

UE ALGORITHMIQUE

Examen - Session 1

Durée : 2 h 30 – Tous documents autorisés. Appareils électroniques interdits.

Il sera tenu compte de la lisibilité des algorithmes. Pour chaque algorithme réalisé, on définira soigneusement les variables, les fonctions et les actions introduites pour résoudre le problème.

Les exercices sont indépendants et peuvent être traités dans n'importe quel ordre.

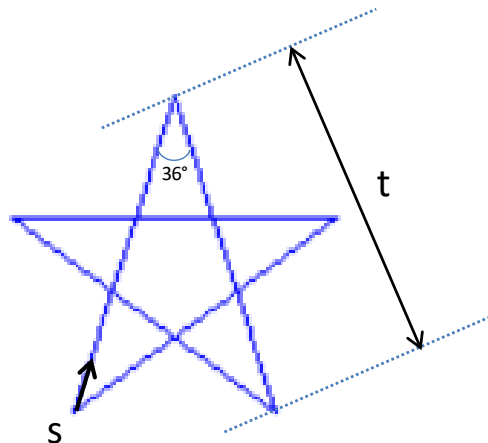
Dans un exercice, pour résoudre une question, vous pouvez utiliser une action ou une fonction définie dans une question précédente. Le barème donné est indicatif.

SUJET (4 pages)

Exercice 1 : Etoiles (5 points)

1) Tracé d'une étoile à cinq branches

A l'aide d'une machine-tracés, on veut tracer une étoile à cinq branches comme celle de la figure suivante :



Les angles formés par les 5 branches (pointes) de l'étoile sont de 36°. Nous rappelons aux étourdis que $180 - 36 = 144$.

Une telle étoile est caractérisée par :

- son sommet (s sur la figure) : c'est l'un des sommets de l'étoile,
- sa taille (t sur la figure) : c'est la longueur de tous les segments reliant deux sommets de l'étoile,
- son cap (marqué par une flèche sur la figure) : c'est le cap de l'un des segments passant par le sommet s de l'étoile, et tel qu'un observateur placé au sommet s et regardant dans le sens défini par ce cap, voit l'autre extrémité du segment devant lui et le segment suivant **à sa droite**.

Réaliser l'action de tracé d'une étoile à partir de la position courante de la plume, après avoir précisé l'état initial et final de l'action :

action **tracerUneEtoile** (consulté t : réel, modifié m : Machine-tracés)

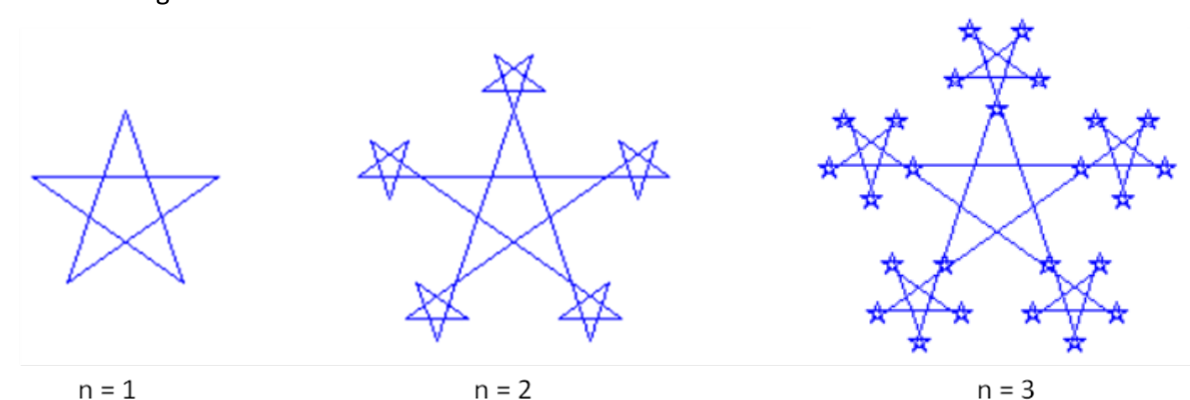
// Effet : *trace une étoile de taille t à partir de la position courante de la plume*

// E.I. : à préciser

// E.F. : à préciser

2) Une explosion d'étoiles

On veut construire un algorithme de tracé d'un dessin récursif formé d'étoiles à cinq branches, comme l'illustre la figure suivante :



Un tel dessin est caractérisé par :

- son ordre, comme le suggèrent les figures : la figure d'ordre 1 comporte une étoile, la figure d'ordre 2 comporte une étoile de taille t et cinq étoiles de taille $t/3$, placées à chacun de ses sommets,
- son sommet, sa taille, son cap : ce sont le sommet, la taille et le cap de l'étoile la plus grande.

Spécifier et donner une réalisation récursive d'une action nommée **tracerDesEtoiles** permettant de tracer le dessin de taille t et d'ordre n , sachant qu'à l'état initial, la position dans le plan et le cap de la plume sont respectivement le sommet et le cap du dessin que l'on veut tracer.

```
action tracerDesEtoiles (..... à compléter .....)  
// Effet : trace le dessin de taille  $t$  et d'ordre  $n$  à partir de la position courante de la plume  
// E.I. : à compléter  
// E.F.: à compléter
```

Exercice 2 : Mots palindromes (5 points)

On considère un fichier de caractères comportant un texte formé de lettres et d'espaces. Un mot est une suite de lettres délimitée par des espaces ou par les extrémités du texte. Le texte peut commencer par des espaces. Ecrire un algorithme qui affiche un message indiquant le nombre de mots palindromes contenus dans le texte. Rappel : un mot palindrome est un mot qui se lit de gauche à droite comme de droite à gauche. Pour résoudre le problème, vous pouvez reprendre des actions ou fonctions étudiées en cours ou en TD et les recopier ou les adapter.

Exemple : fichier lu : ELLE PARTIT TOT POUR LAVAL DANS SON KAYAK BLEU `fdf`

résultat affiché : nombre de mots palindromes : 4

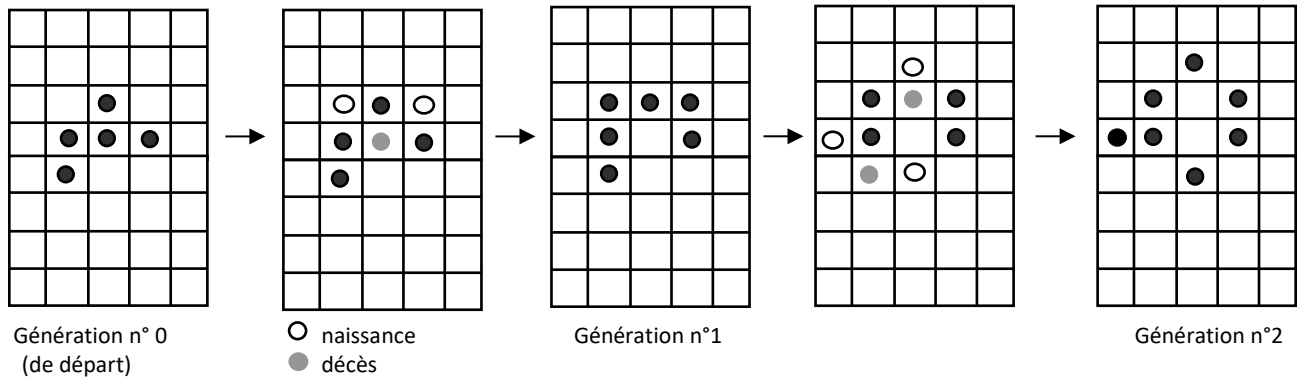
`fdf` représente la marque de fin de fichier

Exercice 3 : Jeu de la vie (10 points)

Le jeu de la vie, proposé par John Horton Conway dans la fin des années soixante, consiste à faire évoluer une population de points positionnés dans une grille, de génération en génération, en appliquant les règles d'évolution suivantes :

- tout point au contact de 0 ou un point, meurt de solitude ;
- tout point au contact de plus de 3 points, meurt étouffé ;
- tout point en contact de 2 ou 3 points survit dans la prochaine génération ;
- un nouveau point naît dans toute case vide, au contact de 3 points exactement.

Exemple de générations successives obtenues en appliquant ces règles :



Représentation informatique des informations

On considère que le monde dans lequel évolue la population de points est une grille de NMAX lignes et de NMAX colonnes. On représente donc la population de points par un tableau de booléens à deux dimensions.

Afin de faciliter l'écriture de l'algorithme de calcul d'une nouvelle génération, 2 lignes et 2 colonnes bordent le monde dans lequel évolue la population : lignes 0 et NMAX+1, colonnes 0 et NMAX+1 :

Population : type tableau sur [0..NMAX+1, 0..NMAX+1] de booléens

Si **p** est une Population, $p[i][j]$ a la valeur vrai si la case de coordonnées (i, j) contient un point, et la valeur faux sinon.

La figure ci-dessous décrit la représentation de la population de départ de l'exemple précédent :

		0							NMAX+1
0	F	F	F	F	F	F	F	F	F
	F	F	F	F	F	F	F	F	F
	F	F	F	F	F	F	F	F	F
	F	F	F	F	V	F	F	F	F
	F	F	F	V	V	V	F	F	F
	F	F	F	V	F	F	F	F	F
	F	F	F	F	F	F	F	F	F
	F	F	F	F	F	F	F	F	F
NMAX+1	F	F	F	F	F	F	F	F	F

Question 1

Ecrire l'action **lirePopulation** qui initialise une population à partir de la liste des coordonnées (indice ligne et indice colonne) des points formant cette population, représentée dans un fichier d'entiers. Pour la population de la génération n°0 de l'exemple, on a le fichier suivant :

3 4 4 3 4 4 4 5 5 3 fdf

lirePopulation : action (Consulté f : fichier d'entiers, Elaboré p : Population)

// Effet : construit la population p à partir du fichier f

Question 2

Ecrire l'action **enregistrerPopulation** qui sauvegarde une population donnée dans un fichier d'entiers :

enregistrerPopulation : action (Consulté p : Population, Elaboré f : Fichier d'entiers)

// Effet : enregistre les coordonnées des points de la population p dans le fichier f (2 entiers par point)

Question 3

Ecrire l'action **nouvelleGénération** qui calcule la génération de points obtenue à partir de la population donnée en paramètre, en appliquant les règles d'évolution énoncées :

nouvelleGénération : action (Consulté p1 : Population, Elaboré p2 : Population)

// Effet : calcule la population p2 à partir de la population p1

Question 4

Ecrire un algorithme principal qui :

- lit une population initiale dans un fichiers d'entiers
- lit le nombre MaxGen de générations à calculer,
- calcule Maxgen générations à partir de la population initiale si cela est possible. On arrête les calculs avant la génération Maxgen si la population a disparu (tous les points sont morts).
- enregistre la génération n°Maxgen dans un fichier.

Si la population a disparu, on ne crée pas de fichier mais on affiche un message spécial indiquant à quelle génération la population a disparu.