text{exit}}(3) - \{x\} \\
 LV_{\text{entry}}(4) &= LV_{\text{exit}}(4) \cup \{x, y\} \\
 LV_{\text{entry}}(5) &= (LV_{\text{exit}}(5) - \{y\}) \cup \{x, y\} \\
 LV_{\text{entry}}(6) &= LV_{\text{exit}}(6) - \{y\}
 \end{aligned}$$

$$\begin{aligned}
 LV_{\text{exit}}(1) &= LV_{\text{entry}}(2) \\
 LV_{\text{exit}}(2) &= LV_{\text{entry}}(3) \\
 LV_{\text{exit}}(3) &= LV_{\text{entry}}(4) \\
 LV_{\text{exit}}(4) &= LV_{\text{entry}}(5) \cup LV_{\text{entry}}(6) \\
 LV_{\text{exit}}(5) &= LV_{\text{entry}}(4) \\
 LV_{\text{exit}}(6) &= \emptyset
 \end{aligned}$$

- iii. We begin the iteration by initialising every set to \emptyset . Then, we iteratively update the sets by applying the equation system above. (For simplicity, the columns omit sets when a particular iteration does not update the previous value.)

LV Sets	Iterations \longrightarrow				Final Values
$LV_{\text{entry}}(1)$	\emptyset	$\{y\}$			$\{y\}$
$LV_{\text{entry}}(2)$	\emptyset	$\{x\}$		$\{x, y\}$	$\{x, y\}$
$LV_{\text{entry}}(3)$	\emptyset		$\{y\}$		$\{y\}$
$LV_{\text{entry}}(4)$	\emptyset	$\{x, y\}$			$\{x, y\}$
$LV_{\text{entry}}(5)$	\emptyset	$\{x, y\}$			$\{x, y\}$
$LV_{\text{entry}}(6)$	\emptyset				\emptyset
$LV_{\text{exit}}(1)$	\emptyset	$\{x\}$		$\{x, y\}$	$\{x, y\}$
$LV_{\text{exit}}(2)$	\emptyset		$\{y\}$		$\{y\}$
$LV_{\text{exit}}(3)$	\emptyset	$\{x, y\}$			$\{x, y\}$
$LV_{\text{exit}}(4)$	\emptyset	$\{x, y\}$			$\{x, y\}$
$LV_{\text{exit}}(5)$	\emptyset	$\{x, y\}$			$\{x, y\}$
$LV_{\text{exit}}(6)$	\emptyset				\emptyset

- iv. We eliminate blocks b of the form $[x := \dots]^b$ if x is not an element of $LV_{\text{exit}}(b)$:

```

[x := y]1
[x := 4]3
while [y < x]4 do
  [y := y+x]5
end
    
```

- v. (*) The program is not yet free of dead variables: x in block 1 is still dead. We strengthen the definition of LV_{entry} :

$$LV_{\text{entry}}(b) = \begin{cases} (LV_{\text{exit}}(b) - \text{kill}_{LV}(b)) \cup \text{gen}_{LV}(b) & \text{if } \text{kill}_{LV}(b) \subseteq LV_{\text{exit}}(b) \\ LV_{\text{exit}}(b) & \text{otherwise} \end{cases}$$

The rationale is this: if a block assigns to a variable that is not live afterwards, then it must be eliminated, and should not influence the analysis by adding the variables it reads to the live variable set.

Performing a chaotic iteration with this new equation yields the following results:

LV Sets	Final Values
$LV_{\text{entry}}(1)$	$\{y\}$
$LV_{\text{entry}}(2)$	$\{y\}$
$LV_{\text{entry}}(3)$	$\{y\}$
$LV_{\text{entry}}(4)$	$\{x, y\}$
$LV_{\text{entry}}(5)$	$\{x, y\}$
$LV_{\text{entry}}(6)$	\emptyset
$LV_{\text{exit}}(1)$	$\{y\}$
$LV_{\text{exit}}(2)$	$\{y\}$
$LV_{\text{exit}}(3)$	$\{x, y\}$
$LV_{\text{exit}}(4)$	$\{x, y\}$
$LV_{\text{exit}}(5)$	$\{x, y\}$
$LV_{\text{exit}}(6)$	\emptyset

with which we can eliminate all of the dead code in the program:

```

[x := 4]3
while [y < x]4 do
  [y := y+x]5
end
    
```