



# Projet individuel: rédaction d'un rapport de recherche sur l'algorithme Welzl résolvant le problème du cercle minimum.

BM Bui-Xuan

**Problème du cercle minimum :** Le problème du cercle minimum consiste à déterminer, à partir d'un ensemble de points dans le plan, un cercle d'aire minimum contenant tout point de l'ensemble.

## 1 L'énoncé du projet

Il s'agit d'analyser la performance en temps de calcul de l'algorithme Welzl calculant un cercle couvrant un ensemble de points dans le plan. L'évaluation se fera en fonction de la lisibilité du code, et surtout (à lire : 90%) de la rédaction du rapport.

ATTENTION : le travail de groupe pour le code et l'implémentation est autorisé. La rédaction du rapport doit être individuelle, ainsi que son rendu.

### Exercice 1

1. Implanter un algorithme naïf calculant exactement le cercle minimum couvrant un ensemble donné de points dans le plan (voir p.e. le TME1).
2. Implanter l'algorithme Welzl décrit à la fin de la page 362 de l'article original [http://www.stsci.edu/~RAB/Backup%20Oct%2022%202011/f\\_3\\_CalculationForWFIRSTML/Bob1.pdf](http://www.stsci.edu/~RAB/Backup%20Oct%2022%202011/f_3_CalculationForWFIRSTML/Bob1.pdf)
3. Récupérer une base de test de taille conséquente : le nombre minimum d'instances de test dans la base de test doit être 1664 ; le nombre minimum de points dans une instance doit être 256. Par exemple, une base de test vérifiant ces contraintes est celle de VAROUMAS disponible en ligne à l'adresse suivante : [http://www-apr.lip6.fr/~buixuan/files/ado23cpa/Varoumas\\_benchmark.zip](http://www-apr.lip6.fr/~buixuan/files/ado23cpa/Varoumas_benchmark.zip)
4. Confronter l'algorithme Welzl aux instances de la base de test. Pour chaque instance on sera notamment intéressé par la correction de l'algorithme Welzl, comparée à l'algorithme naïf (algorithme de contrôle) que l'on suppose correctement implanté. On sera également intéressé par le temps de calcul des deux algorithmes.
5. Etablir au moins un diagramme-bâtons (moyenne + déviation standard) représentant le gain en temps de votre implantation de l'algorithme Welzl par rapport à l'implantation de l'algorithme naïf.
6. Rédiger un rapport de cette expérience (introduction, résultats, discussion, conclusion) : on peut utiliser comme modèle le rapport de recherche suivant, qui traite une question dans le même domaine de recherche : <http://hal.inria.fr/inria-00072354/PDF/RR-4233.pdf>

**Question bonus :** Quelle est votre impression par rapport aux notions similaires en 3D ?

## Remarques sur la rédaction du rapport

Un effort particulier doit être mis sur la phase de test et celle de la rédaction du rapport. On veillera à expliciter les points suivants, pour chaque algorithme implémenté, e.g. Welzl :

- définition du problème et la structure de données utilisée.
- analyse et présentation théorique des algorithmes connus dans la littérature.

- argumentation concise appuyant toute appréciation, amélioration, ou critique à propos de ces algorithmes existants dans la littératures.
- partie test : méthode d’obtention des *testbeds*. En particulier, il est important de citer la provenance des fichiers sources, e.g. celle de VAROUMAS si l’on avait décidé de l’utiliser.
- test de performance : mieux vaut privilégier les courbes, diagrammes bâton (moyenne + déviation standard) et diagrammes de fréquence, plutôt qu’exhiber des colonnes de chiffre sans fins...
- une discussion sur les résultats de test de performance est toujours la très bienvenue.
- conclusion et perspectives sur le problème de cercle minimum.

Il est prudent d’avoir entre 8 et 15 pages pour un rapport avec un contenu moyen. Le nombre de pages recommandé pour ce rapport est 12 pages. Il convient de bien respecter cette limitation : les pages 15+ ne seront pas lues !

#### Contraintes :

- Le rapport est à écrire de façon individuelle. Le plagiarisme est strictement interdit.
- Archiver la totalité du rendu en un seul fichier compressé contenant de la documentation (rapport  $\approx$  8-15 pages), un binaire (ou les .class si Java) et un README expliquant comment exécuter le binaire, le code source commenté, un Makefile (ou Apache Ant si Java), un répertoire contenant une certaine partie de la base de test (attention à la limitation en espace ci-après), et tout ce dont on juge utile à la lecture du projet sans toute fois dépasser la limitation en espace fixée à une dizaine de Méga-octet.
- La nomination de préférence est cpa-projet-2023-NOM.piki, où piki peut être un élément de  $\{tgz, zip, rar, 7z, etc\}$ .
- Deadline : 22 Décembre 2023, 17h00. Pénalité de retard : malus de  $2^{h/24}$  points pour  $h$  heures de retard.