# 5-coloration de graphe en temps linéaire

Sujet proposé par François Morain

morain@lix.polytechnique.fr

**URL de suivi :** http://www.enseignement.polytechnique.fr/profs/informatique/Francois.Morain/INF431/projetX06.html

## 1 Préambule

La coloration des graphes est un problème théorique fascinant, avec des applications dans le monde réel, comme par exemple l'allocation de fréquences sur les stations de base des réseaux cellulaires.

Le coloriage des graphes planaires a attiré et continue à attirer de nombreux chercheurs, même au-delà de la preuve du fameux théorème des quatre couleurs. On sait par exemple depuis plus longtemps que colorier avec 5 couleurs est faisable pour tout graphe planaire et depuis une trentaine d'années qu'on peut le faire en temps linéaire en le nombre de sommets.

Le but de ce projet est de colorier des graphes planaires à l'aide de cinq couleurs, en implantant un algorithme de complexité *linéaire* en le nombre de sommets.

On a vu en cours quelques algorithmes linéaires pour résoudre certains problèmes de graphe, et ceux-ci sont généralement extraordinairement complexes à programmer et valider. Dans le cas présent, la linéarité dépend d'un résultat mathématique (utilisant la formule d'Euler), ainsi que de l'utilisation de structures de données spécifiques et astucieuses.

## 2 Détail du sujet

### 2.1 Un peu de théorie

Un *coloriage* des sommets de $S$ est une application $\varphi\ : S \rightarrow \mathbf{N}$ tel que pour tout $(u, v) \in A$, $\varphi(u) \neq \varphi(v)$. Le problème le plus classique est celui où on cherche à minimiser le nombre de couleurs nécessaires pour colorier les sommets d'un graphe donné. On sait que le nombre minimal de couleurs est 4 pour un graphe plan (résultat difficile), et que 5 et 6 sont plus faciles et réalisables en temps linéaire en $n$.

Soit $G = (S, A)$ un graphe plan à $n$ sommets et $m$ arêtes, c'est-à-dire un graphe planaire équipé d'un plongement dans le plan. Dans la pratique, les algorithmes de planarité fournissent en sortie un tel plongement, qui est représenté en fait par la liste des sommets, chaque sommet étant accompagné de ses voisins dans l'ordre des aiguilles d'une montre.

L'idée de tous ces algorithmes est le suivant : on retire un sommet $x$ de $G$, on colorie le graphe $G' = G - \{x\}$, puis on s'arrange pour prolonger le coloriage à $G$. D'après la formule d'Euler, on a $m \leq 3n - 6$. D'après le *handshaking lemma*, on a aussi $\sum_s \deg(s) = 2m$, ce qui fait qu'il existe un sommet de degré $< 6$. C'est celui-là qu'on doit enlever.

Les détails des algorithmes diffèrent alors. Ici, nous demandons d'implanter l'algorithme contenu dans [1] et disponible sur la page de suivi.

### 2.2 Un exemple

On considère le graphe suivant au format de `grapheX` :

```
#
12 s
0 1 2 3 4 5 6 7 8 9 10 11
0 5 1 2 3 4 5
1 5 0 5 6 7 2
2 5 0 1 7 8 3
3 5 0 2 8 9 4
4 5 0 3 9 10 5
5 5 0 4 10 6 1
6 5 1 5 10 11 7
7 5 1 6 11 8 2
8 5 2 7 11 9 3
9 5 3 8 11 10 4
10 5 4 9 11 6 5
11 5 6 10 9 8 7
```

L'algorithme permet de le colorier comme dans la figure 1.

# 3 Travail demandé

## 3.1 Prérequis

Dans un premier temps, il faut récupérer l'article [1] contenant la théorie et l'algorithme à implanter sur la page de suivi.

On récupérera également le programme `plantri` à partir de la même page. Celui-ci permet de fabriquer des graphes planaires de nombre de sommets et degré donné. Ce programme se compile aisément et fournit des graphes sous différents formats planaires qui permettent de les donner comme input au programme.

De la même façon le programme `PlanarMap` de G. Schaeffer permet de générer des graphes plans aléatoires. Il est conseillé, pour celui-ci, d'utiliser des commandes du type

```
./planarmap -Q3 -V10 -p -O1
```

La sortie peut être :

```
Vertex 1: 1 2 3 4
Vertex 2: -1 5 6 7
Vertex 3: -4 8 9 -5
Vertex 4: 10 11 -8 -3
Vertex 5: -7 12 -10 -2
Vertex 6: -9 -11 -12 -6
```

et doit être interprété comme suit : les nombres après le : sont les indices des *demi-arêtes* entre les sommets, classés dans l'ordre trigonométrique autour du sommet. Le graphe plan correspondant est celui de la figure 2.

Attention, pour des raisons théoriques, le nombre final de sommets sera un peu inférieur à 10. On pourra tester également :

```
./planarmap -Q3 -V10 -d -O1
```

et utiliser les nouveaux numéros de sommets.

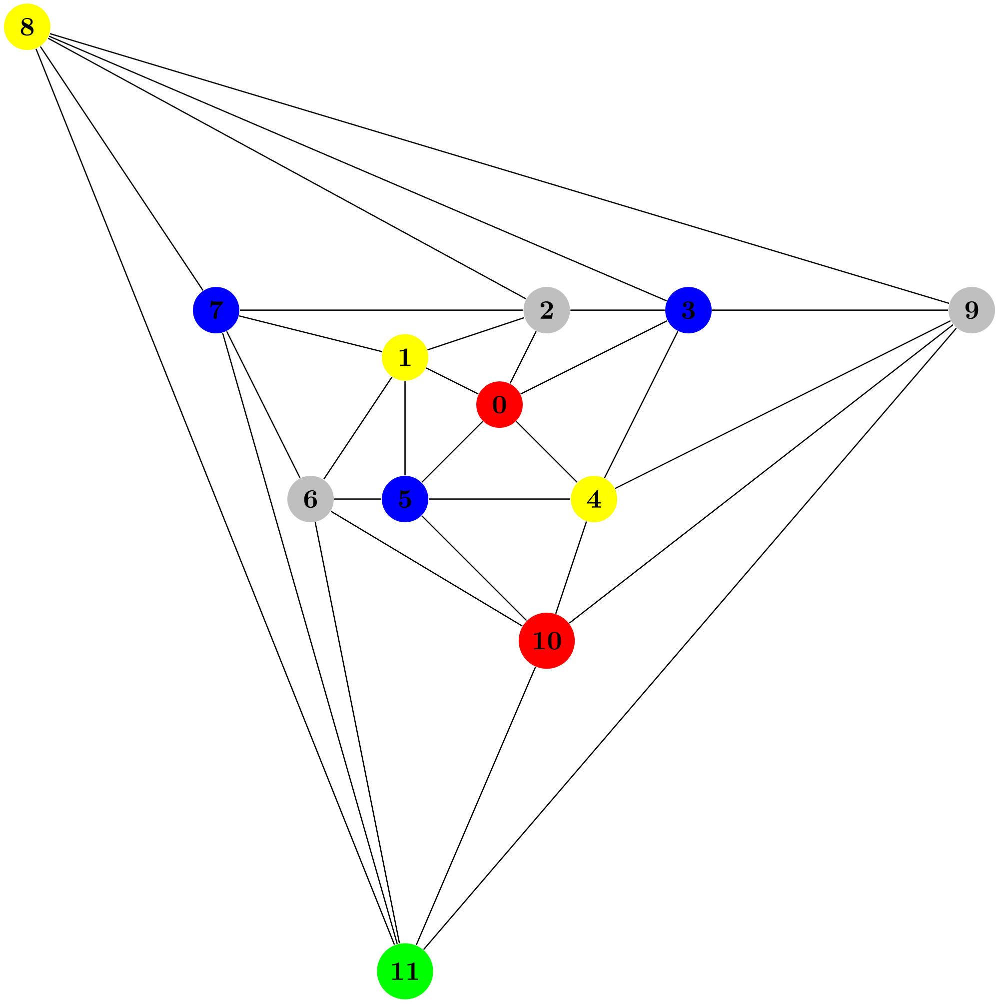


FIG. 1 – Un coloriage possible

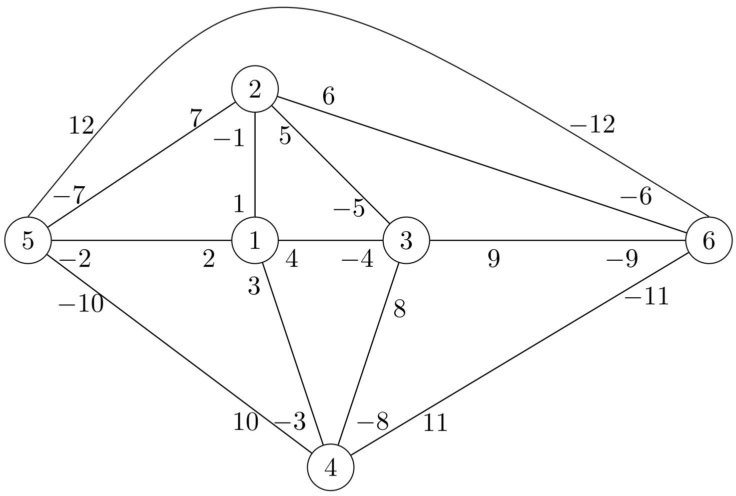


Fig. 2 – Demi-arêtes.

### 3.2 Format des données

On pourra par exemple les convertir au format de `grapheX` en préservant l'ordre des sommets.

Je me charge de fournir des fichiers tests dans le format décrit ci-dessus. Le jour de la soutenance, des fichiers répondant aux mêmes critères seront fournis.

**En sortie :** un fichier sous le format

```
n
0 c[0]
1 c[1]
...
n-1 c[n-1]
```

avec les $n$ couleurs des $n$ sommets.

Pour l'exemple donné ci-dessus, on trouve par exemple :

```
0 0
1 1
2 2
3 3
4 1
5 3
6 2
7 3
8 1
9 2
10 0
11 4
```

### 3.3 Programme, etc.

Le travail consiste à comprendre et implanter l'article en référence. On s'attachera à convaincre les examinateurs que le programme fonctionne bien en temps linéaire en $n$. Cela se fera en utilisant les graphes produits par `plantri` (prendre des cas simples ou des cas compliqués après un peu de réflexion), ainsi qu'avec des graphes aléatoires que l'on pourra fabriquer soi-même (arbres, listes).

### 3.4 Extensions possibles

Non, l'extension aux 4 couleurs ne fait pas partie de la liste !

On pourra éventuellement s'intéresser à créer des familles de graphes dont tous les sommets sont de degré $\geq 5$.

## Références

[1] N. Chiba, I. Nishizeki, and N. Saito. A linear 5-coloring algorithm of planar graphs. *J. Algorithms*, 2 :317–327, 1981.