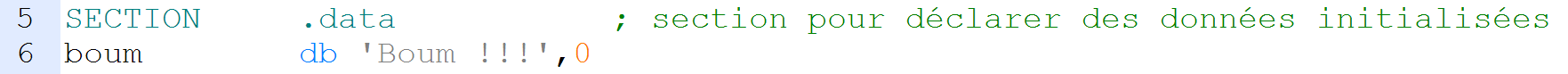
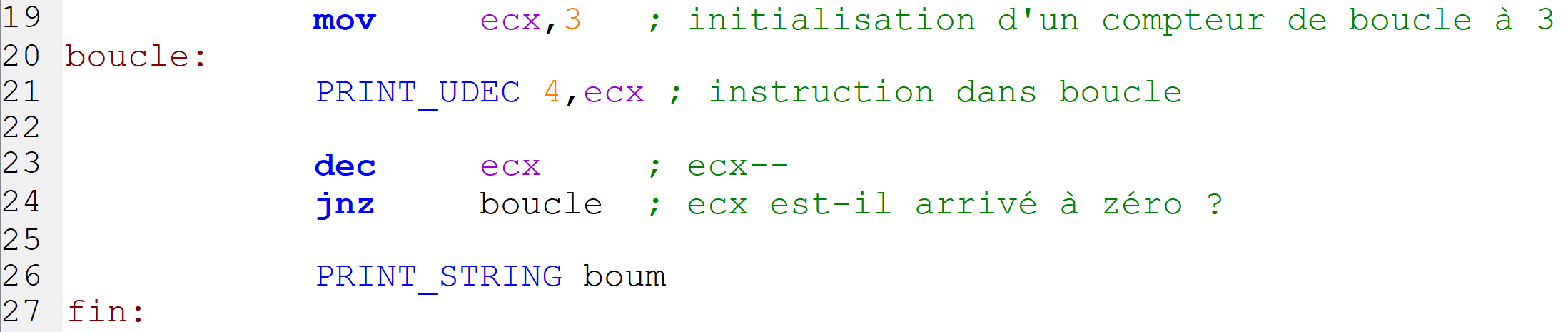
**Les boucles en NASM**

**Do..while :**

* + **étiquette:**
  + **Entrée dans la boucle à l'étiquette (adresse)**
  + **…**
  + **Instructions dans la boucle**
  + **…**
  + **Test de fin de boucle (vrai ou faux) ?**
    - **Retourner au début de la boucle à l’étiquette:**
    - **Ou, continuer la suite du programme**

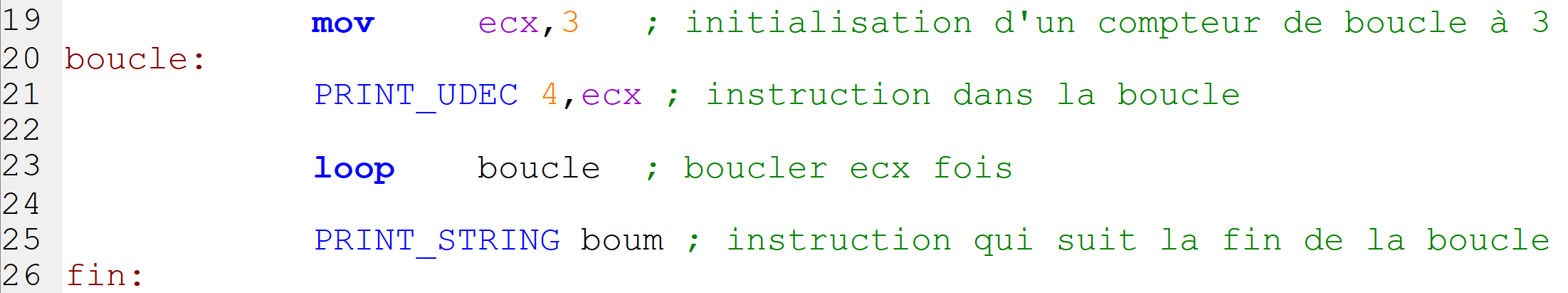
Ex :





Affiche : 321 boum !!!

* **For : Avec l’instruction LOOP**
  + **Initialiser un compteur de boucle dans le registre ECX**
  + **Commencer à écrire les instructions de la boucle à partir d’une étiquette, par exemple « boucle: »**
  + **Après les instructions de la boucle, écrire LOOP boucle, « boucle » étant l’étiquette choisie pour le début de la boucle**
* **Ainsi la boucle se fera ECX fois**



* **LOOP adresse**
  + **Décrémente ECX et si ECX!=0, saute (branche) à l'instruction qui se trouve à l'adresse spécifiée (retourne au début de la boucle)**
  + **Sinon (ECX==0), passe à l'instruction suivante (quitte la boucle)**

**While :**

* Un "*while"* en assembleur
  + Entrée à l'étiquette (adresse) étiquette:
  + Test de début de boucle (vrai ou faux) ?
    - Continuer vers les instructions dans la boucle
    - Ou, aller à la fin de la boucle, à l’étiquettedefin:
  + …
  + Instructions dans la boucle
  + …
  + Retourner au test de début de boucle avec JMP étiquette
  + étiquettedefin:

