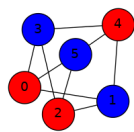


# Algorithmique des Graphes

## L3 informatique

11 décembre 2023

Vous êtes invités à remettre une source unique propre, concise, sans commentaire et sans ligne inutile...



Un graphe 3-régulier est un graphe dont tous les sommets sont de degré 3. L'ordre d'un tel graphe est obligatoirement pair et supérieur à 3. Si  $n > 3$  le graphe d'ordre  $n$  en prenant pour voisins de  $s$  les trois sommets :  $s+1$ ,  $s-1$ , et  $s+n/2$  (calculés modulo  $n$ ) est 3-régulier.

**Q1.** Coder `int test(graphe G)` qui renvoie 0 si  $G$  n'est pas régulier, son degré sinon.

**Q2.** Coder `graphe regulier(int n)` qui retourne un graphe 3-régulier d'ordre  $n$ .

L'algorithme de coloration glouton colore les sommets d'un graphe  $n$  étapes. À chaque étape, il colorie un sommet arbitraire  $s$  qui n'est pas encore colorié, en lui attribuant une couleur non utilisée par ses voisins (disponible).

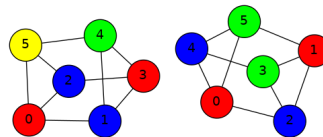
```
1 typedef struct {
2     int **mat;
3     int nbs;
4     int *clr;
5 } graphe;
```

**Q3.** Coder `int glouton( graphe g )` déterminer une coloration gloutonne du graphe  $G$  passée par le champ `clr` de la structure de graphe.

```
1 Glouton( G : graphe )
2 pour chaque sommet s de G
3   clr[ s ] = ordre(G)
4 pour chaque sommet s de G
5   X ← {}
6   pour chaque voisin t de s
7     ajouter clr[t] dans X
8   clr[s] ← disponible( X )
```

Listing 1: coloration gloutonne

**Q4.** Ecrire une commande `regulier.exe [n]` pour construire, colorier et dessiner le graphe régulier d'ordre  $n$ .



Pour quatre sommets distincts  $abcd$  d'un graphe, un croisement d'arêtes consiste à remplacer deux arêtes  $ab$  et  $cd$  par deux nouvelles arêtes  $ac$  et  $bd$ .

**Q5.** Ajouter un argument de sorte que `regulier.exe [n] [t]` pour colorier/dessiner le graphe d'ordre  $n$  obtenu après  $t$  croisements aléatoires d'arête. Observer les résultats obtenus pour formuler une conjecture sur le nombre chromatique d'un graphe 3-régulier,  $k$ -régulier?