

Exercice I : Demandez un nombre compris entre 1 et 3. Pour cela, on pose la question :

Donnez votre choix (1, 2, 3)

Tant que la réponse ne convient pas.

Variable Nombre : numérique

Répéter

Ecrire « Donnez votre choix (1, 2 ou 3) »

Lire Nombre

Tant que Nombre < 1 ou Nombre > 3

Ecrire « OK »

Exemple d'exécution :

Donnez votre choix (1, 2 ou 3)

0

Donnez votre choix (1, 2 ou 3)

2

OK

Exercice II : Ecrire les instructions permettant de répéter la question « voulez-vous continuer » Tant que la réponse est différent de la lettre O (pour Oui) et la lettre N (pour Non).

Variable Reponse : caractère

Répéter

Ecrire « Voulez-vous continuer »

Lire Reponse

Tant que Reponse <> « O » et Reponse <> « N »

Rmq : Là encore, il ne s'agit pas d'un programme complet. De telles instructions ne peuvent présenter un intérêt quelconque que si elles apparaissent au sein d'un programme plus important.

Rmq : La variable « Reponse », de type caractère, ne peut contenir qu'un seul caractère. Or ici, il serait pratique de pouvoir considérer des réponses telles que OUI ou NON, formées de plusieurs caractères. C'est possible dans certains langages qui possèdent le type « chaîne de caractères » ou dans lesquels les variables de type « caractère » contiennent plusieurs caractères. Dans les autres cas, il faut utiliser des *tableaux de caractères*.

Exercice III : Recherche de la première consonne

- 1- Rechercher et écrire la première consonne d'un mot. Nous supposons que ce mot a été écrit, en minuscules, sur une seule ligne et qu'il est possible d'en lire les caractères les uns après les autres.
- 2- Modifier le programme pour prévoir le cas où le mot ne comporte aucune consonne. On supposera que le mot se termine par au moins un caractère espace et qu'il est écrit en minuscules.

Rmq : Pour savoir si une lettre est une consonne, il vaut mieux éviter de la comparer avec toutes les consonnes de l'alphabet. Il est plus simple de s'assurer qu'elle n'est égale à aucune des six voyelles.

Solution question 1-

Variable c : caractère

Répéter

Lire c

Tant que c = « a » ou c = « e » ou c = « i » ou c = « o » ou c = « u » ou c = « y »

Ecrire « première consonne », c

Solution question 2-

Nous prenons comme hypothèse que le mot se termine par un espace que nous noterons « Λ »

```

Variable c : caractère
Répéter
    Lire c
Tant que c =« a » ou c= « e » ou c= « i » ou c= « o » ou c= « u » ou c= « y » ou
c≠ « Λ »
Si c≠ « Λ » alors Ecrire « première consonne », c
    Sinon Ecrire « Pas de consonnes »
Fsi

```

**Rmq :** attention aux conditions de terminaison des boucles. Ici on considère qu'il y a obligatoirement un espace en fin de mot. Dans le cas de la solution de la question 1, s'il n'y a pas de consonne dans le mot, le programme ne se termine jamais.

**Exercice IV :** Rechercher et écrire toutes les consonnes d'un mot supposé porté sur une seule ligne. Les caractères sont lus les uns après les autres.

```

Variable c : caractère
Répéter
    Lire c
    Si c ≠ « a » ou c≠ « e » ou c≠ « i » ou c≠ « o » ou c≠ « u » ou c≠ « y »
ou c≠ « Λ »
    alors Ecrire c
    Fsi
Tant que c <> « Λ »

```

**Exercice V :** Soit le programme suivant :

```

Variable Nombre : numérique
Ecrire « Donnez un nombre inférieur à 100 »
Répéter
    Lire Nombre
Tant que nombre >= 100
Ecrire « OK »

```

**Améliorer ce programme pour que l'exécution se présente ainsi**

```

Donnez un nombre inférieur à 100
248
svp inférieur à 100
103
svp inférieur à 100
83
Ok

```

**Rmq :** L'enchaînement des lignes suivantes ne permet pas d'obtenir le résultat.

```

Ecrire « Donnez un nombre inférieur à 100 »
Lire Nombre
Ecrire « SVP inférieur à 100 »

```

**Car même si le nombre est inférieur à 100, on affichera la deuxième phrase.**

```

Variable Nombre : numérique
Ecrire « Donnez un nombre inférieur à 100 »
Répéter
    Lire Nombre
    Si Nombre ≥ 100 alors Ecrire « SVP inférieur à 100 » Fsi
Tant que nombre >= 100
Ecrire « OK »

```

**Exercice VI :** Adapter de manière analogue le programme suivant :

```

Variable Reponse : caractère
Répéter

```

```

    Ecrire « Voulez-vous continuer »
    Lire Reponse
Tant que Reponse <> « O » et Reponse <> « N »
Ecrire « OK »

```

```

Voulez-vous continuer
Y
O/N svp
l
O/N svp
O
OK

```

```

Variable Reponse : caractère
Ecrire « Voulez-vous continuer »
Répéter
    Lire Reponse
    Si Reponse ≠ « O » ou Reponse ≠ « N »
        Alors Ecrire « O/N svp »
    Fsi
Jusqu'à Reponse = « O » ou Reponse = « N »
Ecrire « OK »

```

**Exercice VII : Adapter le programme précédent pour que, lorsque la réponse est correcte, il écrive, au lieu de OK :**

- vous avez dit OUI
- ou : vous avez dit NON

suivant le cas.

```

Variable Reponse : caractère
Ecrire « Voulez-vous continuer »
Répéter
    Lire Reponse
    Si Reponse ≠ « O » ou Reponse ≠ « N »
        Alors Ecrire « O/N svp »
    Fsi
Tant que Reponse <> « O » et Reponse <> « N »
Ecrire « OK »
Si Reponse = « O » alors Ecrire « Vous avez dit OUI »
    Sinon Ecrire « Vous avez dit NON »
Fsi

```

Rmq : il est possible d'utiliser des si alors sinon imbriqués pour tester dans quel cas on se trouve. Les si imbriqués peuvent être formulés par une suite de si.

**Exercice VIII : Compter le nombre de caractères d'une phrase qui a été portée sur une seule ligne. Comme précédemment, nous faisons l'hypothèse que chaque caractère peut être lu séparément. Par contre, l'espace n'est plus utilisable comme délimiteur puisqu'il figure entre les différents mots de la phrase. Nous supposons donc tout simplement que celle-ci se termine par un point.**

```

Variable nc : numérique
Variable c : caractère
nc ← 0
répéter
    Lire c
    nc ← nc + 1
Tant que c <> « . »
Ecrire « La phrase contient »,nc, « caractères »

```

Rmq : le programme nous fournit le nombre total de caractères de la phrase, espace et point compris.

Exercice IX : Compter le nombre de lettres d'un mot. On suppose qu'il est suivi d'un espace. Ici, on prendra soin de ne pas comptabiliser cet espace dans le nombre cherché.

```
Variable Nl : numérique
Variable Lettre : caractère
Nl ← 0
répéter
    Lire Lettre
    Nl ← Nl + 1
Tant que lettre <> « \ »
Nl ← Nl - 1
Ecrire « Le mot comporte », Nl, « lettres »
```

Exercice X : On lit des notes tapées au clavier. On souhaite savoir combien d'entre elles sont supérieures à 10. Pour terminer la saisie des notes, on fournira une valeur négative.

Rmq : il s'agit d'un problème de comptage sélectif.

Une petite difficulté réside dans la condition d'arrêt de la boucle jusqu'à qui va permettre de prendre en compte chacune des notes. En effet, la valeur 0 n'est plus utilisable car elle peut représenter une note. Nous choisirons (par convention) de fournir une valeur négative comme dernière donnée.

```
Variable Note, n : numériques
n ← 0
répéter
    Lire Note
    Si Note > 10 alors n ← n + 1 Fsi
Tant que note >= 0
Ecrire « Il y a », n, « notes supérieures à 10 »
```

Rmq : ici la dernière valeur fournie (négative) n'a aucune influence sur la valeur finale de n.

Exercice XI : Lire des notes. Compter le nombre de ces notes comprises entre 5 et 15. Comme dans l'exercice précédent, on fournira une dernière valeur négative.

```
Variable Note, n : numériques
n ← 0
répéter
    Lire Note
    Si Note ≥ 5 et Note ≤ 15 alors n ← n + 1 Fsi
Tant que note >= 0
Ecrire « Il y a », n, « notes comprises en 5 et 15 »
```

Exercice XII : Compter le nombre de voyelles d'une phrase se terminant par un point.

```
Variable Nv : numérique
Variable c : caractère
Nv ← 0
Répéter
    Lire c
    Si c =« a » ou c =« e » ou c =« i » ou c =« o » ou c =« u » ou c =« y »
        alors Nv ← Nv + 1
Tant que c <> « . »
Ecrire « La phrase comporte », Nv, « voyelles »
```

Exercice XIII : Lire des notes. Dire quel est le pourcentage de notes supérieures à la moyenne 10. Là encore, la dernière valeur sera négative.

Ici, il faut compter d'une part le nombre total  $N_t$  de notes, d'autre part le nombre  $n$  de notes supérieures à 10. Autrement dit, nous avons à la fois un comptage systématique et un comptage sélectif. Notez que cette fois, il faudra tenir compte de ce que la dernière valeur (négative) a été comptée. Pour cela, on enlèvera un à la valeur finale de  $n$ . Enfin, le rapport  $N_t/n$  nous fournira le résultat cherché. Pour l'exprimer sous forme d'un pourcentage, il suffira de le multiplier par 100

```
Variable Note, n, Nt, Pourcent : numériques
Nt ← 0
n ← 0
répéter
    Lire Note
    Nt ← Nt + 1
    Si Note > 10 alors n ← n + 1 Fsi
Tant que Note >= 0
Nt ← Nt - 1
Pourcent ← n / Nt * 100
Ecrire « Notes supérieures à 10 : », Pourcent, « % »
```

Rmq : Attention à la division par 0 si la première valeur rentrée est négative. Dans un programme réel, il faut ajouter des instructions permettant d'éviter ce type de problèmes. Exemple, ne pas effectuer le calcul de Pourcent si  $N_t = 0$ .

Exercice XIV : Déterminez le pourcentage de lettres « e » d'une phrase. Les espaces et le point compteront dans le nombre total de caractères.

Il suffit de compter à la fois le nombre total de caractères et le nombre de lettres « e ».

Nous utilisons les variables suivantes :

- $N_t$  : nombre total de caractères
- $N_e$  : Nombre de lettres « e »
- Pourcent : pourcentage de lettres « e ».

```
Variable Nt, Ne, Pourcent : numérique
Variable c : caractère
Nt ← 0
Ne ← 0
Répéter
    Lire c
    Nt ← Nt + 1
    Si c =« e » alors Ne ← Ne + 1
Tant que c <> « . »
Pourcent ← Ne / Nt * 100
Ecrire « Il y a », Pourcent, « % de lettres e »
```

Exercice XV : Une phrase terminée par un point est saisie. En dehors du point final, elle ne comporte que des lettres et des espaces. Compter le nombre de lettres de cette phrase (notez qu'il ne s'agit plus du nombre total de caractères, mais simplement du nombre de lettres ; il faut donc éviter de compter les espaces et le point).

```
Variable Nl : numérique
Variable c : caractère
Nl ← 0
Répéter
    Lire c
    Si c ≠ « \ » et c ≠ « . » alors Nl ← Nl + 1 FSI
Tant que c <> « . »
Ecrire « La phrase comporte », Nl, « lettres »
```

Exercice XVI : Déterminez le pourcentage de lettres « e » d'une phrase fournie ne comportant, en dehors du point final que des lettres et des espaces. Cette fois, point et espaces ne doivent plus intervenir sur le nombre total de lettres.

Variable Nl, Ne, Pourcent : numérique

Variable c : caractère

Nl ← 0

Ne ← 0

Répéter

    Lire c

    Si ≠ « \n » et c ≠ « . » alors Nl ← Nl + 1 Fsi

    Si c =« e » alors Ne ← Ne + 1 FSI

Tant que c < > « . »

Pourcent ← Ne / Nl \* 100

Ecrire « Il y a », Pourcent, « % de lettres e »