Exercise 1. Consider the CGS below with two agents, two actions $a, b$ and the propositional variables $p, q, r$.

Determine which statements below hold.

1. $M, s_1 \models p \land \langle\langle 1 \rangle\rangle X q$
2. $M, s_2 \models \langle\langle 1 \rangle\rangle (p \lor r) U \neg q$
3. $M, s_1 \models \langle\langle 1 \rangle\rangle F \neg \langle\langle 2 \rangle\rangle X \neg p$
4. $M, s_1 \models \langle\langle 1 \rangle\rangle G p \land \langle\langle 2 \rangle\rangle G p \land \langle\langle 1, 2 \rangle\rangle F \neg p$
5. $M, s_2 \models \neg \langle\langle 1 \rangle\rangle X (q \land r) \land \neg \langle\langle 2 \rangle\rangle X \neg p \land \neg \langle\langle 1, 2 \rangle\rangle X (p \lor r)$
6. $M, s_3 \models \langle\langle 1 \rangle\rangle G \langle\langle 1, 2 \rangle\rangle (\neg q U p)$

Exercise 2. Let $M = (Agt, S, Act, act, \delta, L)$ be a concurrent game structure (CGS) $A, A' \subseteq Agt$ be coalitions such that $A \cap A' = \emptyset$, $s \in S$ and $\varphi, \varphi'$ be ATL formulae built over coalitions from $Agt$.

1. Show that if $M, s \models \langle\langle A \rangle\rangle G \varphi K M$ then $M, s \models \langle\langle A \cup A' \rangle\rangle G (\varphi \land \varphi')$.
2. Is it always the case that if $M, s \models \langle\langle A \rangle\rangle F \varphi \lor \langle\langle A' \rangle\rangle F \varphi'$ then $M, s \models \langle\langle A \cup A' \rangle\rangle F (\varphi \lor \varphi')$?

Exercise 3. Show that $\langle\langle A \rangle\rangle X \varphi \land \langle\langle A' \rangle\rangle X \varphi' \Rightarrow \langle\langle A \cup A' \rangle\rangle X (\varphi \land \varphi')$ is valid when $A \cap A' = \emptyset$.

Exercise 4. Let $M$ be a CGS, $\varphi, \psi$ be ATL formulae and $A \subseteq Agt$. Show the following characterisations.

1. $\llbracket \langle A \rangle G \varphi \rrbracket^M = \nu Z. (\llbracket \varphi \rrbracket^M \cap \text{pre}(M, A, Z))$.
2. $\llbracket \langle A \rangle \varphi U \psi \rrbracket^M = \mu Z. (\llbracket \psi \rrbracket^M \cup (\llbracket \varphi \rrbracket^M \cap \text{pre}(M, A, Z)))$. 
Exercise 5. Consider the concurrent game structure below with state space \( S \) and set of agents \{Robot_1, Robot_2\}.

1. Let \( F_{\text{Robot}_1} \) be the positional strategy for \( \text{Robot}_1 \) such that \( F_{\text{Robot}_1}(s_0) = \text{wait} \), \( F_{\text{Robot}_1}(s_1) = \text{push} \), \( F_{\text{Robot}_1}(s_2) = \text{wait} \). Then, determine the following sets of maximal computations

\[
\text{Comp}(s_0, F_{\text{Robot}_1}), \quad \text{Comp}(s_1, F_{\text{Robot}_1}), \quad \text{Comp}(s_2, F_{\text{Robot}_1}).
\]

Use \( \omega \)-regular expressions to define such sets of computations.

2. Let \( \text{Robot}_1 \) adopt the following memoryful strategy \( F_{\text{Robot}_1}^m \). Below, \( "F(E) = a" \) for a regular expression \( E \), indicates that the value of \( F \) for every element of \( E \) is \( a \). So, \( a \) is the action chosen by \( \text{Robot}_1 \) (below, we do not use anymore the notation with the joint action \( f \))

\[
F_{\text{Robot}_1}^m(s_0, s_1) = \text{wait}, \quad F_{\text{Robot}_1}^m(s_0, s_1)^* s_2 S^* = \text{push}.
\]

That is, the strategy prescribes waiting until the state \( s_2 \) is visited, if ever, and then pushing forever. Define a Büchi automaton \( B \) over the alphabet \( \Sigma = \{s_1, s_2, s_3\} \) such that the language of \( \omega \)-words accepted by \( B \) is the set of maximal computations \( \text{Comp}(s_1, F_{\text{Robot}_1}^m) \) (omitting the joint actions between two successive states). For instance, \( B \) should accept the word \( s_1 s_0 s_2^\omega \).