**LES FONCTIONS**

**Exercice 9.1**

Parmi ces affectations (considérées indépendamment les unes des autres), lesquelles provoqueront des erreurs, et pourquoi ?

Variables A, B, C en Numérique

Variable D en Caractère

A ← Sin(B)

A ← Sin(A + B \* C)

B ← Sin(A) – Sin(D)

C ← Sin(A / B)

C ← Cos(Sin(A)

**Correction**

**DÉBUT**

**A = Sin(1) OK**

**A = Sin(1 + 2 \* 3) OK**

**B = Sin(1) - Sin("D") IMPOSSIBLE**

**C = Sin(1 / 0) IMPOSIBLE SI B = 0**

**C = Cos(Sin(1)) il manque une parenthèse**

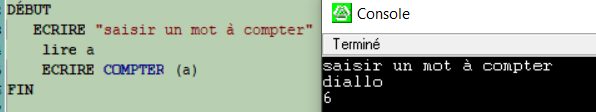
**FIN**

### Exercice 9.2

Ecrivez un algorithme qui demande un mot à l’utilisateur et qui affiche à l’écran le nombre de lettres de ce mot (c'est vraiment tout bête).

**DÉBUT**

**ECRIRE "saisir un mot à compter"**

 **lire a**

**ECRIRE COMPTER (a)**

**FIN**

### Exercice 9.3

Ecrivez un algorithme qui demande une phrase à l’utilisateur et qui affiche à l’écran le nombre de mots de cette phrase. On suppose que les mots ne sont séparés que par des espaces (et c'est déjà un petit peu moins bête).

**DÉBUT**

**nbr =0**

**SEPARATEUR ","**

**ECRIRE "Donnez une phrase"**

**LIRE phr**

**tab = ENCARAC (phr)**

**TABSIZE = LONGUEUR(phr)**

**POUR i = 1 jusqu'à TABSIZE FAIRE**

**SI tab[i] != "" ALORS**

**nbr = nbr + 1**

**FINSI**

**FINPOUR**

**ECRIRE nbr**

**FIN**

### Exercice 9.4

Ecrivez un algorithme qui demande une phrase à l’utilisateur et qui affiche à l’écran le nombre de voyelles contenues dans cette phrase. On pourra écrire deux solutions. La première déploie une condition composée bien fastidieuse. La deuxième, en utilisant la fonction Trouve, allège considérablement l'algorithme.

**DÉBUT**

**ECRIRE "Entrez une phrase"**

**lire wph**

**tabph = ENCARAC(wph)**

**wvoyelle = ["a","e", "i", "o", "u", "y"]**

**wnbrvoyelle = 0**

**i = 1**

**TANTQUE (i <= COMPTER (tabph)) FAIRE**

**j = 1**

**TANTQUE (j <= COMPTER (wvoyelle)) FAIRE**

**SI (wvoyelle [j] = tabph [i]) ALORS**

**wnbrvoyelle = wnbrvoyelle + 1**

**j = 7**

**FINSI**

**j = j + 1**

**FINTANTQUE**

**i = i + 1**

**FINTANTQUE**

**ECRIRE wnbrvoyelle**

**FIN**

### Exercice 9.5

Ecrivez un algorithme qui demande une phrase à l’utilisateur. Celui-ci entrera ensuite le rang d’un caractère à supprimer, et la nouvelle phrase doit être affichée (on doit réellement supprimer le caractère dans la variable qui stocke la phrase, et pas uniquement à l’écran).

**DÉBUT**

**ECRIRE "Entrez une phrase"**

**lire phrase**

**ECRIRE " A quel rang supprimer un caractère ?"**

**lire posi**

**LONG = LONGUEUR (phrase)**

**mvphr = sousensemble (phrase, 1, posi - 1) + sousensemble (phrase, posi + 1, LON)**

**ECRIRE mvphr**

**FIN**