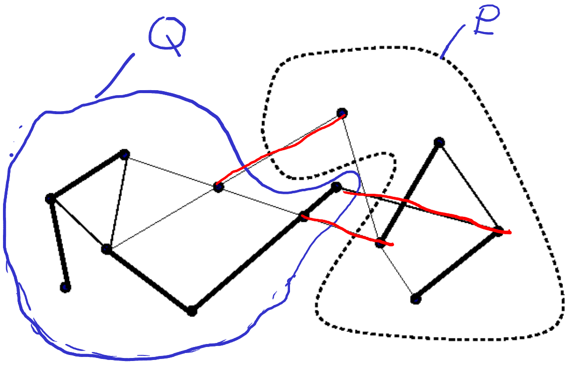
Algorithmique des Graphes
L3 informatique

mercredi 2 avril de l'an $1^3 + 2^3 + 3^3 + 4^3 + 5^3 + 6^3 + 7^3 + 8^3 + 9^3$

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. *Ecrire une source au format* **dot** *pour définir le graphe maison à 5 sommets : A, B, C, D, E.*



Q2. *Compléter la figure ci-dessus pour mettre en évidence une coupure $\{P, Q\}$ qui respecte les arêtes en trait gras, en coloriant les arêtes traversantes.*

$m \log m + 2(m-1) \log m + m-1$

tri des arêtes — calcul des représentants — union

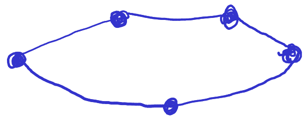
Q3. *Quelle est la complexité de l'algorithme de Kruskal appliqué à un <u>arbre</u> d'ordre n?*

$O(n \log n)$

connexe

$(n-1)$ arêtes

Q4. *Dessiner un graphe d'ordre 5 sans clique d'ordre 3, ni stable d'ordre 3.*

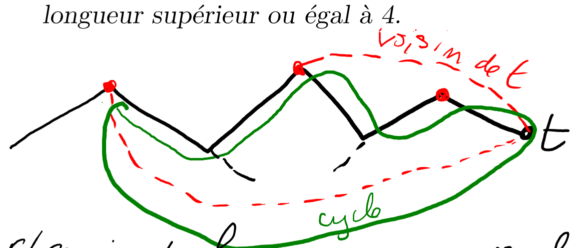


Q5. *Un automorphisme du graphe $G(S, A)$ est une permutation f de S qui conserve l'adjacence :*

$$\forall i, j \in S, \quad ij \in A \Longrightarrow f(i)f(j) \in A.$$

Quels graphes d'ordre n possèdent $n!$ automorphismes ? Le graphe K_n ou bien son complémentaire car tous les sommets sont adjacents ou isolés...

Q6. *Montrer par un dessin commenté qu'un graphe 3-régulier possède au moins un cycle de longueur supérieur ou égal à 4.*



chemin de longueur maximale terminant par un sommet t

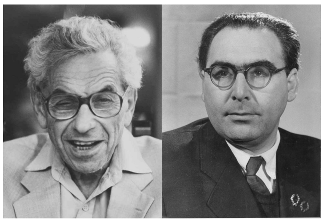
```
typedef struct {
    int nbs;
    char ** mat;
} graphe;

void PPR( int s, graphe g, int *t, int
    *r, int v[ ] );

int X( graphe g )
{ int n = g.nbs;
  int* v = calloc( n, sizeof(int) );
  for( s = 0; s < n ; s++ )
    if ( 0 == v[ s ] ) {
        int r = 0, t = 0;
        PPR( s, g, &t, &r, v );
        r = r/2 ;
        if ( r >= t ) cpt++;
    }
   free ( v );
   return cpt == 0;
}
```

Q7. *Coder une variante du parcours en profondeur récursif qui détermine le nombre de sommets et le nombre d'arêtes de la composante connexe de s.*

```
1  void PPR( int s, graphe g , int *t,  int
      *r,  int v []  )
2  {
3     v[ s ]  = 1;
4     *t  = *t + 1;
5     for( int i  = 0; i  < g.nbs; i ++ ){
6        if  (  g.mat[s][i]    )
7        *r  = *r +1;
8        if  (  v[i]  == 0 )
9        PPR( i, g, t, r,  v );
10    }
11 }
```



Paul Erdős (1913-1996)　　Alfréd Rényi (1921-1970)

Les arêtes sont comptées deux fois ! (voir ligne 1?

beaucoup trop d'erreurs concernant les transmissions par adresse...

Q8. Préciser le résultat retourner par X.

détecte la présence d'un cycle

Q9. Quelles expériences réalisées en travaux-pratiques illustrent un théorème de Erdos et Renyi ?

Si le nombre d'arêtes d'une composante connexe est supérieur ou égal à sa taille alors elle possède un cycle..

En travaux pratiques, nous avons mis en évidence la valeur seuil prédite par Erdos-Renyi, à partir de laquelle une composante géante de taille $m^{2/3}$ apparait dans le graphe aléatoire $G(p,m)$.