

---

## Iconos ASCII

X81846\_es

---

Haz un programa que recibe la codificación de un icono en dos colores y lo muestra a la salida utilizando caracteres ASCII. Los iconos tienen un tamaño de  $20 \times 20$  píxeles y cada línea de píxeles horizontal se codifica utilizando 2 números enteros,  $p$  y  $q$ . Los números  $p$  y  $q$ , en base 2, nos dicen qué píxeles de cada fila hay que llenar: cada bit con valor 1 se debe pintar y cada bit con valor 0 no se debe pintar.

Así pues, los valores enteros de  $p$  y  $q$  están en el intervalo  $[0, 2^{20} - 1]$ , y  $p$  codifica los píxeles que hay que rellenar con el color primario, que mostraremos con el carácter "#", y  $q$  codifica los píxeles que hay que llenar con el color secundario, para el que usaremos ".". Si tanto  $p$  como  $q$  tienen un bit a 1 en la misma posición, el color primario tiene prioridad y en esa posición escribiremos '#'. Dado que escribiremos los caracteres en ASCII, allí donde no haga falta pintar ningún color pondremos un espacio para que la fila de píxeles se mantenga en su posición. Aun así, los espacios al final de la línea no se deben escribir.

Veamos un ejemplo, tomando  $p = 132$  y  $q = 124$ . Pasándolos a base 2, y añadiendo ceros a la izquierda para llegar a las 20 cifras, tenemos, respectivamente:

```
0000000000000010000100
0000000000000001111100
```

Aplicando las reglas mencionadas más arriba el resultado será:

```
_____#....#
```

donde hemos usado el carácter '\_' para hacer visibles los espacios. Hay que observar dos cosas: 1) en la parte final no hay los dos espacios que corresponden a los dos ceros a la derecha de  $p$  y  $q$ , y 2)  $p$  y  $q$  tienen un 1 en la misma posición (tercera comenzando por la derecha) y en la salida tenemos '#'.

### Entrada

La entrada consiste en un entero  $n$  que indica cuantos iconos habrá que escribir. Después vienen  $n$  secuencias de  $20 \times 2 = 40$  enteros entre 0 y  $2^{20} - 1$  que codifican los dos colores de cada línea horizontal de píxeles del icono.

### Salida

La salida es cada una de los  $n$  iconos pintado con caracteres ASCII, es decir, 20 líneas de texto, seguidas de una línea en blanco. El texto en blanco al final de cada línea no se debe escribir, ya que no contribuye a mantener la posición de los píxeles pintados. Por tanto, las líneas de texto de la salida pueden tener desde 0 caracteres (no tienen píxeles pintados), hasta 20 caracteres (el último píxel de la línea está pintado).

### Ejemplo de entrada

2

```
0 0 0 0 0 0 0 0 29120 0
35360 29120 66576 64480 131208
```

```
130928 139784 122352 159816 102320
139528 122608 65552 65504 32800
32704 29120 3584 2560 1024
1024 0 0 0 0 0 0 0 0 0
0 0 0 0 8064
```

```
0 8256 8064 16416 16320
32784 32736 47120 18400 48144
17376 48528 16992 23968 8768
8256 8064 8064 0 8256
8064 16320 0 8256 8064
16320 0 4224 3840 3840
0 0 0 0 0
```

## Ejemplo de salida

```
###  ###
#...# #...#
#.....#.....#
#.....#.....#
#...#...#...#
#..###...#..#
#...#...#...#
#.....#.....#
#.....#.....#
###...###
#.#
#
```

```
#####
#.....#
#.....#
#.....#
#..###...#..#
#..###...#..#
#..###...#..#
#..###...#..#
#.....#
#####
#.....#
#####
#.....#
#####
#.....#
####
```

## Información del problema

Autor : PRO1

Generación : 2024-03-03 22:45:17

© [Jutge.org](https://jutge.org), 2006–2024.

<https://jutge.org>