

Problema 01

```

1. Inicio
2. Funcion principal ( )
3. nombres , x(100) [ { A-Z } , { a-z } , { 0-9 } ]
4. librusCumaron , d [ 0-n ]
5. librusBaculao , d [ 0-n ]
6. Mientras ( true )
7.   << "Bienvenido a la pesqueria de la ciudad "
8.   << "Ingrese su nombre o " salir " para terminar "
9.   >> nombres
10.  si ( nombres == salir ) entonces
11.    salir del bucle
12.  Fin si
13.   << "Ingrese el numero de librus cumaron : "
14.   >> librusCumaron
15.   << "Ingrese el numero de librus Baculao : "
16.   >> librusBaculao
17.   costoTotal , d [ 0-n ]
18.   costoTotal <- calculoCostoTotal ( librusCumaron ,
19.                                     librusBaculao ) ;
20.   << "Pesador : " + nombres
21.   << "Librus de Cumaron : " + librusCumaron + " - costo $ "
22.     + ( librusCumaron * 1.1 ) ;
23.   << "Librus de Baculao : " + librusBaculao + " - costo $ "
24.     + ( librusBaculao * 1.2 ) ;
25.   << "Total a pagar al pesador : $ " costoTotal
26.  Fin Mientras
27. Fin Funcion principal
28. Funcion costoTotal ( librusCumaron , i ; librusBaculao , i ) :
29.   costoCumaron , i [ 0-n ]
30.   costoBaculao , i [ 0-n ]
31.   costoCumaron <- librusCumaron * 1.1
32.   costoBaculao <- librusBaculao * 1.2
33.   retorno ( costoCumaron + costoBaculao )
34. Fin Funcion costoTotal
35. Fin

```

Algoritmo.

- Ingresamos datos por teclado, nombre, libras A, libras B.
- Cada libra equivale a un precio.
- Finalmente se debe pagar al pesador con la suma total de libras tanto cumaron y baculao.