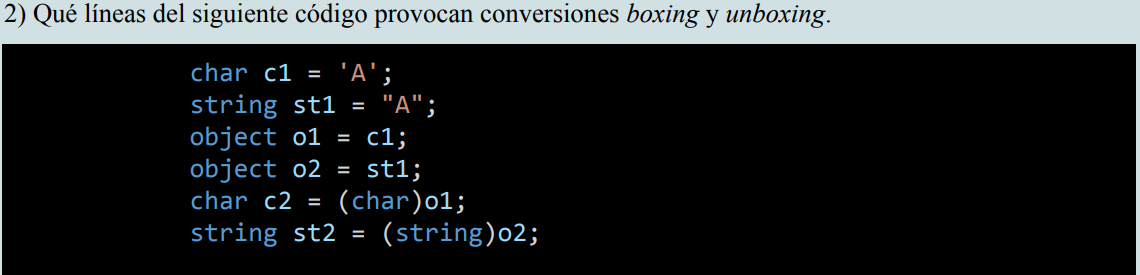
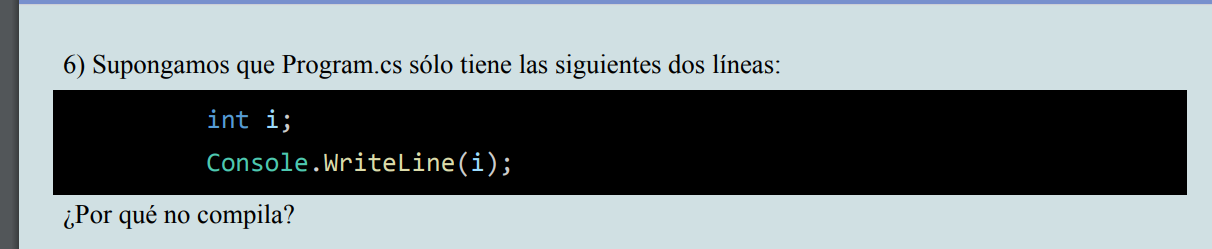


Cuando se le asigna a o2=”Z”; Genera un nuevo objeto Con una nueva dirección de memoria en la pila.

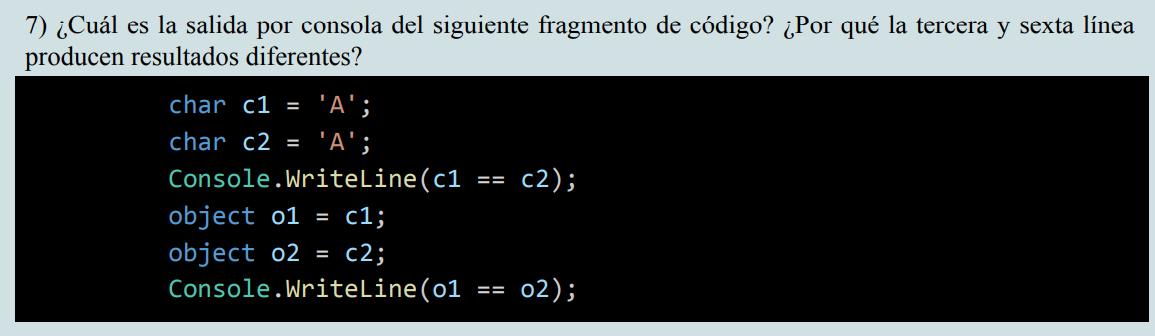


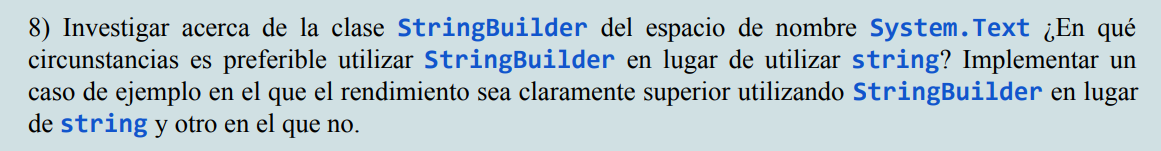
Boxing = Object o1 = c1; //

Unboxing = char c2 = (char)o1; //

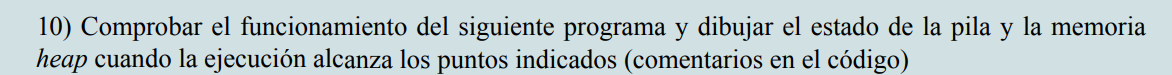


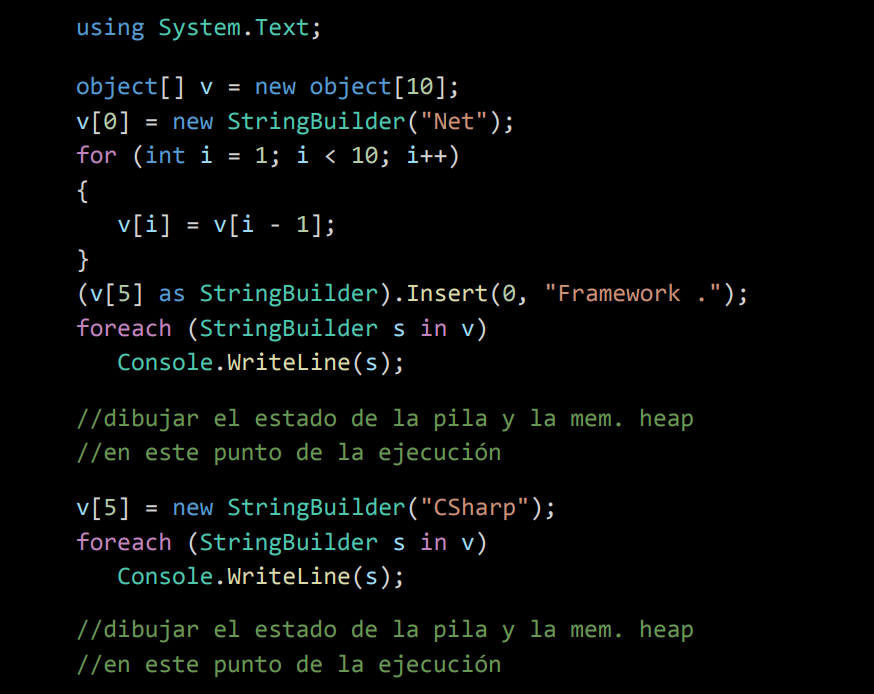
No compila porque i no esta inicializada.

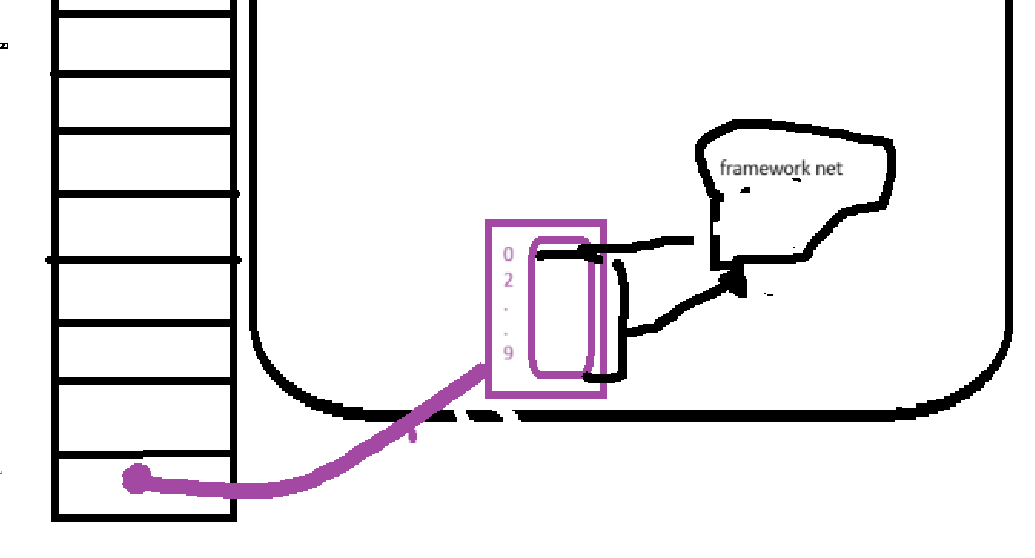
Producen resultados diferentes ya que en la tercera compara dos por tipo de valor por ende analiza el valor de esas variables y el otro analiza de dos variables referencia por ende analiza si apuntan a la misma dirección. (Si se quisiera analizar el contenido debería utilizar o1.Equals(o2).



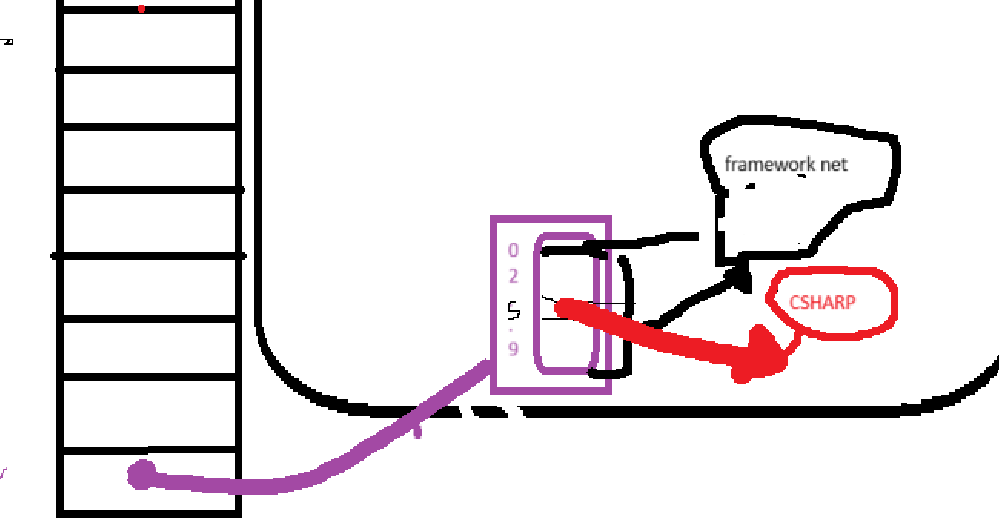
Cuando se quiere se quiere optimizar la memoria conviene utilizar un stringBuilder para no ir creando objetos y ocupando mas memoria.



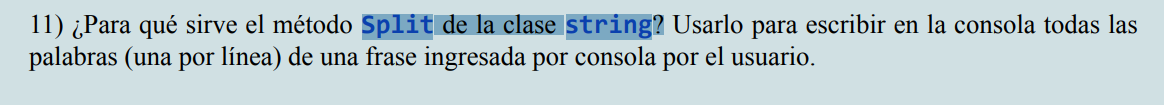




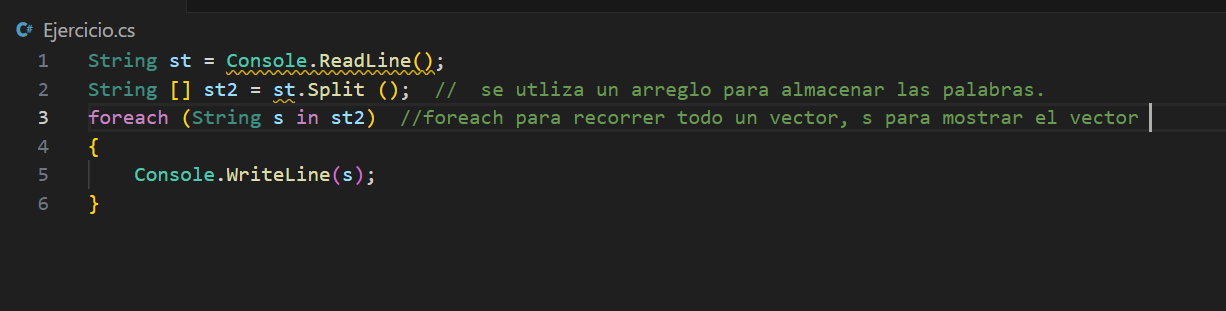
En el primer caso.

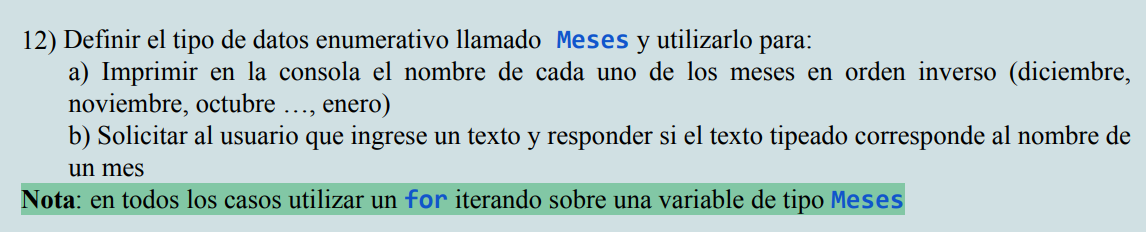


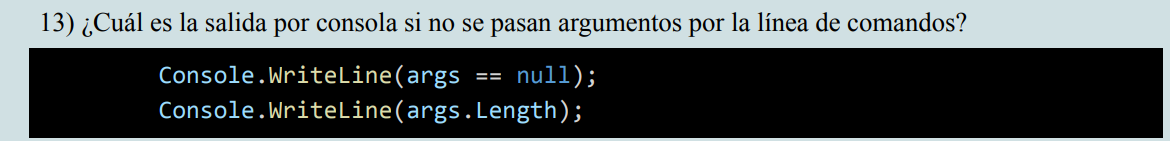
En el segundo.



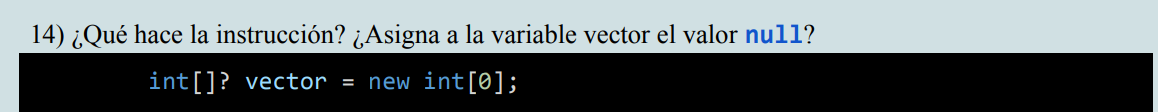
Divide un String es partes mas pequeñas.



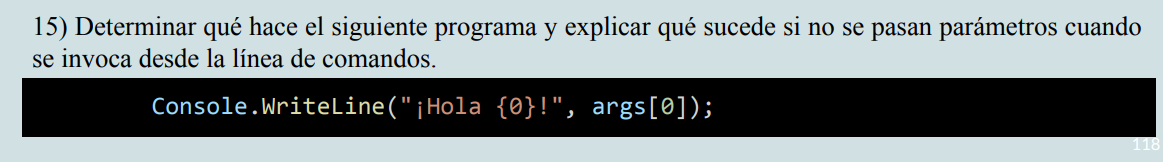




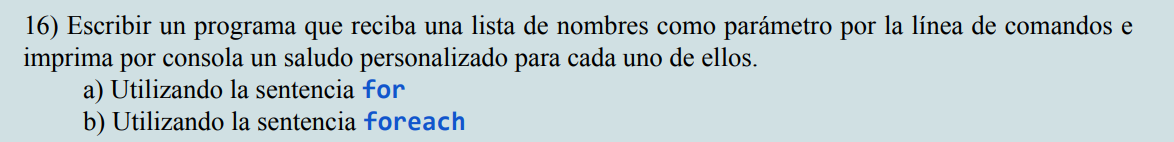
False,0

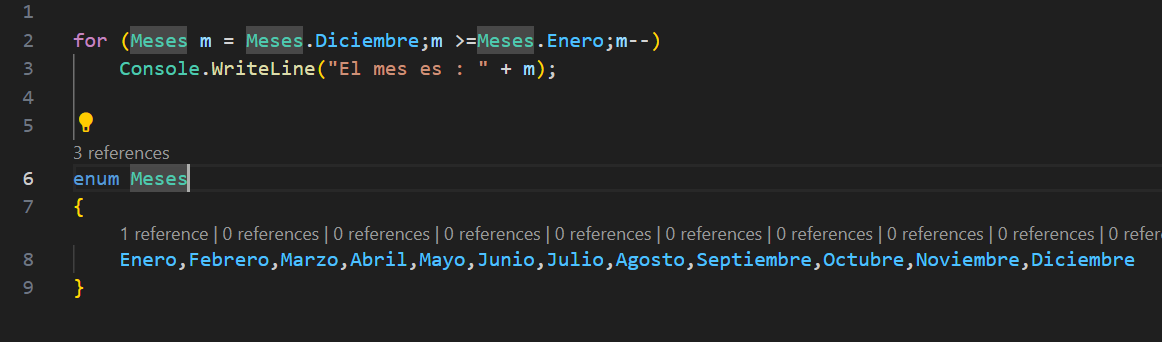


Si bien el vector puede ser null , se crea el vector vacio ósea sin elementos,Por lo que no es null

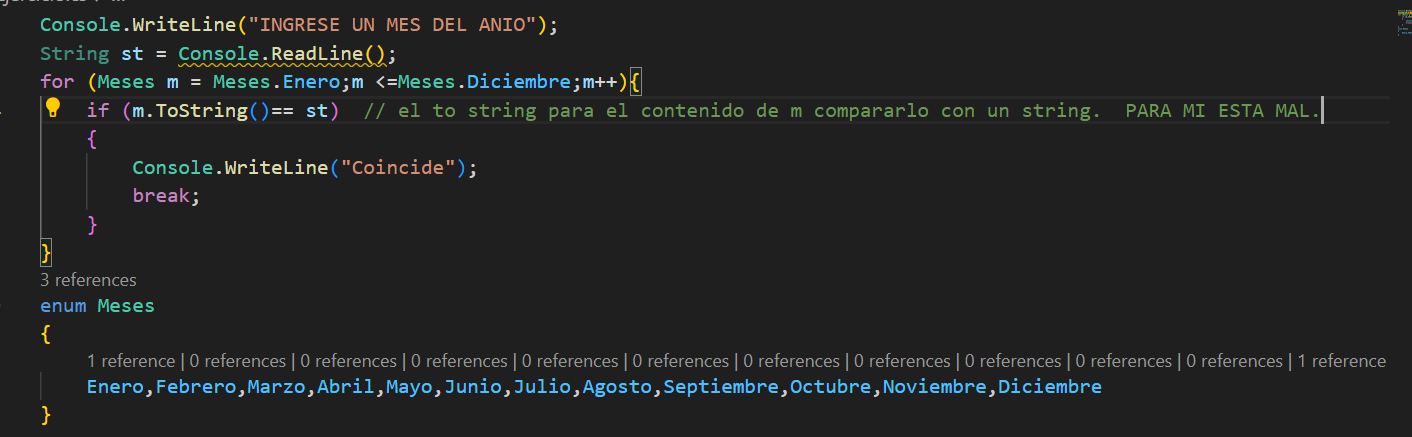


Se quiere acceder a una posicion fuera de rango. Ya que no tienen parámetros el arreglo



a)

b)



16) Escribir un programa que reciba una lista de nombres como parámetro por la línea de comandos e imprima por consola un saludo personalizado para cada uno de ellos. a) Utilizando la sentencia for b) Utilizando la sentencia foreach

a) for (int i=0; i<args.Length;i++){

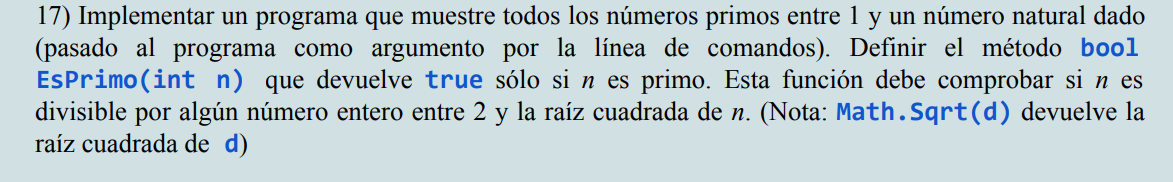
    Console.WriteLine("Hola " + args[i]);

}

b) foreach(String st in args){

    Console.WriteLine("Hola "+ st);

}



int dim = int.Parse(args[0]);

for (int i=0;i<=dim;i++)

{

    if (EsPrimo(i)){   //para cada numero imprimo sus valores primos

        Console.WriteLine("EL NUMERO " + i + " ES PRIMO");

    }

}

bool EsPrimo(int n )

{

    if (n<2)

        return false;

    else

            for (int i=2;i<=Math.Sqrt(n);i++)

            {

                if (n % i == 0){

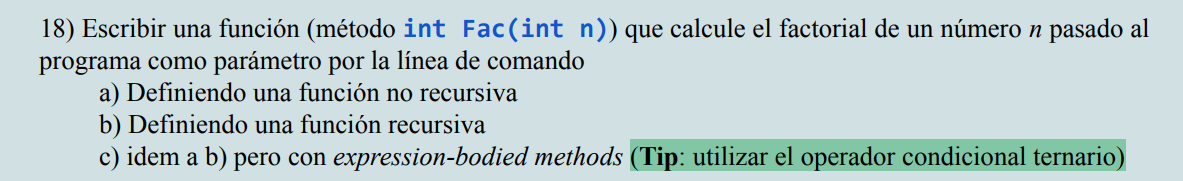
                    return false;

                }

        }

    return true;

}



int num = int.Parse(args[0]);

Console.WriteLine(factorial(num));

int factorial (int n)

{

    int suma = 1;

    if (n == 0 | n ==1 )

        return 1;

    else

        for (int i=n;i>=2;i--)

        { //Normal.

            suma = suma \* i;

        }

    return suma;

}

int num = int.Parse(args[0]);

Console.WriteLine(factorial(num));

int factorial (int n)

{ //factorial

    int suma = n;

    if (n==1 || n==0)

        return 1;

    else

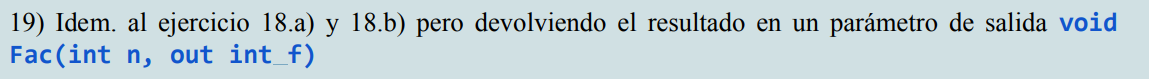
         return n \* factorial(n-1);

}

int num = int.Parse(args[0]);

Console.WriteLine(factorial(num));

int factorial (int n) => (n==1) ? 1 :n \* (factorial(n-1)); //operador ternario



int result = 0;

int num = int.Parse(args[0]);

Factorial(num,out result);

Console.WriteLine(result);

void Factorial (int n,out int result)

{

    int suma = 1;

    if ((n == 1)||(n==0)) /A

        suma = 1;

    else

        for (int i=n;i>=2;i--)

            suma = suma\*i;

    result = suma;

}

int num = int.Parse(args[0]);

int result;

factorial(num,out result);

Console.WriteLine(result);

void factorial (int n, out int result)

{

    int suma = 1 ;

    if (n<=1){

        result = 1;

    } //B

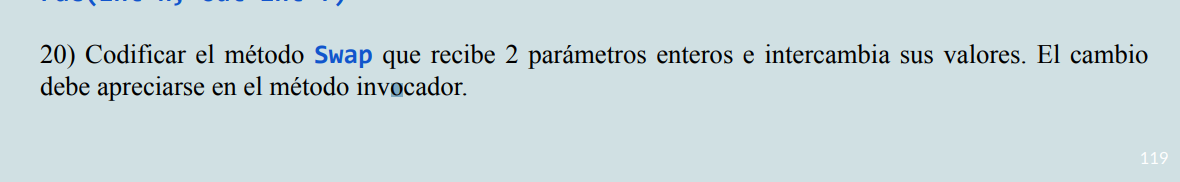
    else

    {

        factorial(n-1,out suma);

        result = n\*suma;

    }                       //factorial



Console.WriteLine("Ingrese dos valores");

int a = int.Parse(Console.ReadLine());

int b = int.Parse(Console.ReadLine());

Swap(a,b);

void Swap (int a,int b)

{

    Console.WriteLine("El numero A es : " +a);

    Console.WriteLine("El numero b es : " +b);

    int item=a;

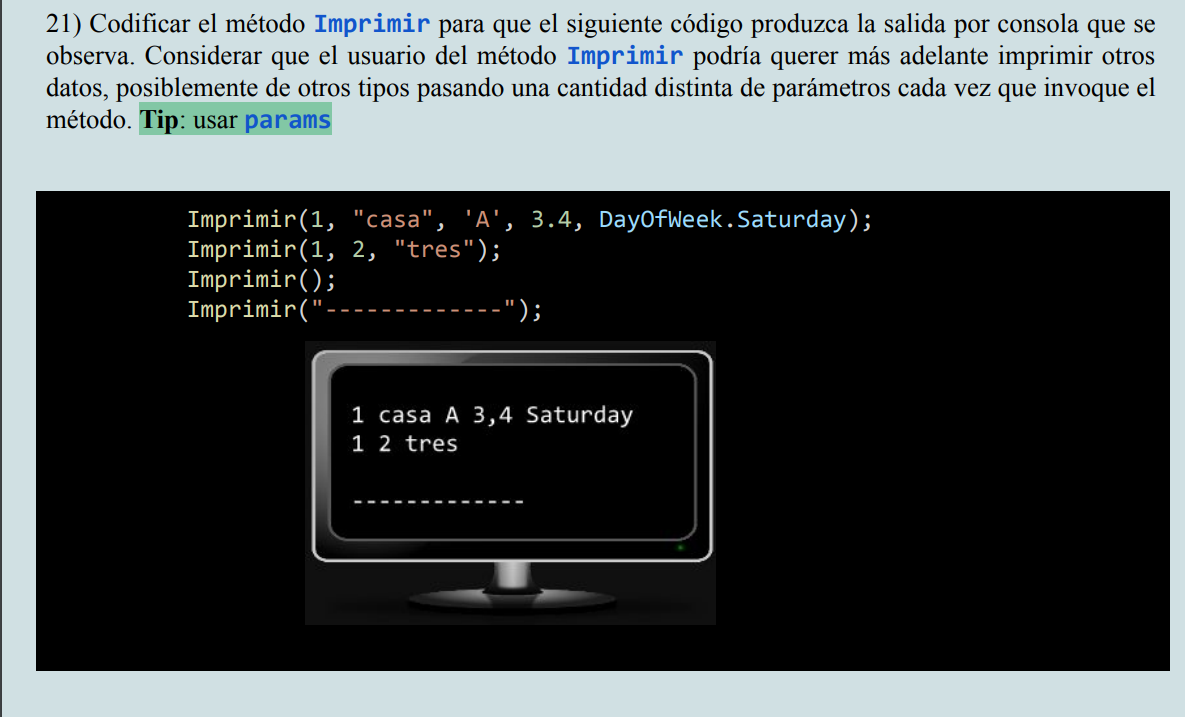
    a=b;

    b=item;

    Console.WriteLine("El numero A es : " +a);

    Console.WriteLine("El numero b es : " +b);

}



Imprimir(1,3,5,"HOLA MUNDO",Tamaño.chico);

void  Imprimir(params object[]vec)

{

    foreach(object o in vec)

    {

       Console.WriteLine(o.ToString());

    }

}