IFT2105–Introduction à l'informatique théorique
Été 2021, (Devoir #1)

Louis Salvail

Université de Montréal (DIRO), QC, Canada
salvail@iro.umontreal.ca
Bureau: Pavillon André-Aisenstadt, #3369

1 Remise

Il s'agit du premier devoir pour le cours. La date de remise est:

Vendredi, 28 mai 2021, 23:59, aucun retard ne sera toléré.

Vous pouvez faire votre devoir en équipe de deux au maximum. Votre démo apprécie les devoirs remis en LaTeX. Un boni de 10% est donné si vous remettez un devoir produit par un traitement de texte. LaTeX est de loin le meilleur pour écrire des textes scientifiques (note: LaTeX a été écrit par Donald Knuth, le même qui a introduit les flèches qui portent son nom). Vous remettez votre devoir sur Studium en un seul fichier PDF. Une seule remise par équipe.

2 Questions

1. Donnez les programmes `RÉPÉTER` qui permettent de calculer les opérations suivantes sur des ensembles de nombres entiers. Vous pouvez utiliser n'importe quelles macros introduites au cours. Avant de donner votre code, expliquez comment vous représentez un ensemble d'entiers naturels dans un seul registre par un codage de Gödel. Ensuite, donnez un programme `RÉPÉTER` pour chacune des macros suivantes. Vous pouvez utilisez celles que nous avons vues en cours.
   Vide: Retourne dans r_0 un ensemble vide.
   EstVide?(r_1)**:** Retourne $r_0 = \texttt{vrai}$ si l'ensemble décrit par r_1 est vide et retourne $r_0 = \texttt{faux}$ sinon.
   Dans?(r_1, r_2)**:** En entrée, vous recevez un ensemble $S \subset \mathbb{N}$ dans r_1 et un entier ℓ dans r_2. Vous retournez $r_0 = \texttt{vrai}$ si $\ell \in S$ et $r_0 = \texttt{faux}$ si $\ell \notin S$.
   Card(r_1)**:** En entrée, vous recevez un ensemble $S \subset \mathbb{N}$ dans r_1 et vous retournez son cardinal dans r_0 (i.e. sa taille).
   Singleton(r_2)**:** En entrée, vous recevez un entier dans r_2 et vous retournez dans r_0 un ensemble qui ne contient que cet entier.
   Union(r_1, r_2)**:** En entrée, vous recevez deux ensembles $S, T \subset \mathbb{N}$ dans r_1 et r_2 respectivement et vous retournez $S \cup T$ dans r_0.
2. En utilisant l'induction mathématique, montrez que pour chaque entier naturel $x \geq 1$,

$$B_3(x) = 2^{x+3} - 3 \ .$$

   Pour y parvenir, vous pouvez supposer que $B_2(x) = 2x + 3$ pour $x \geq 0$.

3. La fonction $B_i(x)$ augmente à très grande vitesse en fonction de x, même pour de petites valeurs de i. Pour $i = 4$, la croissance est déjà monstrueuse. En effet, montrez que pour $x \geq 1$,
$$B_4(x) \geq \underbrace{2^{2^{\cdot^{\cdot^{\cdot^{2}}}}}}_{x+3 \text{ fois}} -3 \ ,$$
où le nombre de 2 empilés est $x + 3$.
4. Définissons un nouveau type de programme, les programmes `RÉPÉTERPASTROP`. Les programmes `RÉPÉTERPASTROP` sont identiques aux programmes `RÉPÉTER` à l'exception des boucles `RÉPÉTER` qui ne peuvent pas être imbriquées dans un programme `RÉPÉTERPASTROP`. Trouvez le plus petit $k \in \mathbb{N}$ tel que n^k ne peut pas être calculé par un programme `RÉPÉTERPASTROP` à partir de n'importe quel input n. Prouvez votre réponse à l'aide de ce qui a été vu en classe et en démo. Pour y parvenir, montrez que pour chaque programme `RÉPÉTERPASTROP`, il existe une valeur n_0 (qui dépend de la taille du programme) telle que pour chaque $n > n_0$, n^k ne peut pas être calculé par le programme. Ainsi, vous pouvez conclure qu'il n'existe pas de programme `RÉPÉTERPASTROP` qui puisse calculer la fonction n^k pour chaque $n \in \mathbb{N}$ donné en input.
5. Est-ce que les programmes `RÉPÉTER` qui n'ont pas d'opération $\mathtt{r_i} \leftarrow \mathtt{r_j}$, mais seulement une opération $\mathtt{r_i} \leftarrow \mathtt{0}$ ont la même puissance de calcul que les programmes `RÉPÉTER` standards? Prouvez votre réponse.