

CNF informations

LAURENT Thomas

Master 2 informatique 2019

0.1 Ce que contient l'archive

__main__.py Le programme qui va traiter le fichier en format DIMACS.

dimacs.in Un fichier au format dimacs contenant le cnf de test.

README.pdf Coucou c'est moi.

0.2 Utilisation du script

Voici un exemple:

python__main__.py < fichierdimacs >

fichier dimacs et le répertoire vers le fichier à traiter

0.3 Procédures complexes

Au tout début le fichier est lue lignes par lignes, si celle ci contient (en premier caractère):

c on l'ignore

p on casse la ligne (en utilisant les espaces en tant que délimiteur) puis on récupère les deux derniers éléments (nb vars, nb lines)

sinon on casse la ligne (en utilisant les espaces en tant que délimiteur) en les cast en *Integer* le tout dans une liste

A l'issue de la lecture nous avons une matrice représentant la formule cnf, le nombre, le nombre de variables puis le nombre de lignes.

0.3.1 détection des 2cnf (binaire)

une *nbcnf2* (nom de la variable dans le code) est donné par l'instruction:

```
1 len(list(filter(lambda x : len(x) == 2, self.cnf)))
```

self.cnf est la matrice.

nous effectuons un filtre sur la matrice pour ne garder que les sous listes de tailles 2. Puis nous récupérons la taille de la liste une fois filtré.

0.3.2 détection des horn

une *nbhorn* (nom de la variable dans le code) est donné par l'instruction:

```
1 len(list(filter(lambda x : len(x) == 1, [list(filter(lambda x : x > 0, sl)) for sl in self.cnf])))
```

self.cnf est la matrice.

dans un premier temps:

```
1 [list(filter(lambda x : x > 0, sl)) for sl in self.cnf]
```

nous allons réduire la matrice en une autre matrice contenant que des littéraux positifs. (que nous allons nommer *M* dans la suite):

```
1 len(list(filter(lambda x : len(x) == 1, M)))
```

Puis nous filtrons les clauses en ne gardant que celle qui contienne que 1 élément, Puis nous récupérons la taille de la liste une fois filtré.

0.3.3 détection des reverse horn

une *nbreversehorn* (nom de la variable dans le code) est donné par l'instruction:

```
1 len(list(filter(lambda x : len(x) == 1, [list(filter(lambda x : x < 0, sl)) for sl in self.cnf])))
```

self.cnf est la matrice.

Le fonctionnement est identique que la détection des horn sauf que l'on ne garde que les littéraux négatifs.