Sémantique des Langages de Programmation (SemLP)
TD n° 2 : WP + Rewriting

Exercice 1 : Weakest Precondition for Imp (Ex. 12 in the course notes)

Let S be a statement and B an assertion. The weakest precondition of S with respect to B is a predicate that we denote with $wp(S, B)$ such that :

(i) $\{wp(S, B)\}\ S\ \{B\}$ is valid, and

(ii) if $\{A\}\ S\ \{B\}$ is valid then $A \supset wp(S, B)$ is valid.

Assuming that the statement S does not contain while loops. Propose a strategy to compute $wp(S, B)$ and derive a method to reduce the validity of the pca triple $\{A\}\ S\ \{B\}$ to the validity of a logical assertion.

Exercice 2 :

Provide an extension to the toy compiler of the course notes (Sec. 1.3) incorportanting the commands break and continue defined in the TD1.
Hint : Consider adding additional arguments to the compiler function C.

Exercice 3 :

Provide *simple* examples of rewriting systems satisfying, if possible, the following properties. Justify you answer in the case that no such system exists.

1. Terminating and not normalizing,

2. Normalizing and non-terminating,

3. Terminating and normalizing,

4. Confluent and non-terminating,

5. Church-Rosser and terminating.

Exercice 4 : (Ex. 25 in the course notes)

Let $(A, \rightarrow_1)$ and $(A, \rightarrow_2)$ be two rewriting systems. We say that they *commute* if $a \xrightarrow{*}_1 b$ and $a \xrightarrow{*}_2 c$ implies $\exists d,\ (b \xrightarrow{*}_2 d$ and $c \xrightarrow{*}_1 d)$ as shown below.

$$\begin{tikzcd} & a \arrow[dl, "*"', "1"] \arrow[dr, "*", "2"'] & \\ b \arrow[dr, dashed, "*", "2"'] & & c \arrow[dl, dashed, "*"', "1"] \\ & d & \end{tikzcd}$$

Show that if $\rightarrow_1$ and $\rightarrow_2$ are *confluent* and *commute* then $\rightarrow_1 \cup \rightarrow_2$ is confluent too.

Exercice 5 :

Consider the following Imp command (extended with integer addition and division) where b is an arbitrary boolean condition :

$$\mathsf{while}\ (u > l+1)\ \mathsf{do}\ (r := (u+l)/2;\ \mathsf{if}\ b\ \mathsf{then}\ u := r\ \mathsf{else}\ l := r)$$

Show that the evaluation of the command starting from any state where $u, l \in \mathbf{N}$ terminates.

Exercice 6 :

Does the evaluation of the following Imp commands terminate assuming that initially $m, n \in \mathbf{N}$?

1. $\mathsf{while}\ (m \neq n)\ \mathsf{do}\ (\mathsf{if}\ (m > n)\ \mathsf{then}\ m := m - n\ \mathsf{else}\ n := n - m)$

2. $\mathsf{while}\ (m \neq n)\ \mathsf{do}\ (\mathsf{if}\ (m > n)\ \mathsf{then}\ m := m - n\ \mathsf{else}\ (h := m; m := n; n := h))$

Exercice 7 :

Let Σ^* denote the set of finite words over the alphabet $\Sigma = \{f, g_1, g_2\}$ with generic elements $w, w', ...$ As usual, ϵ denotes the empty word. Let $\to$ denote the smallest binary relation on Σ^* such that for all $w \in \Sigma^*$:

$$(1)\ fg_1w \to g_1g_1ffw, \qquad (2)\ fg_2w \to g_2fw, \qquad (3)\ f \to \epsilon,$$

and such that if $w \to w'$ and $a \in \Sigma$ then $aw \to aw'$. Prove or give a counter-example to the following assertions :

1. If $w \xrightarrow{*} w_1$ and $w \xrightarrow{*} w_2$ then there exists w' such that $w_1 \xrightarrow{*} w'$ and $w_2 \xrightarrow{*} w'$.

2. The rewriting system $(\Sigma^*, \to)$ is terminating.

3. Replacing rule (1) with the rule $fg_1w \to g_1g_1fw$, the answers to the previous questions are unchanged.