Lab Session 2 – Solutions
Operational Semantics and Interpreters

Interpretation and Compilation of Languages

Nova School of Science and Technology
Mário Pereira   mjp.pereira@fct.unl.pt

Version of April 6, 2025

1 Preamble – The While Language

Before we begin the exercises, let us recap the definition of the WHILE language as presented during lectures.

Syntax. Figure 1 gives the syntax of the WHILE language.

e	::=		EXPRESSIONS
	|	v	integer or Boolean constant
	|	x	variable
	|	$e \oplus e$	binary operator ($+$, $<$, ...)
s	::=		STATEMENTS
	|	`x:=e`	assignment
	|	`if (e) s else s`	conditional
	|	`while (e) s`	loop
	|	`s; s`	sequence
	|	`skip`	do nothing
v	::=		VALUES
	|	n	integer value
	|	b	Boolean value

Figure 1: Syntax of the While language.

Operational Semantics. As in the lectures, we define the big steps operational semantics for our WHILE language, using environments (functions from variables to values) which we denote by σ. For instance, $\sigma = \{a \mapsto 34, b \mapsto 55\}$ is an environment that maps 34 to the variable a and 55 to the variable b. We denote by $\sigma\{x \mapsto v\}$ the environment in which the value for x has been set to v.

We then state the big step operational semantics of WHILE by defining two relations $\sigma, e \Downarrow v$ and $\sigma, s \Downarrow \sigma$ that read respectively "in environment σ, expression e has value v" and "in environment σ, the evaluation of statement s terminates and leads to environment σ'". Figure 2 defines the operational semantics of the WHILE language, via inference rules (as explained in the course).

$$\frac{}{\sigma, n \Downarrow n} \qquad \frac{}{\sigma, b \Downarrow b} \qquad \frac{x \in \mathtt{dom}(\sigma)}{\sigma, x \Downarrow \sigma(x)}$$

$$\frac{\sigma, e_1 \Downarrow n_1 \quad \sigma, e_2 \Downarrow n_2 \quad n \stackrel{\text{def}}{=} n_1 \oplus n_2 \quad \oplus \in \{+, -, *, \ldots\}}{\sigma, e_1 \oplus e_2 \Downarrow n}$$

$$\frac{\sigma, e_1 \Downarrow b_1 \quad \sigma, e_2 \Downarrow b_2 \quad b \stackrel{\text{def}}{=} b_1 \oplus b_2 \quad \oplus \in \{<, \ldots\}}{\sigma, e_1 \oplus e_2 \Downarrow b}$$

Semantics for expressions

$$\frac{}{\sigma, \mathtt{skip} \Downarrow \sigma} \qquad \frac{\sigma, s_1 \Downarrow \sigma_1 \quad \sigma_1, s_2 \Downarrow \sigma_2}{\sigma, s_1\,;\,s_2 \Downarrow \sigma_2}$$

$$\frac{\sigma, e \Downarrow v}{\sigma, x \mathtt{:=} e \Downarrow \sigma\{x \mapsto v\}}$$

$$\frac{\sigma, e \Downarrow \mathtt{true} \quad \sigma, s_1 \Downarrow \sigma_1}{\sigma, \mathtt{if}\ (e)\ s_1\ \mathtt{else}\ s_2 \Downarrow \sigma_1} \qquad \frac{\sigma, e \Downarrow \mathtt{false} \quad \sigma, s_2 \Downarrow \sigma_2}{\sigma, \mathtt{if}\ (e)\ s_1\ \mathtt{else}\ s_2 \Downarrow \sigma_2}$$

$$\frac{\sigma, e \Downarrow \mathtt{true} \quad \sigma, s \Downarrow \sigma_1 \quad \sigma_1, \mathtt{while}\ (e)\ s \Downarrow \sigma_2}{\sigma, \mathtt{while}\ (e)\ s \Downarrow \sigma_2} \qquad \frac{\sigma, e \Downarrow \mathtt{false}}{\sigma, \mathtt{while}\ (e)\ s \Downarrow \sigma}$$

Semantics for statements

Figure 2: Big-step operational semantics of the While language.

1.1 Semantically Equivalent Programs

Two statements s_1 and s_2 of the WHILE language are said to be semantically equivalent if for all states σ and σ'

$$\sigma, s_1 \Downarrow \sigma' \qquad \text{if and only if} \qquad \sigma, s_2 \Downarrow \sigma'$$

Exercise 1. Show that the statement

$$\mathtt{if}\ (e)\ s_1\ \mathtt{else}\ s_2$$

is semantically equivalent to

$$\mathtt{if}\ (\neg\, e)\ s_2\ \mathtt{else}\ s_1$$

where $\neg\, e$ stands for the Boolean negation of the value of expression e. You may consider the following operational semantic rules for $\sigma, \neg\, e \Downarrow b$:

$$\frac{\sigma, e \Downarrow \mathtt{true}}{\sigma, \neg\, e \Downarrow \mathtt{false}} \qquad \frac{\sigma, e \Downarrow \mathtt{false}}{\sigma, \neg\, e \Downarrow \mathtt{true}}$$

Solution

□

Exercise 2. Show that the statement

$$\texttt{while}\,(\texttt{b})\ s$$

is semantically equivalent to

$$\texttt{if}\,(\texttt{b})\ s;\texttt{while}\,(\texttt{b})\ s\ \texttt{else skip}$$

Solution

□

1.2 Extensions to the While Language

Consider we extend the WHILE language with

- pair constructors (e, e) and get the left and the right pair components denoted respectively by `fst` e and `snd` e;
- `repeat` s `until` b loops.

The syntax of the language is extended as follows:

$$
\begin{array}{rlll}
e & ::= & & \textsc{Expressions} \\
 & \ldots & & \\
 & | & (e, e) & \text{pair construction} \\
 & | & \texttt{fst}\ e & \text{first projection} \\
 & | & \texttt{snd}\ e & \text{second projection} \\
 & & & \\
s & ::= & & \textsc{Statements} \\
 & \ldots & & \\
 & | & \texttt{repeat}\ s\ \texttt{until}\ e & \text{repeat loop} \\
 & & & \\
v & ::= & & \textsc{Values} \\
 & \ldots & & \\
 & | & (v, v) & \text{pair value}
\end{array}
$$

1.3 Pairs

Exercise 3. Complete the definition of $\sigma, e \Downarrow v$ by giving the inference rules for the operation semantics of pairs. Concretely, define the inference rule for evaluating (e_1, e_2) and the inference rules to evaluate `fst` e and `snd` e in a given environment σ.

Solution

□

Exercise 4. Give a derivation for the evaluation

$$\{x \mapsto (21, 34)\}, \texttt{if (fst}\ x > 0)\ x\texttt{:=(snd}\ x, \texttt{fst}\ x + \texttt{snd}\ x)\ \texttt{else}\ x\texttt{:=}0 \Downarrow \{x \mapsto (34, 55)\}$$

Solution

□

Exercise 5. Consider the following WHILE program:

```
i := 0;
p := 0;
while (i < 5)
  p := (i + 1, p);
  i := i + 1
```

Consider σ to be the final environment to which the above program evaluates. What is the value of $\sigma(\texttt{p})$?

A $(((((0, 1), 2), 3), 4), 5)$

B $(5, (4, (3, (2, (1, 0)))))$

C $(0, (1, (2, (3, (4, 5)))))$

D there is not such value, since the program produces a runtime error.

Solution

□

1.4 Repeat Loop

Exercise 6. We now consider the `repeat` s `until` e loop. Intuitively, its semantics is that we execute the statement s then check whether the condition e holds: if e evaluates to `true`, we leave the loop. Otherwise, e evaluates to `false`, and the loop is executed again. Define the inferences rules for each case of the `repeat` s `until` e.

Solution

□

Exercise 7. As one may notice, `repeat`-loops do not extend the power of the language. In fact, they can be simply rewritten (for instance by a compiler phase) using sequence and a `while` loop. Explain how this rewriting can be done. (You can simply write an equation `repeat` s `until` $e = \ldots$)

Solution

□

2 Solutions

Exercise 1, page 3

The resolution proceeds in two steps. First, we must show that

$$\sigma, \texttt{if (}e\texttt{)}\ s_1\ \texttt{else}\ s_2 \Downarrow \sigma' \implies \sigma, \texttt{if (}\neg\, e\texttt{)}\ s_2\ \texttt{else}\ s_1 \Downarrow \sigma'$$

which can be read as "*if from the initial environment* σ*, the command* $\texttt{if (}e\texttt{)}\ s_1\ \texttt{else}\ s_2$ *evaluates down to* σ'*, then it must also be the case that from the initial environment* σ*, the command* $\texttt{if (}\neg\, e\texttt{)}\ s_2\ \texttt{else}\ s_1$ *also evaluates down to* σ'".

The second step is to show that

$$\sigma, \texttt{if (}\neg\, e\texttt{)}\ s_2\ \texttt{else}\ s_1 \Downarrow \sigma' \implies \sigma, \texttt{if (}e\texttt{)}\ s_1\ \texttt{else}\ s_2 \Downarrow \sigma'.$$

First Step

For the first step, we need to decide first which rule was used to complete the derivation of $\sigma, \texttt{if (}e\texttt{)}\ s_1\ \texttt{else}\ s_2 \Downarrow \sigma'$. Since we have two rules for if..else commands, we must do some case analysis.

[Case $(\sigma, e \Downarrow \texttt{true})$**].** In this case we have

$$\sigma, e \Downarrow \texttt{true} \tag{H1}$$

and then the derivation for $\sigma, \texttt{if (}e\texttt{)}\ s_1\ \texttt{else}\ s_2 \Downarrow \sigma'$ is as follows:

$$\frac{\mathsf{H1} \qquad \sigma, s_1 \Downarrow \sigma'}{\sigma, \texttt{if (}e\texttt{)}\ s_1\ \texttt{else}\ s_2 \Downarrow \sigma'}$$

Let us refer to premise $\sigma, s_1 \Downarrow \sigma'$ as hypothesis (H2).

Using the new semantic rules proposed in the exercise for $\neg\, e$ expression, we can produce the following derivation:

$$\frac{\mathsf{H1}}{\sigma, \neg\, e \Downarrow \texttt{false}} \tag{H3}$$

We can now produce the following derivation:

$$\frac{\mathsf{H3} \qquad \mathsf{H2}}{\sigma, \texttt{if (}\neg\, e\texttt{)}\ s_2\ \texttt{else}\ s_1 \Downarrow \sigma'}$$

[Case $(\sigma, e \Downarrow \texttt{false})$**].** Similar.

Second Step

For the second step, we need to decide first which rule was used to complete the derivation of $\sigma, \texttt{if (}\neg\, e\texttt{)}\ s_2\ \texttt{else}\ s_1 \Downarrow \sigma'$. Since we have two rules for if..else commands, we must do some case analysis.

[Case $(\sigma, \neg\, e \Downarrow \texttt{true})$**].** In this case we have

$$\frac{\sigma, e \Downarrow \texttt{false}}{\sigma, \neg\, e \Downarrow \texttt{true}}$$

and then the derivation for $\sigma, \texttt{if}\ (\neg\, e)\ s_2\ \texttt{else}\ s_1 \Downarrow \sigma'$ is as follows:

$$\frac{\sigma, \neg\, e \Downarrow \texttt{true} \qquad \sigma, s_2 \Downarrow \sigma'}{\sigma, \texttt{if}\ (\neg\, e)\ s_2\ \texttt{else}\ s_1 \Downarrow \sigma'}$$

Let us refer to premise $\sigma, e \Downarrow \texttt{false}$ as hypothesis $(\mathsf{H1})$ and premise $\sigma, s_2 \Downarrow \sigma'$ as hypothesis $(\mathsf{H2})$.

We can now produce the following derivation:

$$\frac{\mathsf{H1} \qquad \mathsf{H2}}{\sigma, \texttt{if}\ (e)\ s_1\ \texttt{else}\ s_2 \Downarrow \sigma'}$$

[Case $(\sigma, \neg\, e \Downarrow \texttt{false})$**].** Similar.

Exercise 2, page 3

The resolution proceeds in two steps. First, we must show that

$$\sigma, \texttt{while}\ (b)\ s \Downarrow \sigma' \implies \sigma, \texttt{if}\ (b)\ s; \texttt{while}\ (b)\ s\ \texttt{else skip} \Downarrow \sigma'$$

which can be read as "*if from the initial environment* σ*, the command* ***while (e)*** s *evaluates down to* σ'*, then it must also be the case that from the initial environment* σ*, the command* ***if*** *(*b*)* s*;* ***while*** *(*b*)* s ***else*** skip *also evaluates down to* σ'".

The second step is to show that

$$\sigma, \texttt{if}\ (b)\ s; \texttt{while}\ (b)\ s\ \texttt{else skip} \Downarrow \sigma' \implies \sigma, \texttt{while}\ (b)\ s \Downarrow \sigma'$$

First Step

For the first step, we need to decide first which rule was used to complete the derivation of $\sigma, \texttt{while}\ (b)\ s \Downarrow \sigma'$. Since we have two rules for `while` commands, we must do some case analysis.

[Case $(\sigma, b \Downarrow \texttt{true})$**].** In this case we have

$$\sigma, e \Downarrow \texttt{true} \tag{H1}$$

and then the derivation for $\sigma, \texttt{while}\ (e)\ s \Downarrow \sigma'$ is as follows:

$$\frac{\mathsf{H1} \qquad \sigma, s \Downarrow \sigma'' \qquad \sigma'', \texttt{while}\ (b)\ s \Downarrow \sigma'}{\sigma, \texttt{while}\ (e)\ s \Downarrow \sigma'}$$

Let us refer to premise $\sigma, s \Downarrow \sigma''$ as hypothesis $(\mathsf{H2})$ and premise $\sigma'', \texttt{while}\ (b)\ s \Downarrow \sigma'$ as hypothesis $(\mathsf{H3})$.

We can now produce the following derivation:

$$\frac{\mathsf{H1} \qquad \dfrac{\mathsf{H2} \qquad \mathsf{H3}}{\sigma,\ s; \texttt{while}\ (b)\ s \Downarrow \sigma'}}{\sigma, \texttt{if}\ (b)\ s; \texttt{while}\ (b)\ s\ \texttt{else skip} \Downarrow \sigma'}$$

[Case $(\sigma, \mathsf{b} \Downarrow \texttt{false})$]. In this case we have

$$\sigma, \mathsf{e} \Downarrow \texttt{false} \tag{H1}$$

and then the derivation for $\sigma, \texttt{while (e) s} \Downarrow \sigma'$ is as follows:

$$\frac{\mathsf{H1}}{\sigma, \texttt{while (e) s} \Downarrow \sigma}$$

Hence, $\sigma = \sigma'$ which concludes the case by doing the following derivation:

$$\frac{\mathsf{H1} \qquad \sigma, \texttt{skip} \Downarrow \sigma}{\sigma, \texttt{if (b) s; while (b) s else skip} \Downarrow \sigma}$$

Second Step

For the second step, we need to decide first which rule was used to complete the derivation of $\sigma, \texttt{if (b) s; while (b) s else skip} \Downarrow \sigma'$.

Since we have two rules for `if..else` commands, we must do some case analysis.

[Case $(\sigma, \mathsf{b} \Downarrow \texttt{true})$]. In this case we have

$$\sigma, \mathsf{b} \Downarrow \texttt{true} \tag{H1}$$

and then the derivation for $\sigma, \texttt{if (b) s; while (b) s else skip} \Downarrow \sigma'$ is as follows:

$$\frac{\mathsf{H1} \qquad \dfrac{\sigma, \mathsf{s} \Downarrow \sigma'' \qquad \sigma'', \texttt{while (b) s} \Downarrow \sigma'}{\sigma, \texttt{s; while (b) s} \Downarrow \sigma'}}{\sigma, \texttt{if (b) s; while (b) s else skip} \Downarrow \sigma'}$$

Let us refer to premise $\sigma, \mathsf{s} \Downarrow \sigma''$ as hypothesis (H2) and premise $\sigma'', \texttt{while (b) s} \Downarrow \sigma'$ as hypothesis (H3).

We can now produce the following derivation:

$$\frac{\mathsf{H1} \qquad \mathsf{H2} \qquad \mathsf{H3}}{\sigma, \texttt{while (b) s} \Downarrow \sigma'}$$

[Case $(\sigma, \mathsf{b} \Downarrow \texttt{false})$]. In this case we have

$$\sigma, \mathsf{b} \Downarrow \texttt{false} \tag{H1}$$

and then the derivation for $\sigma, \texttt{if (b) s; while (b) s else } \Downarrow \sigma'$ is as follows:

$$\frac{\mathsf{H1}}{\sigma, \texttt{if (b) s; while (b) s else } \Downarrow \sigma}$$

Hence, $\sigma = \sigma'$ which concludes the case by doing the following derivation:

$$\frac{\mathsf{H1}}{\sigma, \texttt{while (b) s} \Downarrow \sigma}$$

Exercise 3, page 3

$$\frac{\sigma, e_1 \Downarrow v_1 \quad \sigma, e_2 \Downarrow v_2}{\sigma, (e_1, e_2) \Downarrow (v_1, v_2)} \qquad \frac{\sigma, e \Downarrow (v_1, v_2)}{\sigma, \texttt{fst}\ \ e \Downarrow v_1} \qquad \frac{\sigma, e \Downarrow (v_1, v_2)}{\sigma, \texttt{snd}\ \ e \Downarrow v_2}$$

Exercise 4, page 4

$$\cfrac{\cfrac{\cfrac{\{x \mapsto (21,34)\}, x \Downarrow (21,34)}{\{x \mapsto (21,34)\}, \texttt{fst}\ x \Downarrow 21} \qquad \{x \mapsto (21,34)\}, 0 \Downarrow 0}{\{x \mapsto (21,34)\}, \texttt{fst}\ x > 0 \Downarrow \texttt{true}} \qquad \mathcal{D}_2}{\{x \mapsto (21,34)\}, \texttt{if (fst}\ x > 0\texttt{)}\ x\texttt{:=(snd}\ x, \texttt{fst}\ x + \texttt{snd}\ x\texttt{) else}\ x\texttt{:=}0 \Downarrow \{x \mapsto (34,55)\}}$$

$\mathcal{D}_2$:

$$\cfrac{\cfrac{\cfrac{\{x \mapsto (21,34)\}, x \Downarrow (21,34)}{\{x \mapsto (21,34)\}, \texttt{snd}\ x \Downarrow 34} \qquad \mathcal{D}_3}{\{x \mapsto (21,34)\}, (\texttt{snd}\ x, \texttt{fst}\ x + \texttt{snd}\ x) \Downarrow (34,55)}}{\{x \mapsto (21,34)\}, x\texttt{:=(snd}\ x, \texttt{fst}\ x + \texttt{snd}\ x\texttt{)} \Downarrow \{x \mapsto (34,55)\}}$$

$\mathcal{D}_3$:

$$\cfrac{\cfrac{\{x \mapsto (21,34)\}, x \Downarrow (21,34)}{\{x \mapsto (21,34)\}, \texttt{fst}\ x \Downarrow 21} \qquad \cfrac{\{x \mapsto (21,34)\}, x \Downarrow (21,34)}{\{x \mapsto (21,34)\}, \texttt{snd}\ x \Downarrow 34} \qquad 55 \stackrel{\text{def}}{=} 21 + 34}{\{x \mapsto (21,34)\}, \texttt{fst}\ x + \texttt{snd}\ x \Downarrow 55}$$

Exercise 5, page 4

The correct answer is **B**.

Exercise 6, page 4

$$\frac{\sigma, s \Downarrow \sigma'' \qquad \sigma', e \Downarrow \texttt{false} \qquad \sigma'', \texttt{repeat}\ s\ \texttt{until}\ e \Downarrow \sigma'}{\sigma, \texttt{repeat}\ s\ \texttt{until}\ e \Downarrow \sigma'}$$

$$\frac{\sigma, s \Downarrow \sigma' \qquad \sigma', e \Downarrow \texttt{true}}{\sigma, \texttt{repeat}\ s\ \texttt{until}\ e \Downarrow \sigma'}$$

Exercise 7, page 4

$\texttt{repeat}\ s\ \texttt{until}\ e \stackrel{\text{def}}{=} s\texttt{;}\ \texttt{while}\ (\neg\ e)\ s$