

---

**F016A. Naturals a cadenes****P30055\_ca**

---

Implementeu una funció

```
string nat_a_cadena(int x, int n, int b);
```

que retorni una cadena de  $n$  caràcters que representi el natural  $x$  en base  $b$ .

En el cas que el nombre  $x$  en base  $b$  no ocupi tots els  $n$  caràcters, cal omplir la cadena amb sostinguts per l'esquerra. En el cas que ocupi més de  $n$  caràcters, cal retornar una cadena amb  $n$  asteriscos. Representeu els dígits superiors a 9 amb 'A', 'B', ..., 'F'.

**Observacions**

- El programa principal ja se us dóna implementat; no el canvieu. Aquest llegeix triplets d'enters  $x$ ,  $n$ ,  $b$  i escriu el resultat de la crida a la funció que heu d'implementar:

```
int main() {  
    int x, n, b;  
    while (cin >> x >> n >> b) cout << nat_a_cadena(x, n, b) << endl;  
}
```

- Us suggerim definir i utilitzar una funció **int** nombre\_digits(**int** x, **int** b); que retorni el nombre de dígits del natural  $x$  en base  $b$ .
- Recordeu que un string  $s$  amb  $n$  caràcters  $c$  es pot declarar així: **string** s( $n$ ,  $c$ );
- Recordeu també que les operacions dels strings com ara  $s += '0'$ ; o bé  $s1 += s2$ ; o bé  $s = s1 + s2$ ; estan prohibides.

**Precondició**

Es té que  $x \geq 0$ ,  $2 \leq b \leq 16$  i que  $n > 0$ .

### Exemple d'entrada

```
12345 5 10
12345 7 10
12345 4 10
12345 4 16
15 2 16
15 2 15
15 2 10
15 2 8
15 2 4
15 2 3
666 11 2
1048576 21 2
1048576 21 4
1048576 21 8
1048576 21 16
0 1 8
1 1 9
0 45 10
1 45 11
```

### Exemple de sortida

```
12345
##12345
****
3039
#F
10
15
17
33
**
#1010011010
1000000000000000000000
#####10000000000
#####4000000
#####100000
0
1
#####0
#####1
```

### Informació del problema

Autor : Professorat de P1  
Generació : 2025-05-13 11:02:06

© *Jutge.org*, 2006–2025.  
<https://jutge.org>