

## Bibliothèque

Ce module offre la possibilité, pendant la phase de compilation, d'insérer dans un programme COBOL n'importe quel texte source enregistré dans une bibliothèque du système.

$$\begin{array}{c} \underline{\text{COPY}} \text{ nom-texte} \left( \begin{array}{l} \left\{ \begin{array}{l} \underline{\text{OF}} \\ \underline{\text{IN}} \end{array} \right\} \text{ bibliothèque} \end{array} \right) \\ \\ \left( \left\{ \begin{array}{l} \underline{\text{REPLACING}} \\ \text{nd-1} \\ \text{lit-1} \\ \text{mot-1} \end{array} \right\} \right) \underline{\text{BY}} \left( \begin{array}{l} \left\{ \begin{array}{l} \text{nd-2} \\ \text{lit-2} \\ \text{mot-2} \end{array} \right\} \\ \left\{ \begin{array}{l} \text{...} \end{array} \right\} \end{array} \right) \end{array}$$

- ◆ l'instruction `COPY` doit être suivie d'un point
- ◆ `mot` désigne un mot COBOL quelconque
- ◆ `pseudo-text-2` peut être vide, mais non `pseudo-text-1`
- ◆ le texte à copier ne peut contenir de directive `COPY`
- ◆ les remplacements portent sur **toutes** les occurrences des éléments cités

En ANS85, l'instruction `REPLACE` permet des remplacements partiels

Format 1      `REPLACE {== pseudo-text-1== BY == pseudo-text-2==} ...`

Format 2      `REPLACE OFF`

- ◆ les règles sont les mêmes que pour `COPY`
- ◆ `COPY` a précédence sur `REPLACE`
- ◆ le texte est remplacé jusqu'à rencontre de la prochaine instruction `REPLACE` (avec une nouvelle spécification ou `OFF`) ou, par défaut, de la fin du programme.