

FACTORIAL

I. ANÁLISIS

Descripción del problema: Dado un número entero y positivo, este código calcula el factorial de dicho número.

Datos de entrada: numeroUsuario

Datos de salida: resultado

II. PSEUDOCÓDIGO

INICIO

numeroUsuario: ENTERO

resultado: ENTERO

ESCRIBIR "Dame un número"

LEER numeroUsuario

resultado:=factorial(numeroUsuario)

ESCRIBIR "El resultado es " resultado

FIN

INICIO

FUNC factorial (numeroFactorial:ENTERO) RET: ENTERO

res:=1: ENTERO

SI numeroFactorial = 0 ENTONCES

RET 1

FIN SI

DE LO CONTRARIO

MIENTRAS (numeroFactorial>0)

res:= res*numeroFactorial

numeroFactorial:= numeroFactorial-1

FIN MIENTRAS

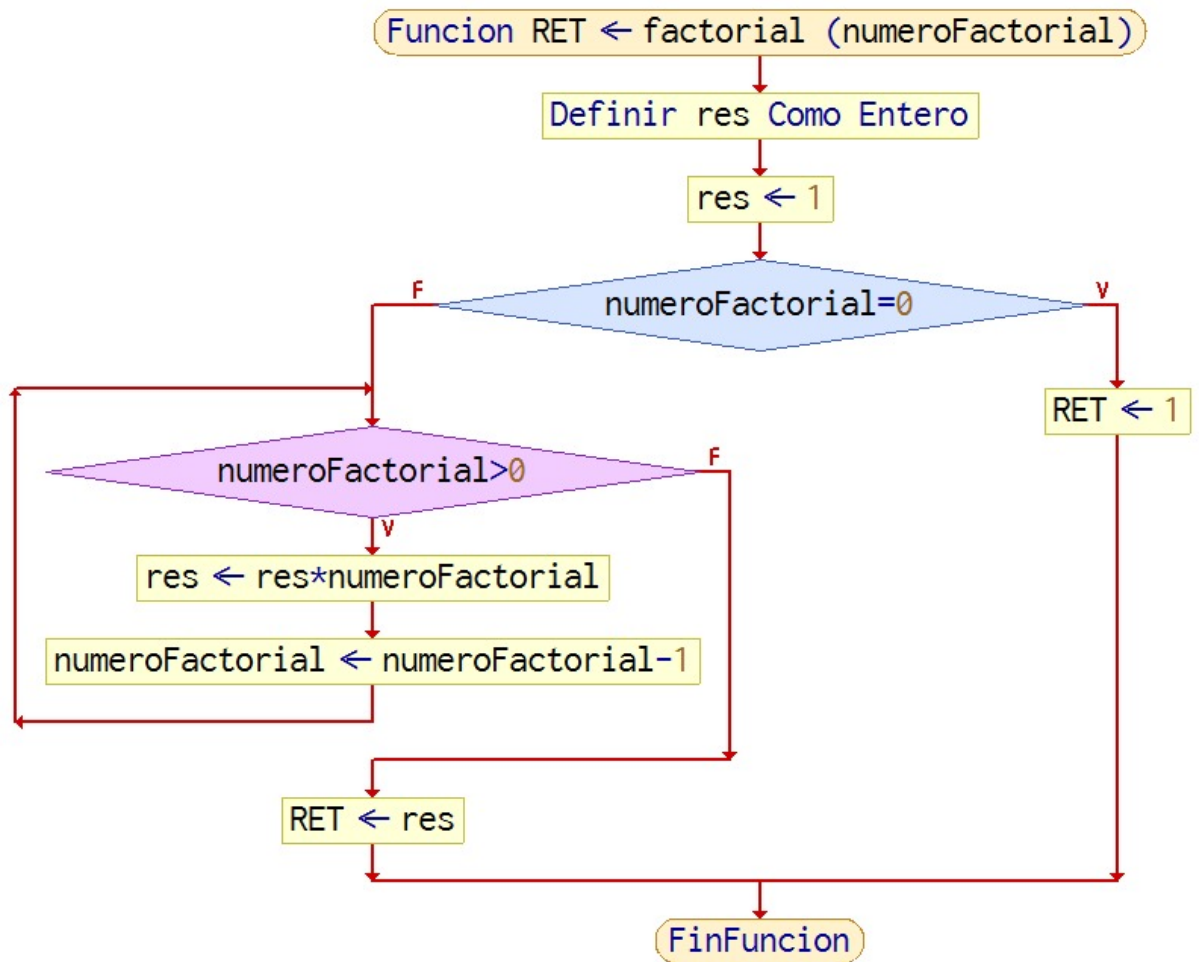
RET res

FIN DE LO CONTRARIO

FIN FUNC

FIN

DIAGRAMA DE FLUJO



III. PRUEBA DE ESCRITORIO

Variables	Valor de la variable				Salida
numeroUsuario	3				El resultado del factorial es: 6
resultado	6				
res	1	3	6	6	
numeroFactorial	3	2	1	0	

```

Dame un numero
3
El resultado del Factorial es: 6

```

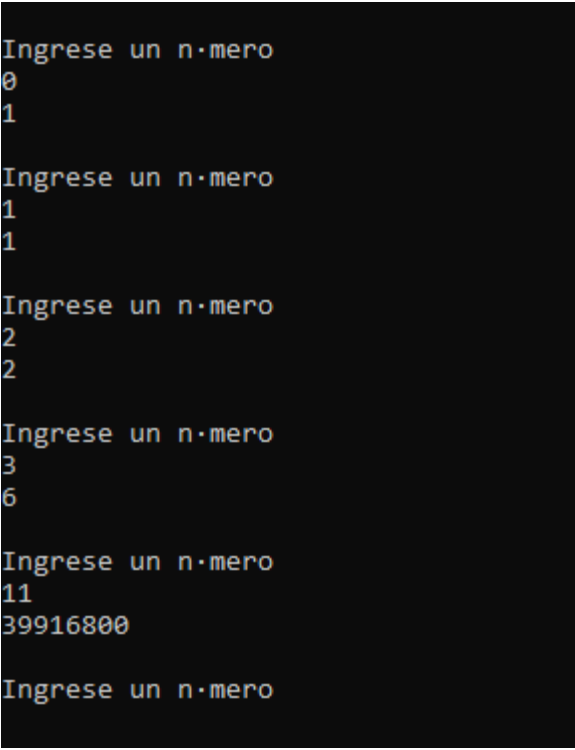
IV. CODIFICACIÓN

```
#include <stdio.h>

long calculoFactorial(int n)
{
    int res;
    res=1;

    if(n==0){
        return res;
    }
    else{
        while(n>0){
            res= res*n;
            n= n-1;
        }
        return res;
    }
}
```

:0



```
Ingrese un n-mero
0
1
Ingrese un n-mero
1
1
Ingrese un n-mero
2
2
Ingrese un n-mero
3
6
Ingrese un n-mero
11
39916800
Ingrese un n-mero
```

CUENTA CARACTERES

I. ANÁLISIS

Descripción del problema: El siguiente programa cuenta las veces que se repite un carácter en una cadena e imprime su código ascii y las repeticiones contadas de cada uno.

Datos de entrada: cadena

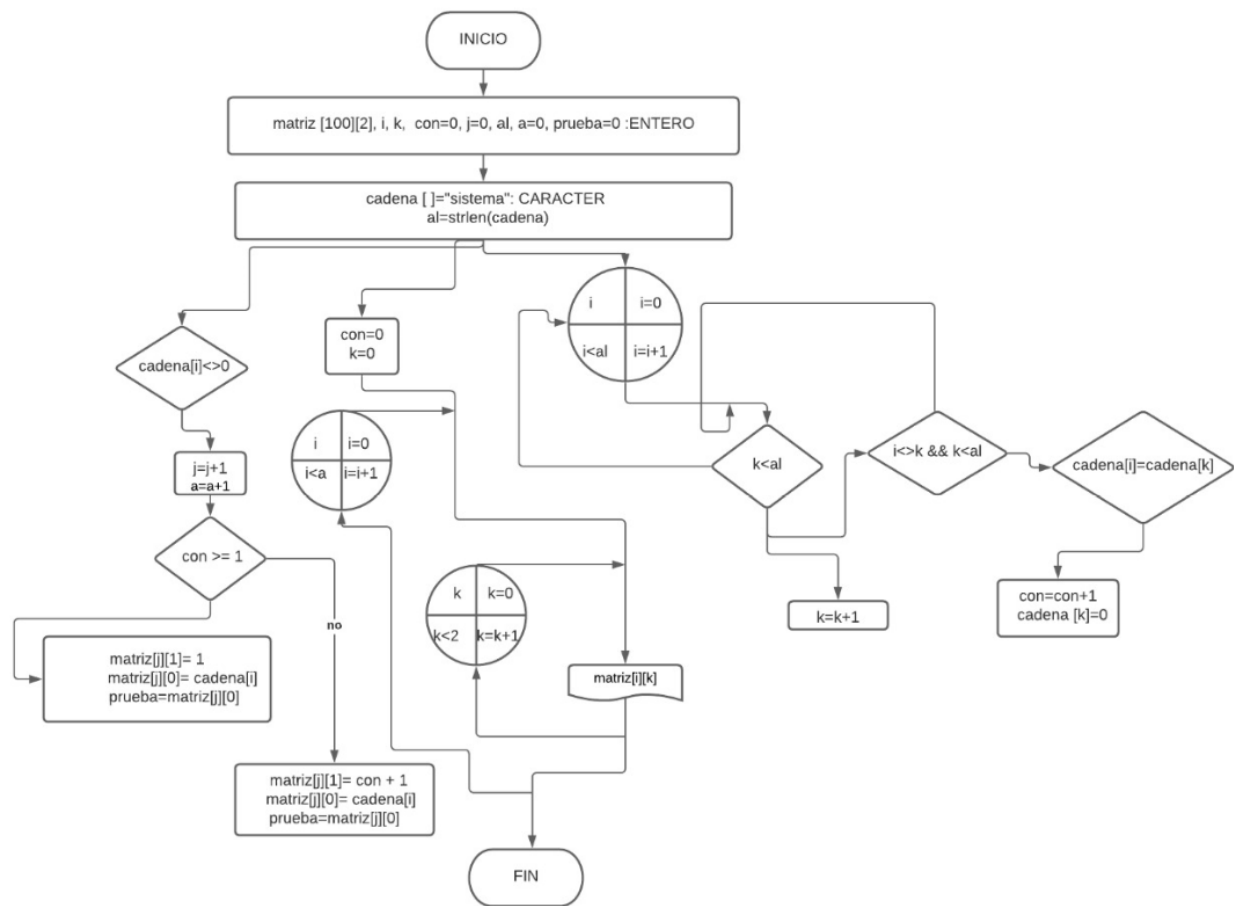
Datos de salida: arreglo[i][k]

II. PSEUDOCÓDIGO

```
INICIO
  FUNC contar(cadena:CADENA, arreglo[ ][2]:ENTERO) RET:
vacío
  i, con:=0, j:=0, al, a:=0, prueba:=0, k:=0:ENTERO
  ESCRIBIR "Introduce una palabra"
  LEER cadena
  al:= STRLEN(cadena)
  i:=0
  MIENTRAS i < al
    MIENTRAS k < al
      SI i <> k ENTONCES
        SI cadena[i] = cadena[k] ENTONCES
          con:= con+1
          cadena[k]:= 0
        FIN SI
      FIN SI
      k:=k+1
    FIN MIENTRAS
    SI cadena[i] <> 0 ENTONCES
      SI con >= 1 ENTONCES
        arreglo[j][1]:=con+1
        arreglo[j][0]:= cadena[i]
        prueba:=arreglo[j][0]
      FIN SI
      DE LO CONTRARIO
        arreglo[j][1]:=1
        arreglo[j][0]:=CADENA[i]
```

```
                prueba:=arreglo[j][0]
            FIN DE LO CONTRARIO
            j:=j+1
            a:=a+1
        FIN SI
        con:=0
        i:=i+1
    FIN MIENTRAS
    i:=0
    MIENTRAS i<a
    k:=0
        MIENTRAS k<2
            ESCRIBIR arreglo[i][k]
            k+1
        FIN MIENTRAS
        i+1
    FIN MIENTRAS
FIN FUNC
FIN
```

DIAGRAMA DE FLUJO



III. PRUEBA DE ESCRITORIO

Cadena que escribe el Usuario	Resultado en pantalla
alma	97, 2 108, 1 109, 1
murcielago	109, 1 117, 1 114, 1 99, 1 105, 1 101, 1 108, 1 97, 1

	103, 1 111, 1
arroz	97, 1 114, 2 111, 1 122, 1
papaya	112, 2 97, 3 121, 1

```
Introduce una palabra
alma
97,2,
108,1,
109,1,
```

```
Introduce una palabra
murcielago
109,1,
117,1,
114,1,
99,1,
105,1,
101,1,
108,1,
97,1,
103,1,
111,1,
```

```
Introduce una palabra
arroz
97,1,
114,2,
111,1,
122,1,
```

```
Introduce una palabra
papaya
112,2,
97,3,
121,1,
```

IV. CODIFICACIÓN

```
#include <stdio.h>
#include <string.h>

void contar(const char * cadena, int arreglo[][2])
{
    int i, con=0, j=0, al, a=0, prueba=0;
    int k=0;
    char* s;
    printf("Introduce una palabra \n");
    cadena = malloc (256);
```

```

scanf("%255s", cadena );
s = strdup(cadena);
al=strlen(cadena);
for (i=0; i<al; i++)
{
    while (k<al)
    {
        if(i!=k && k<al)
        {
            if (s[i]==s[k])
            {
                con=con+1;
                s[k]=0;
            }
        }
        k++;
    }
    if (s[i]!=0)
    {
        if (con >= 1)
        {
            arreglo[j][1]= con + 1;
            arreglo[j][0]= s[i];
            prueba=arreglo[j][0];
        }
        else
        {
            arreglo[j][1]= 1;
            arreglo[j][0]= s[i];
            prueba=arreglo[j][0];
        }
        j++;
        a=a+1;
    }
    con=0;
    k=0;
}
for (i=0 ; i<a ; i++)
{
    for (k=0 ; k<2 ; k++)
    {

```



```
        printf("%ld,", arreglo[i][k]);  
    }  
    printf("\n");  
}  
free(cadena);  
free(s);  
}
```

```
/*int main(){  
    char * cadenita;  
    int a2[1000][2];  
    contar(cadenita, a2);  
}*/
```