

Algorithmique et structures de données

Projet

Ensimag - 1A

Année 2021-2022

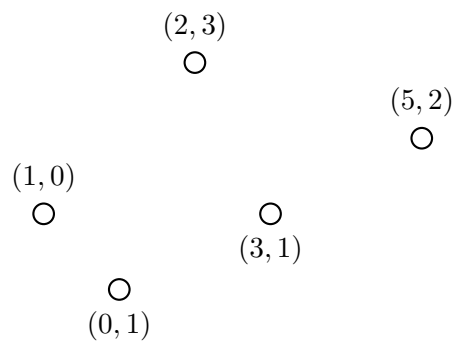


FIGURE 1 – Nuage de points dans le plan.

1 Problématique

On considère un ensemble S de n points dans le plan. On cherche un couple de points distincts de S de distance minimale. La sortie est donc un tel couple de points ; s'il y en a plusieurs il suffit d'afficher l'un d'entre eux.

Dans l'exemple de la Figure 1 la distance minimale est égale à $\sqrt{2}$ et cette distance est atteinte uniquement par les deux points (1,0) et (0,1). La sortie attendue est donc :

0, 1; 1, 0

ou

1, 0; 0, 1

2 Fichiers Fournis

On vous fournit un mini-module de géométrie "*geo*". Le module *geo* fournit une classe *Point*. La bibliothèque contient également une fonction *tycat* qui permet d'afficher des itérables de points et segments dans *terminology*. Un petit exemple de son utilisation est donnée dans le fichier *hello.py*.

Le répertoire de travail contient également un ensemble de fichiers de tests "exemple_*.pts" qui contiennent chacun un ensemble de points.

On vous demande de modifier le fichier *main.py* en évitant de changer la fonction *main*. La fonction *load_instance* permet de charger un fichier de test. Le *main* lance donc les calculs sur tous les fichiers donnés en argument. Vous pouvez bien sûr créer d'autres fichiers python ou fonctions mais les tests automatiques seront réalisés en lançant *main.py* sur mes jeux de tests.

3 Résultats Attendus

Pour réaliser ce projet on vous demande d'écrire au moins un algorithme répondant au problème donné. De plus, il vous est demandé de réaliser un rapport de quelques pages incluant au évaluation expérimentale de votre/vos algorithme(s). En particulier on vous demande d'expliquer la problématique, les approches vous avez considérées et de tracer au moins une courbe de performance. Nous vous demandons d'explicitier sous quelles conditions sur les entrées votre/vos algorithme(s) fonctionne(nt).

Le code et le rapport sont à uploader sur TEIDE pour le 24 avril 2022.

Attention ! Vous pouvez échanger entre vous sur les idées et méthodes de travail mais vous ne devez pas échanger de code. Chacun est en charge de la sécurisation de ses fichiers.

4 Gitlab

Je vais créer un répertoire par équipe sur gitlab. Afin de pimenter un peu le projet je propose de réaliser un high-score.

Je dispose pour cela de trois jeux de tests.

- un jeu facile, que tout le monde devrait réussir à passer ;
- un jeu plus difficile ;
- un jeu très difficile.

Il y aura donc trois high-score, un par jeu de test.

Chaque nuit je vais lancer sur mon pc de bureau un script qui va pour chaque groupe :

- récupérer la dernière version du code sur gitlab avec un git pull ;
- lancer le test 1 (avec timeout) ;
- si le test 1 passe, lancer le test 2 (timeout plus élevé) ;
- si le test 2 passe, lancer le test 3 (timeout encore plus élevé).

Il n'y aura pas d'accès au réseau, pas d'accès au disque et pas d'accès à des modules python.

Je garde un historique des tous les temps obtenus et génère les high score utilisant pour chaque équipe le meilleur temps obtenu à chaque test (potentiellement des algos différents).

Vous pourrez retrouver sur chamilo un lien vers les pages de high-score.

Quelques informations supplémentaires :

- Le high score est là pour rigoler un peu mais ne sera pas pris en compte dans l'évaluation du projet qui évalue les algorithmes développés et vos expériences.
- C'est compliqué d'écrire des tests. Surtout pour faire passer tous les projets en une nuit. Je n'ai donc aucune idée que cela peut donner. Trop facile ? Trop difficile ?
- Si tout ne se passe pas comme prévu je rajouterai de nouveaux jeux de tests.