

HOMEWORK N. 2, MATHEMATICAL LOGIC FOR COMPUTER SCIENCE 2020/2021

ASSIGNMENT: DO 2 EXERCISES.

DEADLINE: APRIL 12, 2021.

1. EF-GAMES AND (NON)-EXPRESSIBILITY

Exercise 1.1. Prove using EF-games that the property $P =$ “having domain of finite cardinality” is not finitely expressible in any first-order language over the class of all structures. Is the property finitely expressible over finite structures?

Exercise 1.2. Consider the following two structures \mathfrak{G}_1 and \mathfrak{G}_2 for the language of graphs: Write at least



two sentences distinguishing the two structures. Discuss the EF-game played on these structures: for what k can the Duplicator win the k -rounds game? For what k can Spoiler win?

Exercise 1.3. Consider the property of being an Eulerian graph. This is equivalent to the fact that every node has even degree. Use the following structure to prove that the Eulerian (boolean) query is not finitely expressible in the language of graphs over finite graphs:

\mathfrak{G}_n : The set of vertices is $\{a_1, \dots, a_n, v, w\}$. The only edges are as follows: each a_i is connected by an edge to v and to w . Show that for every $m \leq n$ the Duplicator wins the m -moves game on \mathfrak{G}_n and \mathfrak{G}_{n+1} .

Exercise 1.4. Express the following queries with first-order sentences in the language of graphs:

- (1) G is a complete graph.
- (2) G has two isolated nodes.
- (3) G has degree at least 2.
- (4) G contains a path of length 4.

Exercise 1.5. Consider the following two structures (for the language of graphs):

- (1) \mathfrak{G}_1 is a cycle of length $4n$;
- (2) \mathfrak{G}_2 consists of two disjoint cycles of length $2n$.

Analyze the EF games on these structures for $n = 1, 2$ and write a sentence of minimal quantifier rank distinguishing the two structures for $n = 1, 2$.

BONUS: Formulate a generalization of your observations and prove that the Connectivity query is not-expressible in the language of graphs over finite graphs, using EF-games.

Exercise 1.6. Consider the following two graphs:

- (1) \mathfrak{G}_1 is a line of length $4n$;
- (2) \mathfrak{G}_2 consists of a line of length $2n$ and a cycle of length $2n$ (the two components are disjoint).

Analyze the EF games on these structures for $n = 1, 2$ and write a sentence of minimal quantifier rank distinguishing the two structures for $n = 1, 2$.

BONUS: Formulate a generalization of your observations and prove that the Acyclicity query is not-expressible in the language of graphs over finite graphs, EF-using games (a graph is acyclic iff it does not contain a cycle).

Exercise 1.7. Consider the proof of the following Theorem in Handout 8: if \mathfrak{A} and \mathfrak{B} are finite linear orders of size larger than 2^k then $\mathfrak{A} \equiv_k \mathfrak{B}$ in the language $\mathcal{L} = \{min, max, <\}$.

- (1) Show by an example that the proof can fail if the Duplicator answers a move of the Spoiler without guaranteeing conditions (1) and (2) in the Inductive Hypothesis but only condition (3) (partial isomorphism).
- (2) Does the proof work if \mathfrak{A} and \mathfrak{B} are of cardinality exactly 2^k and $2^k + 1$, respectively?
- (3) What can you say about (bounded) elementary equivalence of finite linear orders \mathfrak{A} and \mathfrak{B} if they have the same cardinality?

2. OTHER

Exercise 2.1. Let S be a class of formulas over a finite relational language \mathcal{L} . We denote by $BC(S)$ the class of all Boolean Combinations of formulas in S (i.e., the class of all formulas obtained by applying Boolean connectives \neg, \wedge, \vee to formulas in S). A class S of formulas is finite modulo logical equivalence if there exists a finite subset $S_0 \subseteq S$ such that for each $F \in S$ there exists a $G \in S_0$ such that $F \equiv G$. Show the following two points:

- (1) Let $T = BC(S)$. If two structures \mathfrak{A} and \mathfrak{B} satisfy the same formulas in S then they also satisfy the same formulas in T .
- (2) If S is finite modulo logical equivalence then T is finite modulo logical equivalence.

Exercise 2.2. Let \mathcal{L} be a finite vocabulary with no constant or function symbols. The atomic formulas in some fixed set of variables x_1, \dots, x_n are obviously finitely many. Let \mathfrak{A} be a structure adequate for \mathcal{L} and (a_1, \dots, a_n) and (b_1, \dots, b_n) be two vectors in A . We write $(a_1, \dots, a_n) \equiv_{at} (b_1, \dots, b_n)$ iff the two vectors satisfy the same atomic formulas in \mathfrak{A} . It is easy to observe that in this case they also satisfy the same quantifier-free formulas, a fact we denote by $(a_1, \dots, a_n) \equiv_{qf} (b_1, \dots, b_n)$.

Let T be a theory in \mathcal{L} . We say that T **admits Quantifier Elimination** if for all formulas of the form $\exists x_{n+1} F(x_1, \dots, x_n, x_{n+1})$ there exists a quantifier-free formula $G(x_1, \dots, x_n)$ that is T -equivalent to it.

We say that T **has the Extension Property** if the following holds: for all models \mathfrak{A} of T , for all (a_1, \dots, a_n) and (b_1, \dots, b_n) in A^n , if $(a_1, \dots, a_n) \equiv_{at} (b_1, \dots, b_n)$ then for any $a_{n+1} \in A$ there exists a $b_{n+1} \in A$ such that $(a_1, \dots, a_n, a_{n+1}) \equiv_{at} (b_1, \dots, b_n, b_{n+1})$.

- (1) Show that for each (a_1, \dots, a_n) in A^n there exists a formula $H(x_1, \dots, x_n)$ with free variables x_1, \dots, x_n such that for all (b_1, \dots, b_n) in A^n

$$(a_1, \dots, a_n) \equiv_{at} (b_1, \dots, b_n) \text{ if and only if } \mathfrak{A} \models H(x_1, \dots, x_n)[b_1, \dots, b_n].$$

- (2) Show that if T admits Quantifier Elimination then T has the Extension Property.

BONUS: Show that if T has the Extension Property and is semantically complete then T admits Quantifier Elimination.

Exercise 2.3. Show that the theory DLO has the Extension Property (defined in the previous Exercise).

Exercise 2.4. Show that the Random Graph Theory has the Extension Property (defined above).