

Ah, le beau $\text{cro}(u)p$

3 février 2016

1 le background

Le traitement d'image par ordinateur est un domaine particulièrement foisonnant en terme d'algorithmes pour réaliser des traitements qui seraient fastidieux, voire impossible à la main. Une des opérations les plus utilisées est le cropping, qui permet par exemple de recadrer une photo. Généralement, les logiciels permettent de spécifier une boîte englobante de la zone à conserver grâce à la souris. Parfois, ils possèdent la capacité de déterminer eux-mêmes l'emplacement et la taille de cette boîte, grâce à des critères fournis par l'utilisateur.

2 l'énoncé

L'objectif est de réaliser une fonction de cropping automatique sur une image monochrome. Pour cela, votre programme doit lire sur l'entrée standard :

1. une ligne contenant un couple *largeur hauteur* donnant la taille de l'image.
2. *hauteur* lignes, faisant chacune *largeur* caractères, chaque caractère étant soit 0 (= pixel blanc), soit 1 (= pixel noir).

Une fois l'image lue, votre programme doit calculer la boîte englobant **tout les pixels noirs**. Les paramètres de cette boîte sont :

- le numéro de colonne et de ligne (en partant de 0) du pixel le plus en haut à gauche,
- la largeur et hauteur en pixels de cette boîte.

Votre programme doit ensuite écrire sur la sortie standard une seule ligne donnant ces paramètres, séparés par des espaces. Un exemple d'entrée et sortie attendue est donnée en figure 1

3 les ressources

Pour vous aider dans la réalisation du programme, vous trouverez sur <http://cours-info.iut-bm.univ-fcomte.fr>

entrée	sortie
10 5	2 2 6 2
0000000000	
0000000000	
0001100100	
0010001000	
0000000000	

TABLE 1 – Exemple d’entrée et la sortie attendue

un article dans la section **hackaton** → **édition 2016**, portant le même titre que l’exercice. Il contient un lien permettant de télécharger un canevas de code, ainsi que le fichier d’entrée donné ci-dessus.

Bien entendu, vous êtes libres d’utiliser ou non ce canevas, mais c’est un gain de temps que de s’en servir comme base.