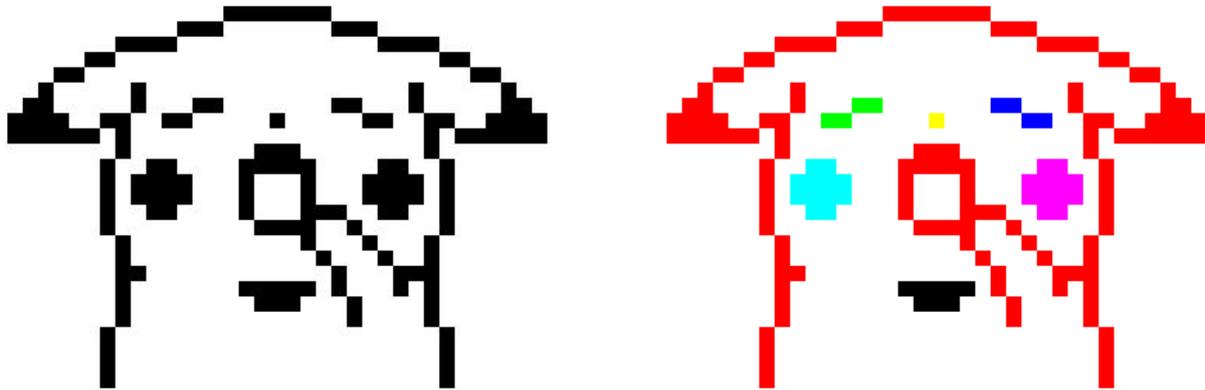


Composantes connexes

PCSI



I Algorithmique des graphes

On s'intéresse à des graphes dont les sommets sont des couples d'entier. On utilisera un dictionnaire d'adjacence pour les représenter.

On définit le type `vertex = tuple[int, int]` des couples d'entiers pour représenter les sommets du graphe.

Exercice 1. Distance

Implémentez une fonction `distance(g : dict, s1 : vertex, s2 : vertex) -> int` qui prend en entrée un graphe `g` et deux sommets, `s1` et `s2`, de ce graphe et qui renvoie la distance entre `s1` et `s2` dans `g`.

Exercice 2. Composante connexe

Une composante connexe d'un graphe non-orienté $G = (S, A)$ est un sous-ensemble $S' \subseteq S$ tel que :

1. Pour toute paire $(s, s') \in S'^2$, s' est accessible depuis s .
2. Pour toute paire $(s, s') \in S' \times S \setminus S'$, s' n'est pas accessible depuis s .

Pour $s \in S$, on note $\text{Acc}(s)$ l'ensemble des sommets accessibles depuis s . Justifier que pour toute composante connexe $S' \subseteq S$ non vide, et pour tout $s \in S'$, $S' = \text{Acc}(s)$.

Implémentez une fonction `accessible(g : dict, s : vertex) -> list[vertex]` qui prend en entrée un graphe `g` et un sommet, `s`, de ce graphe et qui renvoie la liste contenant tous les sommets de $\text{Acc}(s)$.

On utilisera un parcours en profondeur récursif.

Exercice 3. Composantes connexes

Implémentez une fonction `composantes(g : dict) -> list[list[vertex]]` qui prend en entrée un graphe `g` et qui renvoie la liste contenant toutes les composantes connexe du graphe.

II Graphe sur une image

On considère une variable globale proposant douze couleurs non blanches.

```
COULEURS = [ (255,0,0), (0,255,0), (0,0,255), (255,255,0), (0,255,255),  
             (255,0,255), (0,0,0), (120,0,120), (255,120,120), (120,255,120),  
             (120,120,255), (120,120,120) ]
```

On définit le type `color = tuple[int,int,int]` des triplets d'entiers.

Exercice 4. Graphe d'image

Implémentez une fonction `grapheIMG1(m : list[list[bool]]) -> dict` qui prend en entrée une matrice représentant une image en noir et blanc et qui renvoie le dictionnaire d'adjacence du graphe $G = (S, A)$ tel que S est l'ensemble des couples (i, j) des coordonnées des pixels noirs de l'image et chaque arête de A relie deux pixels noirs adjacents, *i.e* à 1 pixels de décalage en abscisse ou en ordonnée.

Exercice 5. Coloration des composantes connexes.

Implémentez une fonction `coloration_cc(m : list[list[bool]]) -> list[list[color]]` qui prend en entrée une matrice représentant une image en noir et blanc et qui renvoie une matrice représentant une image en couleur telle que :

1. les pixels initialement blanc restent blanc.
2. les pixels initialement noirs ne sont pas blanc.
3. deux pixels initialement noirs sont de la même couleur ssi ils sont dans la même composante connexe dans le graphe de l'image.