**DEFINIR LES STRUCTURES DE DONNEES :**

En assembleur pour définir des structures de données on utilise des **procédures**. Les procédures sont des sous-programmes qui permettent d'effectuer un ensemble d'instructions. Elles sont généralement appelées fonctions dans d'autres langages tels que Python, JavaScript, etc.

Les différentes directives utilisées pour mettre en place les procédures sont les suivantes :

* La directive **PROC,** qui est la principale, permet de créer une fonction. Elle est précédée du nom de la fonction et est suivie de la directive **near** qui permet de signaler que la procédure est située dans le même segment que le programme.
* La directive **CALL** qui permet d’appeler notre fonction.
* La directive **RET** qui permet de désigner l’instruction final et de mettre fin à la procédure.
* La directive **ENDP** qui permet de mettre fin à la procédure.

En assembleur les registres sont utilisés comme variables. Nous utiliserons donc les registres suivants :

Pour les registres généraux :

* **AX** (Accumulateur) qui permet de stocker les résultats de certains calculs logiques et arithmétiques. Ainsi que des valeurs entières (Int).

* **BX** (Base) qui permet de stocker les résultats des résultats de certains calculs arithmétiques et d’adresses.
* **CX** (Compteur) pour créer des boucles.

Registres de segments :

* **DS** enregistre les données des registres définis.

Pour mettre en place les boucles nous aurons besoin d’opérateurs suivis d’instructions tel que :

* **CMP** opérateur qui permet de comparer deux registres. Il ne concerne que les boucles “while“ et “for“.
* **JA** instruction qui permet de vérifier si un registre est supérieur à un autre, il ne concerne que les boucles “for“.
* **JE** instruction qui permet de vérifier si un registre vaut 0.
* **JL** instruction qui permet de vérifier si un registre est inférieur à un autre.
* **JLE** instruction qui permet de vérifier si un registre est inférieur ou égal à un autre.
* **JMP** instruction qui permet d’indiquer l’emplacement où doit-être poursuivi l’exécution de notre boucle.