

## ***Module « Algorithmique Avancé »***

**Tronc Commun / Année 2019-2020**

### **Liste des mini-projets**

#### **Préambule**

Il est demandé que chaque étudiant prépare soi-même un mini-projet en algorithmique avancé. Il y a cinq mini-projets affectés selon le tableau suivant :

Numéros des étudiants	Numéro de sujet affecté
Entre 1 et 18	1 (ensembles finis)
Entre 19 et 36	2 (polygones convexes)
Entre 37 et 54	3
Entre 55 et 72	4
Plus que 73	5

A la fin, chaque étudiant doit **fournir un rapport bien rédigé** contenant son travail (Description des algorithmes, Description du code, Ecrans de démonstration d'exécution). Une **présentation orale** du travail est également envisageable.

Chaque sujet doit être réalisé de la façon suivante :

- **Ecrire les algorithmes.** Pour chaque algorithme : préciser vos choix et vos contraintes, présenter l'idée de votre algorithme (sans pseudo-code), présenter son pseudo-code et évaluer sa complexité.
- **Coder les algorithmes en C.** Vous devez réaliser un module C qui contient une interface (fichier .h) et une implémentation (fichier .c). Un programme principal doit être fourni sous forme d'un menu principal pour tester tous les cas du module.

#### **Sujet N° 1**

Réaliser un module pour le traitement des ensembles finis au sens mathématique en utilisant les listes chaînées. Le module doit fournir les opérations suivantes : créer un ensemble, afficher un ensemble sous le format usuel, calculer le cardinal d'un ensemble, tester si un ensemble est vide, calculer la réunion, l'intersection, la différence et la différence symétrique de deux ensembles, calculer le complémentaire d'un ensemble A dans un ensemble E, tester l'appartenance d'un élément à un ensemble, comparer deux ensembles (égalité, inclusion large et inclusion stricte), tester si deux ensembles sont disjoints.

#### **Sujet N° 2**

Un polygone est une figure géométrique fermée composée de plusieurs segments. On se restreint aux polygones convexes. Réaliser un module pour le traitement des polygones convexes en utilisant les listes chaînées. Le module doit fournir les opérations suivantes : créer un polygone, calculer le périmètre et la surface d'un polygone, tester si un polygone est un triangle et le qualifier (isocèle, équilatéral, rectangle, isocèle rectangle), tester si un polygone est un quadrilatère et le qualifier (trapèze, parallélogramme, carré, rectangle, losange), tester si un point est à l'extérieur, à l'intérieur ou sur un côté d'un polygone, tester si deux polygones sont similaires (leurs côtés correspondants sont proportionnels et leurs angles correspondants sont égaux), tester si un polygone est régulier (ses côtés sont isométriques et ses angles sont égaux), tester si un polygone est à l'intérieur d'un autre.

### **Sujet N° 3**

Réaliser un module pour le traitement des polynômes à coefficients entiers en utilisant les listes chaînées. Le module doit fournir les opérations suivantes : créer un polynôme, afficher un polynôme sous le format usuel, calculer la somme, la différence et le produit de deux polynômes, calculer l'opposé et la puissance  $n^{\text{ème}}$  d'un polynôme, calculer la division euclidienne (quotient et reste) d'un polynôme par un autre, calculer la dérivée d'un polynôme, calculer une primitive d'un polynôme, calculer le plus grand diviseur commun de deux polynômes, calculer la valeur d'un polynôme en un entier  $x$ , tester si un entier  $x$  est un zéro d'un polynôme.

### **Sujet N° 4**

Réaliser un module pour le traitement des vecteurs à coefficients entiers en utilisant les listes chaînées. Le module doit fournir les opérations suivantes : créer un vecteur, afficher un vecteur sous le format usuel, calculer la somme, la différence de deux vecteurs, calculer l'opposé d'un vecteur, calculer le produit par un scalaire d'un vecteur, calculer le produit scalaire de deux vecteurs, calculer le produit vectoriel de deux vecteurs, calculer la norme euclidienne d'un vecteur, tester si deux vecteurs sont orthogonaux, tester si une famille de vecteurs sont liées ou libres, tester si deux vecteurs sont égaux, calculer le cosinus de l'angle entre deux vecteurs, calculer le déterminant d'une famille de vecteurs.

### **Sujet N° 5**

Réaliser un module pour le traitement des grands nombres entiers naturels en utilisant les listes chaînées. Le module doit fournir les opérations suivantes : créer un nombre entier, afficher un nombre entier sous le format usuel, calculer la somme, la différence et le produit de deux nombres entiers, calculer le quotient et le reste de la division euclidienne de deux nombres entiers, tester si un nombre entier est divisible par un autre, tester si un nombre est premier, calculer la puissance  $n^{\text{ème}}$  d'un nombre entier.

**N.B.** Dernier délai pour rendre les mini-projets : **30 décembre 2019.**