

## PREMIER GROUPE

**A RENDRE SUR MOODLE :** Vos programmes. Puis un 'makefile' ou un 'readme' de comment compiler et/ou exécuter vos programmes. Un petit 'rapport' de maximum 5 pages.

**Travail individuel.**

Date limite : jeudi 27 février à 23h59. Compresser le tout et mettre dans un fichier Nom.tar ou Nom.zip ou Nom.tar.gz

**Exercice 1 : Allons à Louvain**

Coder pour générer le graphe suivant ( $n, p, q$  sont des paramètres qui seront amener à fluctuer) :

- Le graphe a  $n$  sommets réparti au uniformément aléatoirement au hasard (uah) dans 4 lots de sommets :  $n_1, n_2, n_3, n_4$  (tels que  $\sum_{i=1}^4 n_i = n$ ).
  - Dans chaque lot de sommets, les sommets entre eux sont connectés avec une probabilité  $p$ .
  - Pour deux sommets dans deux différents lots, la probabilité de connexion est  $q$ .
  - Dessiner le graphe obtenu.
1. En utilisant l'algorithme de Louvain et selon le rapport  $\frac{p}{q}$ , quelles sont les structures communautaires qui émergent ? Il est fortement recommandé de faire plusieurs jeux de tests et d'en commenter quelques uns (images à l'appui).

**Exercice 2 : 'Benchmarks'**

Il y a plusieurs algorithmes possibles ('Louvain' est une possibilité parmi d'autres).

1. Faites une recherche d'au moins 3 algorithmes possibles ET lister les explicitement.
2. Faire un tableau comparatif (temps d'exécution, passage à l'échelle, ...) des algorithmes, selon les valeurs de  $n, p, q$  de l'exercice précédent. Commentez.