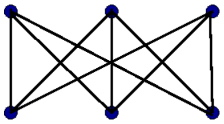
Algorithmique des Graphes
L3 informatique

4 janvier 2021

Vous êtes invités à remettre une copie claire, concise, sans rature ni surcharge en répondant aux questions dans l'ordre de l'énoncé... La note tiendra compte de la présentation générale de la copie.



Q1. On considère le graphe $K_{3,3}$ ci-dessus.

1. Combien de 2-colorations ?
2. Combien de 3-colorations ?

Q2. On considère une variable `int v[5]`. Quel est l'affichage produit par l'exécution de `nuplet( 0, 3, v )` ?

```c
void nuplet( int p, int n, int v[])
{ int i;
  if ( p == n ) {
      for( i = 0; i < n; i++)
          printf(".%d", v[ i ] );
      putchar('\n');
      return;
  }
  for( i = 0; i < n; i++){
      v[ p ] = i;
      nuplet( p + 1, n, v );
  }
}
```

Listing 1: nuplet

Q3. Pour un entier $n > 0$, on appelle tableau de permutation de l'ensemble $\{0, 1, \dots, n-1\}$, un tableau de taille n à valeurs dans $\{0, 1, \dots, n-1\}$ dont les éléments sont tous distincts.

1. Quels sont les tableaux de permutations pour $n = 3$?
2. Quel est le nombre de tableaux de permutations en fonction de n ?

```c
1  void G(int p, int pi[],  int tr[],  int n)
2  { int  j;
3     if ( p == n ) {
4     traitement( pi, n );
5     return;
6     }
7     for ( j = 0; j < n; j++ )
8     if ( tr[j] == 0 ) {
9
10          // bloc incomplet
11
12       }
13 }
14
15 void permutation(int n)
16 {
17     tr = calloc(n, sizeof(int));
18     pi = calloc(n, sizeof(int));
19     genere(0, pi, tr, n);
20     free ( tr );
21     free ( pi );
22 }
```

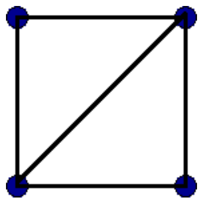
Listing 2: permutation

Q4. Comment compléter le bloc des lignes 9-

11 de la fonction G pour pour obtenir un code permettant le traitement de tous les tableaux de permutations.

Q5. Soit c_n le nombre de 3-colorations d'un graphe bicolore d'ordre n. Quelles sont les affirmations correctes :

$$c_n = \Omega(\sqrt{2^n}), \quad c_n = \Theta(\sqrt{2^n}), \quad c_n = \mathrm{O}(\sqrt{2^n}).$$

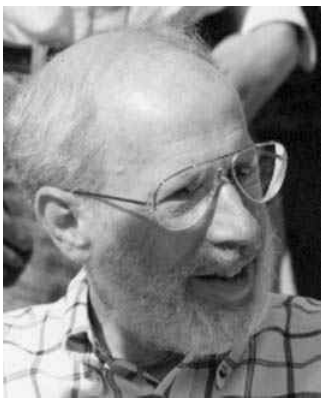


Q6. Le graphe d'adjacence d'un graphe $\Gamma(S, A)$ est le graphe $\Gamma^*(A, B)$ pour lequel chaque arête de Γ est un sommet de Γ^*. L'ensemble B des arêtes de ce nouveau graphe correspondant aux paires d'arêtes adjacentes dans Γ.

1. Dessiner le graphe d'adjacence du graphe ci-dessus.

2. Montrer que si Γ est eulérien alors Γ^* est hamiltonien.

3. La réciproque est fausse. Donner un contre-exemple.



Q7. Quel algorithme du cours est attribué au mathématicien Joseph Kruskal ? Préciser l'objectif de l'algorithme ainsi qu'une application.

Q8. Est-il possible de construire un graphe d'ordre 7 dont tous les sommets sont de degré 3?

```c
int test ( graphe g )
{
  int i;
  liste  aux;
  disjoint  r, s, *t;
  t = calloc( g.nbs,sizeof( disjoint ));
  for( i = 0; i < g.nbs; i++ )
    t[i] = singleton( i );
  for( i = 0; i < g.nbs; i++ ){
    aux = g.adj[i];
    while ( aux ) {
      if ( aux->num > i ) {
        r = representant(i);
        s = representant( aux->num );
        if ( r == s ) return 0;
        reunion( r, s );
      }
      aux = aux -> svt;
    }
  }
  for( i = 0; i < g.nbs; i++ )
      free ( t[i] );
  free ( t );
  return 1;
}
```

Listing 3: permutation

Q9. Observez le code de la fonction `test`. Proposer une définition de structure pour chacun des types : `liste`, `graphe` et `disjoint`.

Q10. Observez le code `test.c`.

1. Le code contient une grosse maladresse. Laquelle ?

2. Préciser la valeur retournée par `test`.

3. Décrire avec précision le temps de calcul.