

<https://laurentbloch.org/BlogLB/Lire-une-sequence-SwissProt>



Lire une séquence SwissProt

- Zinformatiques - Cours de bioinformatique au CNAM -

Date de mise en ligne : jeudi 19 janvier 2006

Copyright © Blog de Laurent Bloch - Tous droits réservés

Tout d'abord, lire le [manuel de référence](#) de la banque SwissProt. Regarder [une séquence](#).

Le petit programme se propose de lire une séquence dans une copie locale d'un extrait de la banque, la première séquence du fichier par exemple, et de placer les lignes successives dans une liste. La seule mise en forme sera de produire la séquence proprement dite sous forme d'une unique chaîne de caractères, sans espaces ni sauts de ligne.

```
(define (lire-une-sequence flux)
  (let boucle ((L '()))
    (ligne (read-line flux)))
    (if (eof-object? ligne)
      ligne
      (let ((type-ligne (substring ligne 0 2)))
        (cond ((string=? type-ligne "//")
              (reverse (cons ligne L)))
              ((string=? type-ligne " ")
              (aligne-aa ligne L flux))
              (else
              (boucle (cons ligne L)
                      (read-line flux)))))))

(define (aligne-aa ligne L flux)
  (let boucle ((source ligne)
                (L2 L))
    (sequence ""))
    (if (string=? (substring source 0 2) "//")
      (reverse (cons source
                    (cons sequence L2)))
      (do ((i 3 (+ 1 i))
          (les-aa
           ""))
        (let ((le-char
                (substring source i (+ 1 i))))
          (if (string=? " " le-char)
              les-aa
              (string-append les-aa
                              le-char))))
          ((= i (string-length source))
           (boucle (read-line flux)
                   L2
                   (string-append sequence les-aa))))))
```