Consigna: Dados dos números enteros m y n,construir una función recursiva que devuelva el producto de ambos, calculando el mismo como sumas sucesivas. Esto es:
m*n=m+m+...+m, n veces.
Ejemplos:
producto(5, 3) => 15
producto (10, 4) => 40
Solución: Sabemos que un número m multiplicado n, es igual a $(n + n)$ m veces.
Caso base: si el multiplicador igual 1, el resultado es el multiplicando.
Si el multiplicador fuera 0, el resultado es 0.
A partir de estos casos base, construiremos casos más simples de manera tal que podamos resolver de manera recursiva.

Tomamos un ejemplo simple: Supongamos que $m = 2$ y $n = 3$. Sabemos que esto es $2 + 2 + 2$

$$\begin{aligned} 2+2+2 \\ 2+4 \\ 6 \end{aligned}$$

Como podemos observar, $2 * 3$ es el resultado de $(2 * 1 + 2 * 2)$. Siendo $2 * 1 = 2 + 0$ y $2 * 2 = 2 + 2$
Sea $M(m, n)$ una función que denota la multiplicación de m por n; donde $m, n \in \mathcal{Z}^+$:

$$M(m,n) = \begin{cases} M(m,n) = 0; & n = 0 \\ M(m,n) = m; & n = 1 \\ M(m,n) = m + M(m, n-1); & m, n > 1 \end{cases}$$

Resolviendo el caso de 2*3 basándonos en la definición anterior.

$$M(2,3) = 2 + M(2,2) \tag{1}$$
$$M(2,2) = 2 + M(2,1) \tag{2}$$
$$M(2,1) = 2 + M(2,0) \tag{3}$$
$$\text{Substituyendo los valores encontrados y viéndolo de «atrás para adelante»} \tag{4}$$
$$M(2,1) = 2 + [\mathbf{M(2,0)} = \mathbf{0}] = 2 \tag{5}$$
$$M(2,2) = 2 + [\mathbf{M(2,1)} = \mathbf{2}] = 4 \tag{6}$$
$$M(2,3) = 2 + [\mathbf{M(2,2)} = \mathbf{4}] = 6 \tag{7}$$

Hasta acá hemos definido la función solo para los enteros positivos. Ahora procederemos a desarrollar para que que la función también se encuentre definida cuando $m < 0$ o $n < 0$. Sabemos que si m y n son negativos, entonces es lo mismo que $-m * -n = [(-1) * (-1)] * (m * n) = 1 * (m * n) = m * n$, por lo tanto en este caso se sigue aplicando el mismo procedimiento para cuando m y n son positivos. Cuando el multiplicador es negativo y el multiplicando positivo, o el multiplicador positivo y el multiplicando negativo, el procedimiento se comprende como una serie de restas sucesivas, esto es: Si

$$(m < 0 \vee n < 0) \wedge \neg(m < 0 \wedge n < 0)$$

Entonces $m * n = (-m + (-m) + \ldots + (-m))$ esto sucediendo n veces.
Supongamos el caso $-2 * 3$:

$$-2 * 3 = -2 + -2 + -2 = -6$$

Si extendemos la definición anterior para los negativos tenemos: **Sean $m, n \in \mathcal{Z}$;Donde $|x|$ denota el valor absoluto de x:**

$$M(m,|n|) = \begin{cases} M(m,n) = 0; & n = 0 \\ M(m,n) = m; & n = 1 \\ M(m,n) = m + M(m, n-1); & m, n > 1 \\ M(m,n) = -|m| - |M(m, |n| - 1)|; & (m < 0 \vee n < 0) \wedge \neg(m < 0 \wedge n < 0) \end{cases}$$

Entonces $2 * (-3)$:

$$M(2,-3) = -2 - |M(2,2)|$$
$$M(2,2) = -2 - |(2,1)|$$

Substituyendo y aplicando según definición:

$$M(2,2) = -2 - |\mathbf{M(2,1)} = \mathbf{2}| = -4$$
$$M(2,-3) = -2 - |\mathbf{M(2,2)} = \mathbf{4}| = -6$$

Con base a todo lo construido anteriormente se procederá a desarrollar el código:

```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <stdbool.h>
// multiplicacion_recursiva: toma dos numeros enteros y retorna su producto(int).
long int multiplicacion_recursiva(long int multiplicando, long int multiplicador);
long int multiplicacion_recursiva(long int m, long int n)
{
    // Sabemos que si multiplicando y multiplicador son negativos sera equivalente a que ambos sean positivos
    if(n<0 && m<0)
    {
    n = abs(n);
    m = abs(m);
    }

    // CASOS BASE:
    // I) si n = 0, entonces la multiplicacion sera 0.
    // II) si n = 1, entonces la multiplicacion sera igual al multiplicando.

    if(n==0)
    return 0;
    else if(n==1)
        return m;

    // Si alguno de los dos fuera negativo
    if((n < 0 || m < 0))
    return ((-abs(m)) - abs(multiplicacion_recursiva(m, abs(n) - 1)));

    return m + multiplicacion_recursiva(m, n-1);
}
}
```