

MCM

El MCM¹ de dos números naturales es el más pequeño entre los múltiplos comunes de ambos números. Así, por ejemplo, dado que los múltiplos de 15 son los números 15, 30, 45, 60... y los de 6 son los números 6, 12, 18, 24, 30, 36..., tenemos $MCM(15, 6) = 30$.

Es bien conocido que el producto de dos números coincide con el de su MCM y su MCD.

1. Utilizando esta propiedad, defina en OCaml una función ***mcm*** : *int* -> *int* -> *int*, tal que, para cualesquiera $x > 0$ e $y > 0$, *mcm* *x* *y* sea el MCM de *x* e *y*. Debe incluir esta definición en un archivo con nombre ***mcm.ml***

La definición debe hacerse de modo que siempre que el MCM de *x* e *y* sea menor o igual que *max_int*, *mcm* *x* *y* devuelva correctamente su valor.

Así por ejemplo, si *max_int* = $2^{62} - 1$, deberían obtenerse los siguientes resultados:

```
# mcm 15 6;;
- : int = 30
# mcm 540 84;;
- : int = 3780
# mcm 2100 198;;
- : int = 69300
# mcm 12884901888 32212254720;;
- : int = 64424509440
# mcm 90352612350 11394068049;;
- : int = 3214836300025350
# mcm 2305843009213693952 1152921504606846976;;
- : int = 2305843009213693952
```

Defina (también en el archivo ***mcm.ml***) una función ***mcm'*** : *int* -> *int* -> *int* que *perfeccione* la definición de *mcm*, en el sentido de que si uno de los argumentos no es mayor que 0 o si el MCM de ambos es mayor que *max_int*, devuelva el *int* (-1) para indicar *el error*. Esto es realmente *poco elegante*, pero ya aprenderemos a *hacerlo mejor*...

```
# mcm' 90352612350 11394068049;;
- : int = 3214836300025350
# mcm' 2305843009213693952 1152921504606846976;;
- : int = 2305843009213693952
# mcm' 0 3;;
- : int = -1
# mcm' 1 (-1);;
- : int = -1
# mcm' 2147483649 2147483650;;
- : int = -1
# mcm' 4294967297 2147483648;;
- : int = -1
```

¹ "mínimo común múltiplo". Se usa, por ejemplo, para establecer un denominador común para la suma de fracciones.