Series 1

Exercise 4
We wish to add the following statement to the While language:

\[ \text{repeat } S \text{ until } b \]

The rules we add to the rules of natural semantics are:

- If \( B[b] \sigma' = \text{ff} \) then
  \[
  (S, \sigma) \rightarrow \sigma' \quad (\text{repeat } S \text{ until } b, \sigma') \rightarrow \sigma''
  \]
  \[
  (\text{repeat } S \text{ until } b, \sigma) \rightarrow \sigma''
  \]

- If \( B[b] \sigma' = \text{tt} \) then
  \[
  (S, \sigma) \rightarrow \sigma'
  \]
  \[
  (\text{repeat } S \text{ until } b, \sigma) \rightarrow \sigma'
  \]

Indeed, the meaning we want to give to this command is that we first perform \( S \) and then, according to whether \( b \) is true, we re-enter the repeat command or we stop.

Semantic equivalence proof
We prove that

- \( \text{repeat } S \text{ until } b \)
- and \( S; \text{if } b \text{ then skip else (repeat } S \text{ until } b) \)

are semantically equivalent.

To do this, we have to prove that for any states \( \sigma, \sigma' \) we have that \( (\text{repeat } S \text{ until } b, \sigma) \rightarrow \sigma' \) iff \( (S; \text{if } b \text{ then skip else (repeat } S \text{ until } b), \sigma) \rightarrow \sigma' \).

We first prove the \( \Rightarrow \) implication. We assume \( (\text{repeat } S \text{ until } b, \sigma) \rightarrow \sigma' \) and have to prove \( (S; \text{if } b \text{ then skip else (repeat } S \text{ until } b), \sigma) \rightarrow \sigma' \). Assuming \( (\text{repeat } S \text{ until } b, \sigma) \rightarrow \sigma' \) is assuming that there exists a derivation tree \( T \) whose conclusion is this statement. Two cases can arise:

- the tree \( T \) can be the following:
  \[
  (S, \sigma) \rightarrow \sigma_1 \quad (\text{repeat } S \text{ until } b, \sigma_1) \rightarrow \sigma'
  \]
  \[
  (\text{repeat } S \text{ until } b, \sigma) \rightarrow \sigma'
  \]

In this case, we know that \( \sigma_1 \) exists and that \( B[b] \sigma_1 = \text{ff} \).

We are searching for a tree \( T' \) whose conclusion is \( (S; \text{if } b \text{ then skip else (repeat } S \text{ until } b), \sigma) \rightarrow \sigma' \).

The program is the sequence of \( S \) and an if command. Such a tree \( T' \) would necessary look like:

\[
(S, \sigma) \rightarrow \sigma_2 \quad \text{(if } b \text{ then skip else (repeat } S \text{ until } b), \sigma_2) \rightarrow \sigma''
\]

\[
(S; \text{if } b \text{ then skip else (repeat } S \text{ until } b), \sigma) \rightarrow \sigma''
\]
for some candidate $\sigma_2$ we have to exhibit. If we look at the tree $T$, we see that we know $(S, \sigma) \rightarrow \sigma_1$. Hence we choose $\sigma_2 = \sigma_1$. Our tree $T'$ becomes:

\[
(S, \sigma) \rightarrow \sigma_1 \quad \frac{(\mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma_1) \rightarrow \sigma'}{(S; \mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma) \rightarrow \sigma'}
\]

We still have to replace $\sigma$, which we can do because we know that $B[b]_1 = \mathit{ff}$. Hence, we apply the if-false rule to derive a tree for $(S; \mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma) \rightarrow \sigma'$. $T'$ thus looks like:

\[
(S, \sigma) \rightarrow \sigma_1 \quad \frac{(\mathit{repeat} \ S \ \mathit{until} \ b, \sigma_1) \rightarrow \sigma_3}{(S; \mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma) \rightarrow \sigma'}
\]

for some $\sigma_3$ we have to find. Looking at $T$, we see that $\sigma_3 = \sigma'$ fits.

\[
(S, \sigma) \rightarrow \sigma_1 \quad \frac{(\mathit{repeat} \ S \ \mathit{until} \ b, \sigma_1) \rightarrow \sigma'}{(S; \mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma) \rightarrow \sigma'}
\]

is the derivation tree we are looking for. QED

- the tree $T$ can be the following:

\[
(S, \sigma) \rightarrow \sigma' \quad \frac{\mathit{(repeat} \ S \ \mathit{until} \ b, \sigma \rightarrow \sigma')}{\mathit{(repeat} \ S \ \mathit{until} \ b, \sigma \rightarrow \sigma')}
\]

In this case, we know that $\sigma'$ exists and that $B[b]_1 = \mathit{tt}$.

We are searching for a tree $T'$ whose conclusion is $(S; \mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma) \rightarrow \sigma'$. The program is the sequence of $S$ and an if command. Such a tree $T'$ would necessary look like:

\[
(S, \sigma) \rightarrow \sigma_1 \quad \frac{(\mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma_1) \rightarrow \sigma'}{(S; \mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma) \rightarrow \sigma'}
\]

for some candidate $\sigma_1$ we have to exhibit. If we look at the tree $T$, we see that we know $(S, \sigma) \rightarrow \sigma'$. Hence we choose $\sigma_1 = \sigma'$. Our tree $T'$ becomes:

\[
(S, \sigma) \rightarrow \sigma' \quad \frac{(\mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma') \rightarrow \sigma'}{(S; \mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma) \rightarrow \sigma'}
\]

We still have to replace $\sigma'$, which we can do because we know that $B[b]_1 = \mathit{tt}$. Hence, we apply the if-false rule to derive a tree for $(\mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma') \rightarrow \sigma'$. $T'$ thus looks like:

\[
(S, \sigma) \rightarrow \sigma' \quad \frac{(\mathit{skip}, \sigma') \rightarrow \sigma'}{(S; \mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma) \rightarrow \sigma'}
\]

It is the derivation tree we are looking for. QED

We then prove the $\Leftarrow$ implication. We assume $(S; \mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma) \rightarrow \sigma'$ and have to prove $(\mathit{repeat} \ S \ \mathit{until} \ b, \sigma) \rightarrow \sigma'$. Our assumption yields the existence of a derivation tree $T$ whose conclusion is $(S; \mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma) \rightarrow \sigma'$. It necessarily looks like:

\[
\begin{align*}
(S, \sigma) & \rightarrow \sigma_1 \\
(S; \mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma_1) & \rightarrow \sigma' \\
(S; \mathit{if} \ b \ \mathit{then} \ \mathit{skip} \ \mathit{else} \ (\mathit{repeat} \ S \ \mathit{until} \ b), \sigma) & \rightarrow \sigma'
\end{align*}
\]
\[(S, \sigma) \rightarrow \sigma_1 \quad \text{(if } b \text{ then skip else (repeat } S \text{ until } b), \sigma_1) \rightarrow \sigma'\]

\[ (S; \text{if } b \text{ then skip else (repeat } S \text{ until } b), \sigma) \rightarrow \sigma' \]

with an actual state \( \sigma_1 \). '?' depends on the truth value of \( b \) in state \( \sigma_1 \).

Two cases can arise:

- if \( B[b] \sigma_1 = \text{ff} \), we know that \( T \) is the following tree:

\[ \begin{align*}
(S, \sigma) \rightarrow \sigma_1 & \quad \text{(repeat } S \text{ until } b, \sigma_1) \rightarrow \sigma' \\
(S; \text{if } b \text{ then skip else (repeat } S \text{ until } b), \sigma) \rightarrow \sigma' \\
\end{align*} \]

We want to build a tree \( T' \) whose conclusion is \( (\text{repeat } S \text{ until } b, \sigma) \rightarrow \sigma' \). Such a tree necessarily ends with the application of one of the rules for the repeat command. We know that: \( (S, \sigma) \rightarrow \sigma_1 \), \( (\text{repeat } S \text{ until } b, \sigma_1) \rightarrow \sigma' \), and \( B[b] \sigma_1 = \text{ff} \). Hence, it is the repeat-true rule we use to build \( T' \):

\[ \begin{align*}
(S, \sigma) \rightarrow \sigma_1 & \quad (\text{repeat } S \text{ until } b, \sigma_1) \rightarrow \sigma' \\
(S; \text{if } b \text{ then skip else (repeat } S \text{ until } b), \sigma) \rightarrow \sigma' \\
\end{align*} \]

- Similarly, if \( B[b] \sigma_1 = \text{tt} \), we know that \( T \) is the following tree:

\[ \begin{align*}
(S, \sigma) \rightarrow \sigma_1 & \quad (\text{skip}, \sigma_1) \rightarrow \sigma' \\
(S; \text{if } b \text{ then skip else (repeat } S \text{ until } b), \sigma) \rightarrow \sigma' \\
\end{align*} \]

Moreover, according to the skip rule, \( \sigma_1 = \sigma' \).

We want to build a tree \( T' \) whose conclusion is \( (\text{repeat } S \text{ until } b, \sigma) \rightarrow \sigma' \). Such a tree necessarily ends with the application of one of the rules for the repeat command. We know that \( (S, \sigma) \rightarrow \sigma' \), and \( B[b] \sigma' = B[b] \sigma_1 = \text{tt} \). So we can build \( T' \) as follows:

\[ (S, \sigma) \rightarrow \sigma' \]

\[ (\text{repeat } S \text{ until } b, \sigma) \rightarrow \sigma' \]