I Occurrence d'une lettre dans un message

1. `Travail par écrit` :
   on considère l'algorithme suivant

```
Algorithme A1
Début ...
1|   Saisir message
2|   n ← longueur(message)
3|   i ← 0
4|   cpt ← 0
5|   Tant que i<n
6|       Si message[i]="a" alors
7|           cpt ← cpt+1
8|       Fin Si
9|   ..............
10|  Fin Tant que
11|  Afficher cpt
Fin
```

   (a) Lister les variables utilisées ainsi que leur type.

   (b) Il manque une instruction à la ligne 9. Laquelle ?

   (c) Quel est le rôle de cet algorithme ?

   (d) Faire fonctionner cet algorithme avec :
   - message = « J'aime les ananas ! »
   - message = « Avez vous appris votre cours ? »

   (e) Quelle modification pourrait-on faire à la ligne 6 afin de prendre en compte le fait que la lettre "a" peut être écrite en majuscule ?

   (f) Proposer une modification $A2$ de l'algorithme $A1$ pour qu'il affiche sous forme de tableau, le nombre d'occurrence de toutes les lettres de l'alphabet.
   Par exemple, avec le message « Bonjour, comment allez-vous ? », l'algorithme affichera :

   [1,1,1,0,2,0,0,0,0,1,0,2,2,2,4,0,0,1,1,1,2,1,0,0,0,1]

   Remarque 1 On évitera les messages avec accents

2. `Travail sur poste informatique` :

   (a) Implémenter et tester l'algorithme $A1$ en Python.

   (b) Implémenter et tester l'algorithme $A2$ en Python.

   (c) Modifier ce dernier programme pour qu'il affiche la lettre la plus fréquente du message. Si plusieurs lettres ont le même nombre d'occurrences, on fera apparaître la première dans l'ordre alphabétique.

II Codage affine

1. `Travail par écrit` : Codage César.
   on considère l'algorithme suivant

```
Algorithme B1
Début
1|   alphabet ← "abcdefghijklmnopqrstuvwxyz"
2|   Saisir mot
3|   nouveaumot ← ""
4|   Pour i allant de 1 à longueur(mot)
5|       indice ← position dans l'alphabet de mot[i]
6|       nouvelindice ← (indice + 3)%26
7|       nouveaumot ← nouveaumot + alphabet[nouvelindice]
8|   Fin Pour
9|   Afficher nouveaumot
Fin
```

on pourra dans cette partie utiliser la correspondance suivante : $a \to 0$, $b \to 1$, $c \to 2$, $\ldots$, $z \to 25$.

(a) Lister les variables utilisées ainsi que leur type.

(b) Quel est le rôle de cet algorithme ?

(c) Faire fonctionner cet algorithme avec *mot*=**"bonjour"**.

(d) Proposer un modification de l'algorithme de l'algorithme $B1$ pourqu'il puisse gérer des messages avec espaces ou ponctuation (*on ne gère pas dans ce travail les messages avec majuscules et accents*).

(e) Proposer une modification de l'algorithme $B1$ pour qu'il décode le message suivant, crypté avec le codage César : **« dx uhyrlu »** .

2. `Travail sur poste informatique : Décalage inconnu`

(a) Implémenter et tester l'algorithme $B1$.

(b) On considère une nouvelle méthode pour crypter un message. On décale toujours les lettres mais pas forcément de 3 rangs. On considère le message suivant, codé avec un décalage inconnu :

« ath bpiwtbpixfjth, r'thi qxtc ! »

i. En utilisant la partie A, analyser les fréquences d'apparition de chacune des lettres du message.

ii. Modifier le programme précédent pour qu'il repère la lettre la plus fréquente, en déduire le décalage qui a pu être appliqué pour coder le message, et décoder le message.

(c) il arrive que le message ne comporte pas suffisamment de lettres pour observer une différence notable de fréquences d'apparition et donc ne permette pas à coup sû de décoder le message de manière correcte.

`Par exemple` : **« ru jkjwmxwwj »**

Comment modifier le programme précédent pour qu'il teste plusieurs décodages possibles ? On pourra considérer que si la lettre la plus fréquente du message codé ne correspond pas à la lettre « e », alors peut-être qu'elle correspond à « s », ou « a » ...