
Polinomis amb forats**X47545_ca**

Representarem un polinomi com la llista que conté els seus coeficients en ordre creixent per grau. Per exemple, si p és $4x^6 + 2x^5 + 3x + 9$ llavors es representa com $[9, 3, 0, 0, 0, 2, 4]$. És a dir, a cada posició i de la llista, es guarda el coeficient del monomi de grau i del polinomi. Noteu que l'últim coeficient que es guarda és el de grau màxim (4 a l'exemple). Com a conseqüència, les llistes que representen polinomis no nuls, sempre han de tenir un valor diferent de zero a l'última posició.

Donada una llista q que representa un polinomi no nul d'enters, es diu que $q[i : j]$ és un forat si $q[i : j] == [0] * (j - i)$ i $q[j] \neq 0$. La longitud d'un forat $q[i : j]$ és $j - i$. Per exemple, el polinomi p d'amunt té un forat de longitud 3 a la posició 2, perquè $p[2 : 5] == [0, 0, 0] == [0] * (5 - 2)$ i $p[5] \neq 0$.

Considereu la següent funció `len_forat(p, i)` que donat un polinomi no buit p , retorna la longitud del forat que comença a la posició i , $0 \leq i \leq \text{len}(p) - 1$:

```
def len_forat(p, i):  
    '''  
    Donada una llista p que representa un polinomi no buit d'enters,  
    i una posicio i, 0 <= i <= len(p)-1, retorna la longitud del forat  
    a la posicio i  
    >>> len_forat([9, 3, 0, 0, 0, 2, 4], 2)  
    3  
    >>> len_forat([9, 3, 0, 0, 0, 2, 4], 1)  
    0  
    '''  
    j = i  
    while p[j] == 0:  
        j += 1  
    return j-i
```

Implementeu una funció `cerca_forat(p, n)` que retorni les dues posicions $(i, j - 1)$, tal que $p[i : j]$ és el primer forat de longitud més gran o igual que n del polinomi no nul p . Si p no té cap forat més llarg o igual que n llavors ha de retornar $(-1, -1)$. És obligatori que la implementació de `cerca_forat(p, n)` usi `len_forat(p, i)` i que sigui una cerca.

Exemple de sessió

```
>>> cerca_forat([9, 3, 0, 0, 0, 2, 4], 2)  
(2, 4)  
>>> cerca_forat([9, 3, 0, 0, 0, 2, 4], 3)  
(2, 4)  
>>> cerca_forat([9, 3, 0, 0, 0, 2, 4], 4)  
(-1, -1)  
>>> cerca_forat([0, 0, 9, 3, 0, 0, 0, 2, 4], 3)  
(4, 6)  
>>> cerca_forat([1, 1, 1], 0)  
(0, -1)
```

Informació del problema

Autor : InfBesos

Generació : 2025-10-20 11:27:08

© *Jutge.org*, 2006–2025.

<https://jutge.org>