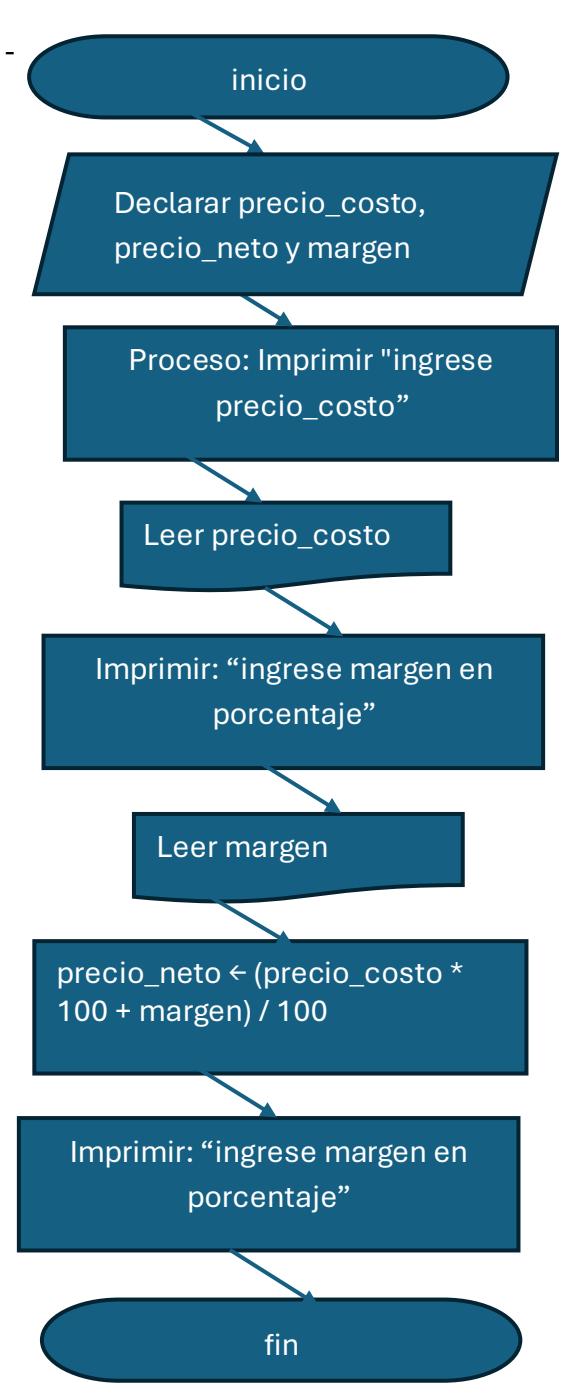


<p>Inicio</p> <p> Imprimir Ciberseguridad</p> <p>Fin</p>	<pre> graph TD inicio([inicio]) --> proceso[Proceso: Imprimir "Ciberseguridad"] proceso --> fin([fin]) </pre>
<p>Inicio</p> <p> Declarar nombre</p> <p> Imprimir Hola, ¿cómo te llamas?</p> <p> Ler nombre</p> <p> Imprimir Mucho gusto + nombre</p> <p>Fin</p>	<pre> graph TD inicio([inicio]) --> declarar[/Declarar nombre/] declarar --> proceso[Proceso: Imprimir "Hola ¿Cómo te llamas?"] proceso --> leer[Leer nombre] leer --> imprimir[Imprimir "mucho gusto nombre"] imprimir --> fin([fin]) </pre>

```

Inicio
    Declarar variable precio_costo,
    precio_neto y margen
    Imprimir Ingrese precio de costo
    Leer precio_costo
    Imprimir Ingrese margen en porcentaje
    Leer margen
    precio_neto ← (precio_costo * 100 +
    margen) / 100
    Imprimir El precio neto es: +
    precio_neto
Fin

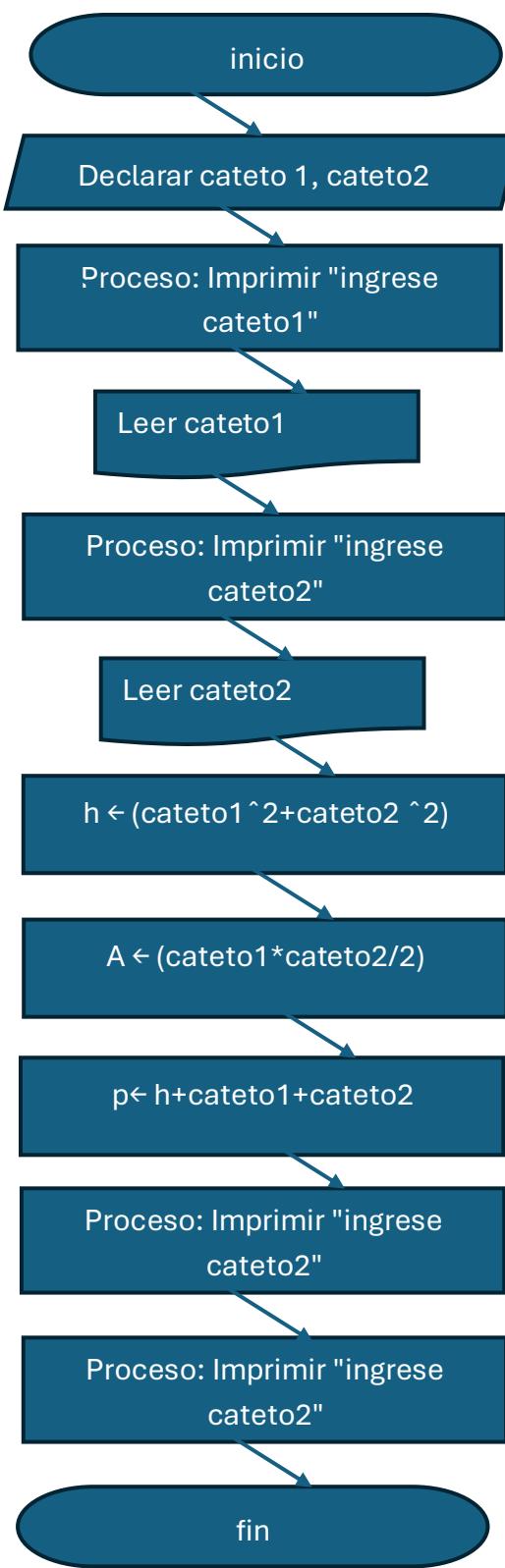
```



```

Inicio
    Declarar variables cateto1, cateto2.
    Imprimir Ingrese cateto1
    Leer cateto1
    Imprimir Ingrese cateto2
    Leer cateto2
    h ← (cateto1^2+cateto2 ^2)
    A ← (cateto1*cateto2/2)
    p← h+cateto1+cateto2
    Imprimir Hipotenusa: + h
    Imprimir Área: + A
    Imprimir Perímetro: + p
Fin

```



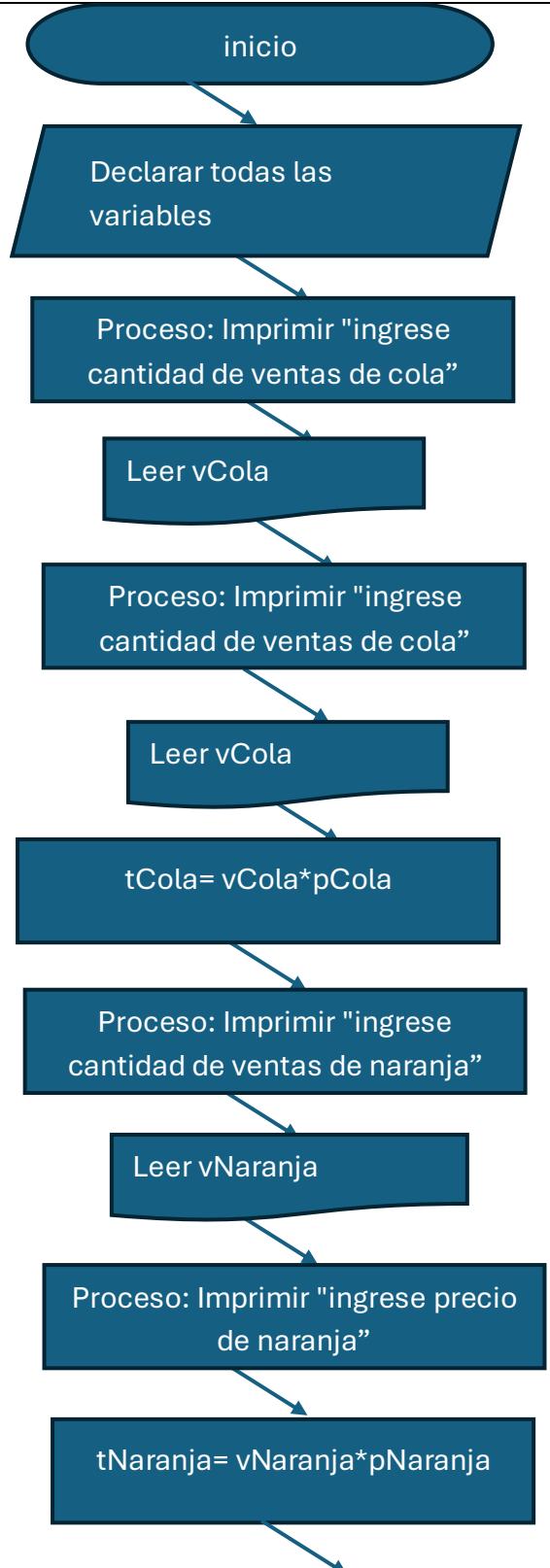
```

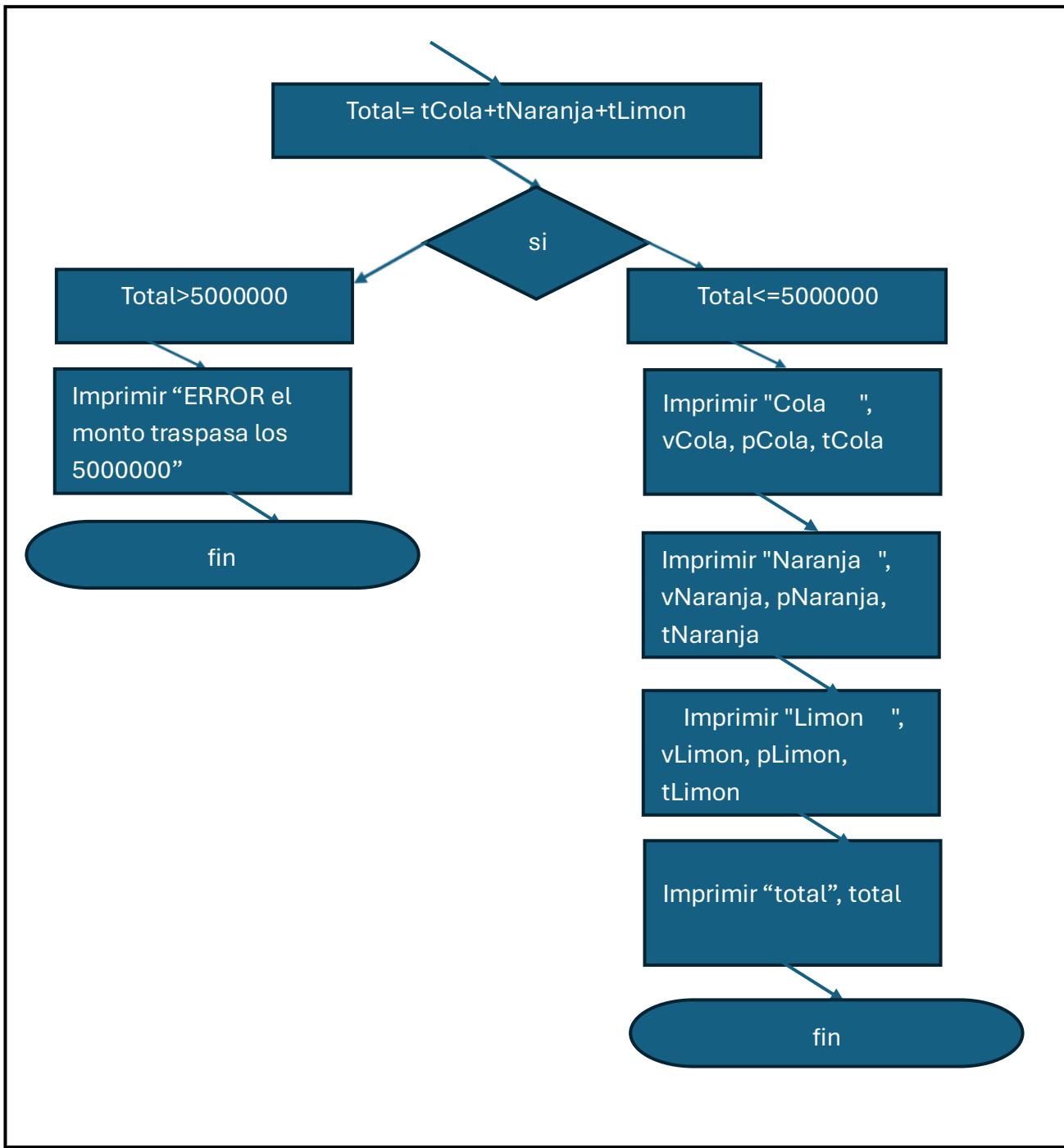
Inicio
Declarar vCola, tCola, pCola, vNaranja,
tNaranja, pNaranja, vLimon, pLimon,
tLimon
    Imprimir Ingrese cantidad de ventas
    de Cola
        Leer vCola
        Imprimir Ingrese precio de Cola
        Leer pCola
        tCola ← vCola * pCola

    Imprimir Ingrese cantidad de ventas
    de Naranja
        Leer vNaranja
        Imprimir Ingrese precio de Naranja
        Leer pNaranja
        tNaranja ← vNaranja * pNaranja

    Imprimir Ingrese cantidad de ventas
    de Limón
        Leer vLimon
        Imprimir Ingrese precio de Limón
        Leer pLimon
        tLimon ← vLimon * pLimon
        total ← tCola + tNaranja + tLimon
        si total<=5000000
            Imprimir Producto  Ventas  Precio
            Total
                Imprimir Cola , vCola, pCola, tCola
                Imprimir Naranja , vNaranja,
                pNaranja, tNaranja
                Imprimir Limon , vLimon, pLimon,
                tLimon
                Imprimir TOTAL , total
            Si total >5000000
                Imprimir Error se paso del monto
                maximo
            Fin

```





PRIMER PROBLEMA

<pre> 1 // Online C compiler to run C program online 2 #include <stdio.h> 3 4 int main() { 5 printf ("Ciberseguridad"); 6 7 return 0; 8 }</pre>	Ciberseguridad ==== Code Execution Successful ===
---	--

SEGUNDO PROBLEMA

```
// Online C compiler to run C program online
#include <stdio.h>

int main() {
    char nombre[40];

    printf("Hola, ¿cómo te llamas? ");
    scanf("%s", nombre);
    printf("Mucho gusto, %s \n", nombre);

    return 0;
}
```

Hola, ¿cómo te llamas? fernando castillo
Mucho gusto, fernando

==== Code Execution Successful ===

TERCER PROBLEMA

```
// Online C compiler to run C program online
#include <stdio.h>

int main() {
    float precio_costo, margen, precio_neto;

    printf("Ingrese precio de costo (balboas): ");
    scanf("%f", &precio_costo);

    printf("Ingrese margen en porcentaje: ");
    scanf("%f", &margen);

    precio_neto = (precio_costo * 100 + margen) / 100;

    printf("El precio neto es: %f balboas\n", precio_neto);

    return 0;
}
```

Ingrese precio de costo (balboas): 100
Ingrese margen en porcentaje: 20
El precio neto es: 100.199997 balboas

==== Code Execution Successful ===

CUARTO PROBLEMA

```

int main() {
    float c1, c2, h, area, perimetro;

    printf("Ingrese el valor del primer cateto: ");
    scanf("%f", &c1);

    printf("Ingrese el valor del segundo cateto: ");
    scanf("%f", &c2);

    h = ((c1 * c1) + (c2 * c2));
    area = (c1 * c2) / 2;
    perimetro = h + c1 + c2;

    printf("Hipotenusa: %.2f\n", h);
    printf("Área: %.2f\n", area);
    printf("Perímetro: %.2f\n", perimetro);

    return 0;
}

```

Ingrese el valor del primer cateto: 4
 Ingrese el valor del segundo cateto: 5
 Hipotenusa: 41.00
 Área: 10.00
 Perímetro: 50.00
 === Code Execution Successful ===

QUINTO PROBLEMA

```

// Online C compiler to run C program online
#include <stdio.h>

int main() {
    float vCola, vNaranja, vLimon;
    float pCola, pNaranja, pLimon;
    float tCola, tNaranja, tLimon, total;

    printf("Ingrese cantidad de ventas de Cola: ");
    scanf("%f", &vCola);
    printf("Ingrese precio de Cola: ");
    scanf("%f", &pCola);

    printf("Ingrese cantidad de ventas de Naranja: ");
    scanf("%f", &vNaranja);
    printf("Ingrese precio de Naranja: ");
    scanf("%f", &pNaranja);

    printf("Ingrese cantidad de ventas de Limón: ");
    scanf("%f", &vLimon);
    printf("Ingrese precio de Limón: ");
    scanf("%f", &pLimon);

    tCola = vCola * pCola;
    tNaranja = vNaranja * pNaranja;
    tLimon = vLimon * pLimon;
    total = tCola + tNaranja + tLimon;

    printf("\n");
    printf("Producto      Ventas      Precio      Total\n");
    printf("Cola          %f          %f          %f\n", vCola, pCola, tCola);
    printf("Naranja       %f          %f          %f\n", vNaranja, pNaranja, tNaranja);
    printf("Limon         %f          %f          %f\n", vLimon, pLimon, tLimon);
    printf("TOTAL          %f          %f          %f\n", total);

    return 0;
}

```

Ingrese cantidad de ventas de Cola: 10
 Ingrese precio de Cola: 10
 Ingrese cantidad de ventas de Naranja: 10
 Ingrese precio de Naranja: 10
 Ingrese cantidad de ventas de Limón: 10
 Ingrese precio de Limón: 10
 Producto Ventas Precio Total
 Cola 10.000000 10.000000 100.000000
 Naranja 10.000000 10.000000 100.000000
 Limon 10.000000 10.000000 100.000000
 TOTAL 300.000000
 === Code Execution Successful ===

Ingrese cantidad de ventas de Cola: 10
 Ingrese precio de Cola: 10
 Ingrese cantidad de ventas de Naranja: 10
 Ingrese precio de Naranja: 10
 Ingrese cantidad de ventas de Limón: 10
 Ingrese precio de Limón: 10
 Producto Ventas Precio Total
 Cola 10.000000 10.000000 100.000000
 Naranja 10.000000 10.000000 100.000000
 Limon 10.000000 10.000000 100.000000
 TOTAL 300.000000
 === Code Execution Successful ===