

## Examen

### *Documents autorisés*

Pour chaque question, on séparera les déclarations dans les fichiers en-tête `point.hpp`, `disque.hpp` et `ensemble.hpp` le code correspondant dans les fichiers `point.cpp`, `disque.cpp` et `ensemble.cpp` pour le code du corps des fonctions. Le programme principal sera codé dans `exe.cpp`.

La présentation qui devra être parfaitement claire et lisible.

#### **Point**

Définir une classe `Point` avec des coordonnées réelles x et y, et le constructeur à deux paramètres.

Ecrire une fonction amie `Distance` calculant la distance entre deux points.

#### **Disque**

Définir une classe `Disque` avec un centre et un rayon, écrire le constructeur.

Ecrire une fonction membre `Intersection` retournant vrai si un disque coupe un autre, faux sinon.

#### **Ensemble**

Définir une classe `Ensemble` avec contenant un tableau de taille variable de `Disque`.

Ecrire une fonction membre `Taille` retournant la taille de l'`Ensemble`.

Ecrire une procédure membre `Ajoute` ajoutant un `Disque` à l'`ensemble` uniquement si le disque passé en paramètre ne coupe pas les disques existant dans l'`Ensemble`, et ne l'ajoute pas sinon.

Ecrire une fonction `Test` testant si un `Point` est dans un `Disque` de l'`Ensemble`, si oui la fonction retournera vrai et l'indice dans le tableau, sinon la fonction retournera faux, et l'indice sera défini égal à -1.

#### **Programme principal**

On suppose qu'il existe une fonction `Disque Aleatoire()` ; qui construit un disque de centre et de rayon aléatoire.

Ecrire le programme principal construisant un `Ensemble` rempli de 100 `Disque` générés aléatoirement.

#### **Compilation**

Ecrire le fichier `Makefile` pour compiler le programme à partir des sources.