

## LIF1 – Interrogation 3, correction

### Exercice 1

Fonction `sous_chaine`( $c_1, c_2$  : chaînes de 50 caractères) : entier

Données :  $c_1, c_2$

Résultat : nombre d'occurrences de  $c_1$  dans  $c_2$

Variables locales  $a, i, j$  : entiers

Début

$i \leftarrow 0$

$a \leftarrow 0$

Tant que  $c_2[i] \neq '\0'$  faire

$j \leftarrow 0$

Tant que  $c_1[j] \neq '\0'$  et  $c_2[i+j] \neq '\0'$  et  $c_1[j] = c_2[i+j]$  faire

$j \leftarrow j + 1$

FinTantque

Si  $c_1[j] = '\0'$  alors

$a \leftarrow a + 1$

FinSi

$i \leftarrow i + 1$

FinTantque

Renvoyer  $a$

Fin

*Remarque* : si  $n$  et  $m$  sont les tailles respectives de  $c_1$  et  $c_2$ , alors cet algorithme fait environ  $nm$  étapes. Il existe des algorithmes plus efficaces (mais plus compliqués) pour rechercher un sous-chaîne : par exemple, l'algorithme de Knuth-Morris-Pratt fonctionne en temps  $O(n + m)$ .

### Exercice 2

1. Structure point

$x$  : réel

$y$  : réel

Fin structure point

$f$  : tableau de 50 structure point

Procédure saisie( $T[50]$  : tableau de 50 structure point)

Donnée/résultat :  $T[50]$

Variable locale  $i$  : entier

Début

$i \leftarrow 0$

$T[0].x \leftarrow -1$

Tant que  $T[0].x < 0$  ou  $T[0].y < 0$  faire

Afficher(Entrez votre premier point)

Saisir  $T[0].x$

Saisir  $T[0].y$

FinTantque

Pour  $i$  allant de 1 à 49 par pas de 1 faire

$T[i].x \leftarrow T[i-1].x - 1$

Tant que  $T[i].x \leq T[i-1].x$  ou  $T[i].y < 0$  faire

Afficher(Entrez le point numéro  $i$ )

Saisir  $T[i].x$

Saisir  $T[i].y$

FinTantque

FinPour

Fin

2. Fonction aire( $T[50]$  : tableau de 50 structure point) : réel

Donnée :  $T[50]$

Résultat : aire sous la courbe

Variables locales  $i$  : entier,  $a$  : réel

Début

$a \leftarrow 0$

Pour  $i$  allant de 0 à 48 par pas de 1 faire

$a \leftarrow a + (T[i+1].x - T[i].x) \times (T[i].y + T[i+1].y)/2$

FinPour

Renvoyer  $a$

Fin

*Remarque* : cette méthode, appelée *méthode des trapèzes* (pourquoi?), est utilisée pour le calcul approché d'intégrales.