

# Interfaces Graphiques

## TP 3 – Graphes colorés

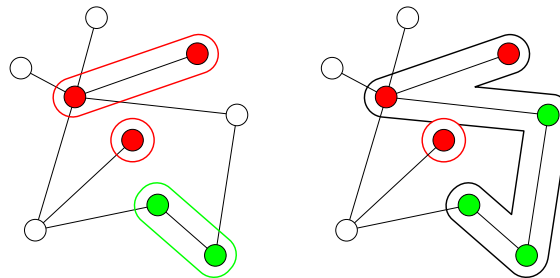
### 1 Graphes colorés

On considère les graphes non orientés dont les sommets sont étiquetés par des couleurs : blanc, rouge, vert... Une seule de ces couleurs est appelée *couleur neutre*.

#### 1.1 Composantes colorées

Un sous-ensemble non vide des sommets d'un graphe est appelée *composante colorée* si ses éléments sont tous de couleurs non neutres, s'ils forment un sous-graphe connexe (pour tout couple de sommets de cet ensemble, il existe une suite d'arêtes menant de l'un à l'autre) et si cet ensemble est maximal pour cette propriété (tout autre sommet du graphe est soit de couleur neutre, soit n'est connecté à aucun sommet de cet ensemble).

Une composante colorée est *monochrome* si tous ses sommets sont de la même couleur, et *polychrome* sinon. Les deux figures ci-dessous montrent deux graphes de même structure sous-jacente mais de colorations distinctes. Le blanc y est considéré comme couleur neutre. Le premier contient trois composantes monochromes, le second une composante monochrome et une composante polychrome.

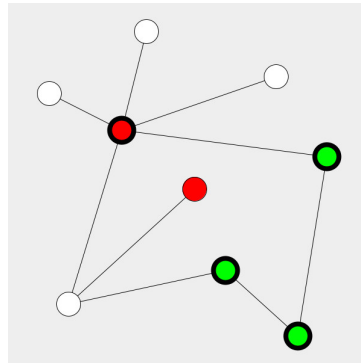


### 2 Implémentation des graphes colorés en Java

Le but de ce mini-projet est d'écrire un programme Java permettant de construire en mémoire la représentation d'un graphe non orienté à partir de sa structure encodée par un fichier texte (.gr). Un second fichier texte (.xy) sera ensuite lu pour charger en mémoire les informations relatives à l'affichage de cette représentation : dimensions du panneau d'affichage, taille des cercles représentant les nœuds, coordonnées initiales des cercles.

Le programme commencera par afficher le graphe à l'écran avec tous ses nœuds de couleur neutre. L'utilisateur doit pouvoir à tout moment modifier librement les coordonnées d'un nœud, ou changer sa couleur – on supposera l'ensemble des couleurs réduit à quatre couleurs de numéros 0 (neutre), 1, 2, 3. Les deux sortes d'actions doivent être réalisables à l'aide de la seule souris.

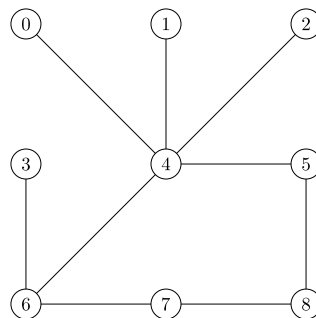
L’affichage sera mis à jour en continu pendant un déplacement de sommet. Les sommets seront affichés de leur couleur. Ceux faisant partie d’une composante polychrome seront mis en valeur par exemple comme ceci :



## 2.1 Représentation des graphes

La représentation interne des graphes non orientés colorés est libre, mais votre modélisation devra respecter la contrainte suivante : la ou les classes représentant un graphe ne devront avoir aucune connaissance de la manière dont ce graphe est rendu à l’écran. Leurs instances ne doivent contenir que des informations relatives à la structure du graphe : sommets, arêtes, numéro de couleur courant de chaque sommet, rien de plus.

## 2.2 Fichiers de graphes (.gr)



Un fichier de graphe est un fichier texte spécifiant la structure d’un graphe non orienté de la manière suivante :

- Il commence par deux entiers  $N$  et  $M$  spécifiant son nombre de sommets et son nombre d’arêtes. Les sommets sont implicitement numérotés de 0 à  $(N - 1)$ .
- Ces entiers sont suivis de  $M$  couples d’entiers spécifiant pour chaque arête les numéros de ses extrémités.

Voici par exemple le contenu du fichier spécifiant le graphe ci-dessus (9 sommets, 9 arêtes) :

```
9 9  4 0  4 1  4 2  4 5  4 6  3 6  6 7  7 8  8 5
```

## 2.3 Fichiers d’affichage (.xy)

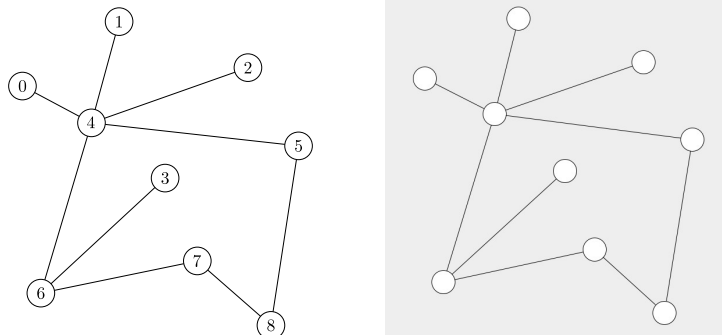
Un fichier d’affichage pour un graphe est un fichier de structure suivante :

- Il commence par trois entiers  $W$ ,  $H$ ,  $R$ ,  $N$  spécifiant respectivement la largeur et la hauteur du panneau d’affichage de la représentation du graphe, le rayon des cercles représentant ses sommets, et le nombre de sommets. Comme pour un fichier `.gr`, les sommets sont implicitement numérotés de 0 à  $(N - 1)$ .
- Ces nombres sont suivis de  $N$  couples de nombres (éventuellement décimaux) spécifiant pour chaque numéro de sommet les coordonnées  $(x, y)$  du centre du cercle représentant ce sommet dans le panneau.

Voici par exemple le contenu d’un fichier `.xy` pour le graphe décrit par le `.gr` précédent :

```
600 600 20 9
68.46030838311843 150.5393256823958
229.2747469898672 47.508705798495676
444.00122453121946 122.00868106741014
309.6620109038187 309.9235933992489
188.99999993360672 211.00005698085994
528.9999998671623 255.0028638870451
100.0 500.0
360.7972801674691 444.4214856701663
480.4232393533589 553.963152160205
```

Ces coordonnées correspondent au placement et à la capture d’écran suivante :



## 2.4 Conseils de réalisation

Le plus simple pour la lecture des fichiers `.gr` et `.xy` est de se servir de la classe `java.util.Scanner` – servez-vous de la méthode `setDelimiter` en spécifiant comme délimiteur les suites de caractères d’espace.

L’algorithme permettant de déterminer si un sommet fait partie d’une composante polychrome demande un peu de réflexion préliminaire.

## 2.5 Si vous ne savez plus quoi faire

Ajoutez un bouton permettant de sauvegarder un fichier `.xy` décrivant l’affichage du graphe dans son état courant. Servez-vous par exemple de `java.io.FileWriter`.