TD 11 – Eh bé, pépé !

Définition. Soit RP la classe[1] des langages L tels qu'il existe une machine de Turing probabiliste M en temps polynomial telle que pour tout $x \in \{0,1\}^*$,

$$x \in L \implies \mathbb{P}[M(x) = 1] \geq 2/3,$$
$$x \notin L \implies \mathbb{P}[M(x) = 0] = 1.$$

On définit également la classe PP par $x \in L \iff \mathbb{P}[M(x) = 1] > 1/2$.

Exercice 1. *Test d'identité de polynômes*

Un *circuit arithmétique* est un graphe orienté sans cycle ayant pour entrées des indéterminées x_i et la constante 1, et dont les portes sont $+$, $\times$ et $-$. Ainsi, un circuit arithmétique calcule un polynôme dans $\mathbb{Z}[x_1, \ldots, x_n]$. Le problème PolyNonNul est l'ensemble des circuits arithmétiques représentant un polynôme non identiquement nul.

1. Quel est le degré maximal d'un polynôme calculé par un circuit arithmétique de taille m ?
2. Montrer le lemme suivant :

   Lemme (Schwartz-Zippel)**.** *Soit $p(x_1, \ldots, x_n)$ un polynôme non indentiquement nul de degré au plus d, et S un ensemble fini d'entiers. Alors si $a_1, \ldots, a_n$ sont des entiers choisis aléatoirement (avec replacement) dans S, alors*

   $$\mathbb{P}[p(a_1, \ldots, a_n) = 0] \leq d/|S|.$$

3. Proposer un algorithme probabiliste qui décide le problème PolyNonNul avec probabilité d'erreur au plus $1/3$. Estimer son temps de fonctionnement.
4. Montrer que PolyNonNul $\in$ RP. **Indication.** On peut calculer *modulo* un entier n, et utiliser la fait qu'il y a $\pi(n) = O(n/\log n)$ nombres premiers inférieurs à n.

Exercice 2. Sat *alors !*

On considère les variantes suivantes du langage Sat :

$$\text{MajSat} = \{\phi : \phi \text{ est satisfaite par } > 1/2 \text{ de ses assignations}\}$$
$$\#\text{Sat} = \{(\phi, k) : \phi \text{ est satisfaite par } > k \text{ assignations}\}$$

1. Montrer que MajSat $\in$ PP.
2. Montrer que #Sat $\leq_p$ MajSat.
3. Montrer que #Sat et MajSat sont PP-complets.

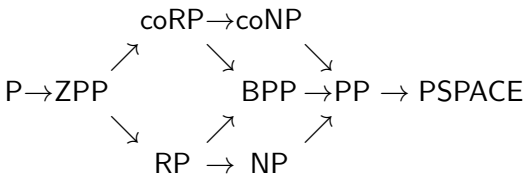
1. Rappelez-vous la définition de BPP et comparez.

Exercice 3. *Amusons nous avec les définitions...*

1. Montrer que dans la définition de BPP, on peut remplacer la constante $2/3$ par $f(|x|)$ où f est n'importe quelle fonction vérifiant $1/2 + n^{-c} \leq f(n) \leq 1 - 2^{-n^d}$ avec $c, d > 0$.
2. A-t-on les mêmes bornes pour RP ?
3. Soit $\mathsf{ZPP} = \mathsf{RP} \cap \mathsf{coRP}$. Montrer que ZPP est la classe des langages tels qu'il existe une MTP polynomiale qui donne la bonne réponse avec probabilité au moins $1/2$, et répond « je ne sais pas » sinon. Montrer qu'on définit encore la même classe avec des MTP qui donnent toujours la bonne réponse et dont l'espérance du temps de calcul est polynomiale.
4. Montrer que $\mathsf{RP} \subseteq \mathsf{NP}$.
5. Et $\mathsf{RP} \subseteq \mathsf{P/poly}$?

Exercice 4. *Dessins !*

Justifier toutes les inclusions suivantes :



Exercice 5. *Probablement Jivaro*

Un langage B **se réduit en temps polynomial probabiliste** à un langage C, noté $B \leq_r C$, s'il existe une MTP M tel que pour tout $x \in \{0,1\}^\star$, $\mathbb{P}[C(M(x)) = B(x)] \geq 2/3$. Pour une classe de complexité $\mathcal{C}$, on définit $\mathsf{BP} \cdot \mathcal{C} = \{L \subset \{0,1\}^\star : \exists C \in \mathcal{C}, L \leq_r C\}$.

1. La réduction probabiliste est-elle transitive ?
2. Montrer que si $B \leq_r C$ et $C \in \mathsf{BPP}$, alors $B \in \mathsf{BPP}$.
3. Montrer que $\mathsf{BP} \cdot \mathsf{P} = \mathsf{BPP}$.

Un **circuit non déterministe** est un circuit C à deux entrées x et y qui accepte un mot x s'il existe y tel que $C(x, y) = 1$.

4. Définir $\mathsf{NP/poly}$ et montrer que $\mathsf{BP} \cdot \mathsf{NP} \subseteq \mathsf{NP/poly}$.