Exercise 1: P-choice
A language $L$ is said P-peek ($L \in \mathbb{P}_p$) if there is a function $f : \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}^*$ computable in polynomial time such that $\forall x, y \in \{0,1\}^*$:
- $f(x, y) \in \{x, y\}$
- if $x \in L$ or $y \in L$ then $f(x, y) \in L$

$f$ is called the peeking function for $L$.

1. Show that $P \subseteq \mathbb{P}_p$
2. Show that $\mathbb{P}_p$ is closed under complementary
3. Show that if there exist $L \in \mathbb{NP}$-hard in $\mathbb{P}_p$, then $P = \mathbb{NP}$
4. Let $r \in [0,1]$ a real number, we define $L_r$ as the set of words $b = b_1...b_n \in \{0,1\}^*$ such that $0.b_1...b_n \leq r$. Show that $L_r \in \mathbb{P}_p$
5. Deduce that there exist a non-recursive language in $\mathbb{P}_p$

Exercise 2: Complete problems for levels of PH
Show that the following problem is $\Sigma^p_k$-complete (under polynomial time reductions).

$\Sigma_k^{\text{QBF}}$ : • INPUT : A quantified boolean formula $\psi := \exists X_1 \forall X_2 \exists...Q_k X_k \phi(X_1,...,X_k)$, where $X_1,...X_k$ are $k$ disjoint sets of variables, $Q_k$ is the quantifier $\forall$ if $k$ is even, and the quantifier $\exists$ if $k$ is odd, $\phi$ is a boolean formula over variables $X_1 \cup \cdots \cup X_k$;
- QUESTION : is the input formula true?
Define a similar problem $\Pi_k^{\text{QBF}}$ such that $\Pi_k^{\text{QBF}}$ is $\Pi^p_k$-complete.

Exercise 3: Oracle machines
Let $O$ be a language. A Turing machine with oracle $O$ is a Turing machine with a special additional read/write tape, called the oracle tape, and three special states : $q_{\text{query}}, q_{\text{yes}}, q_{\text{no}}$. Whenever the machine enters the state $q_{\text{query}}$, with some word $w$ written on the oracle tape, it moves in one step to the state $q_{\text{yes}}$ or $q_{\text{no}}$ depending on whether $w \in O$.

We denote by $P^O$ (resp. $\mathbb{NP}^O$) the class of languages decided in polynomial time by a deterministic (resp. non-deterministic) Turing machine with Oracle $O$. Given a complexity class $C$, we define $P^C = \bigcup_{O \in C} P^O$ (and similarly for $\mathbb{NP}$).

1. Prove that for any $C$-complete language $L$, $P^C = P^L$ and $\mathbb{NP}^C = \mathbb{NP}^L$.
2. Show that for any language $L$, $P^L = P^L$ and $\mathbb{NP}^L = \mathbb{NP}^L$.
3. Prove that if $\mathbb{NP} = P^{\mathbb{SAT}}$ then $\mathbb{NP} = \mathbb{coNP}$.

Exercise 4: Collapse of PH
1. Prove that if $\Sigma^p_k = \Sigma^p_{k+1}$ for some $k \geq 0$ then $PH = \Sigma^p_k$. (Remark that this is implied by $P = \mathbb{NP}$).
2. Show that if $\Sigma^p_k = \Pi^p_k$ for some $k$ then $PH = \Sigma^p_k$ (i.e. PH collapses).
3. Show that if $PH = \text{PSpace}$ then PH collapses.
4. Do you think there is a polynomial time procedure to convert any QBF formula into a QBF formula with at most 10 variables?

Exercise 5: Relativization
Show that there is an oracle $O$ such that $P^O = \mathbb{NP}^O$.