

Estructuras de selección

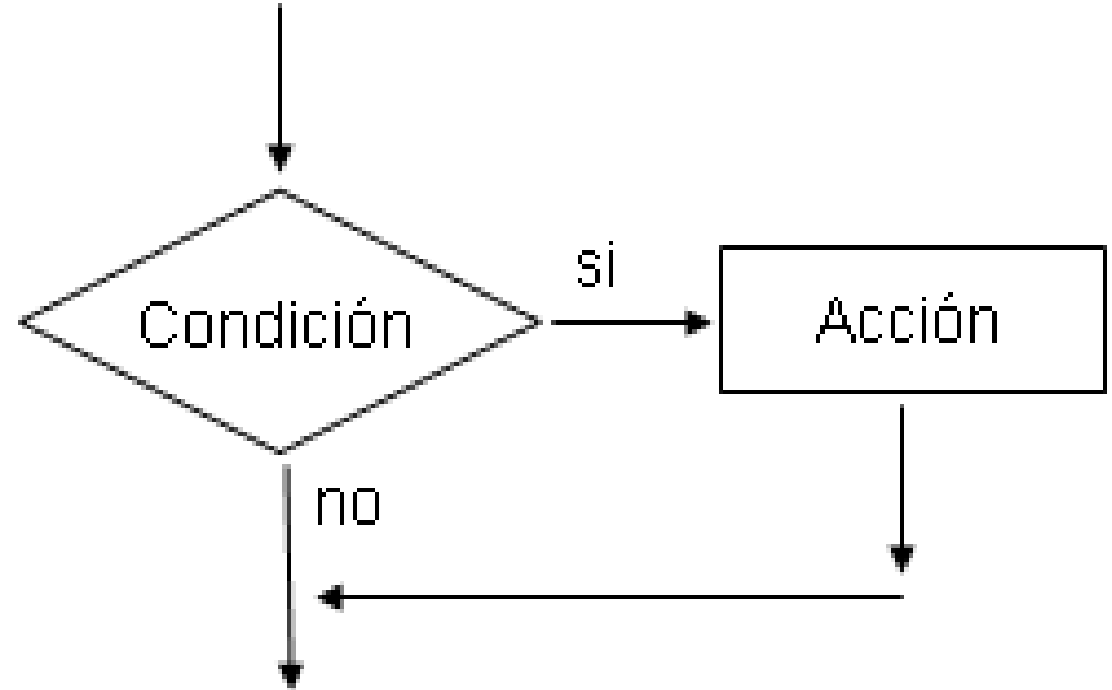
Estructuras de control de flujo

Estructuras selectivas – Condicional simple

- Condicional simple (if/end)

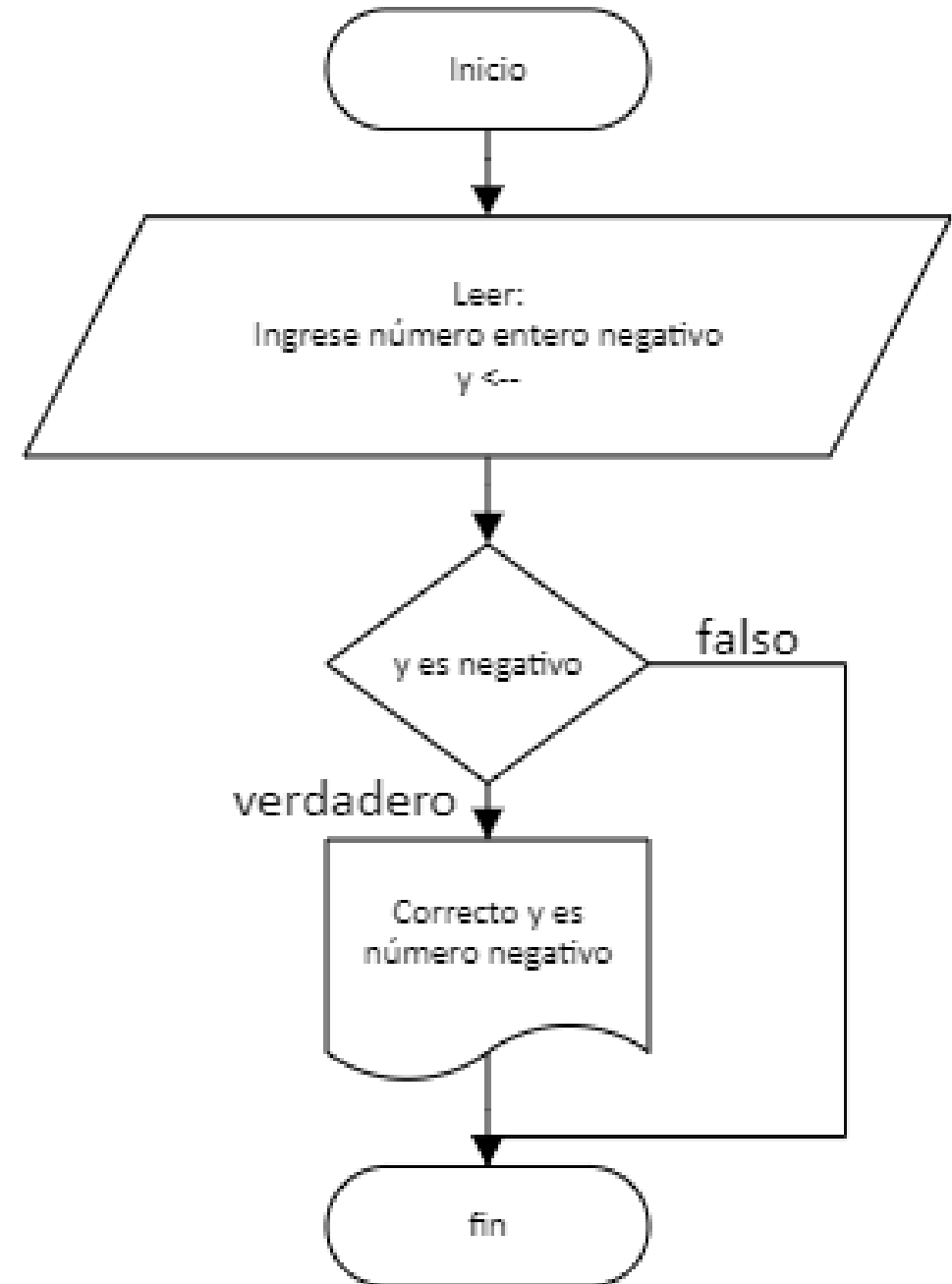
- Para C++:

```
if (condición)
{
    Acción
}
```



Ejemplo 1 – Condicional Simple

- Haga un programa que le solicite al usuario un número entero negativo.
- Su programa debe informar que efectivamente el número ingresado es negativo.

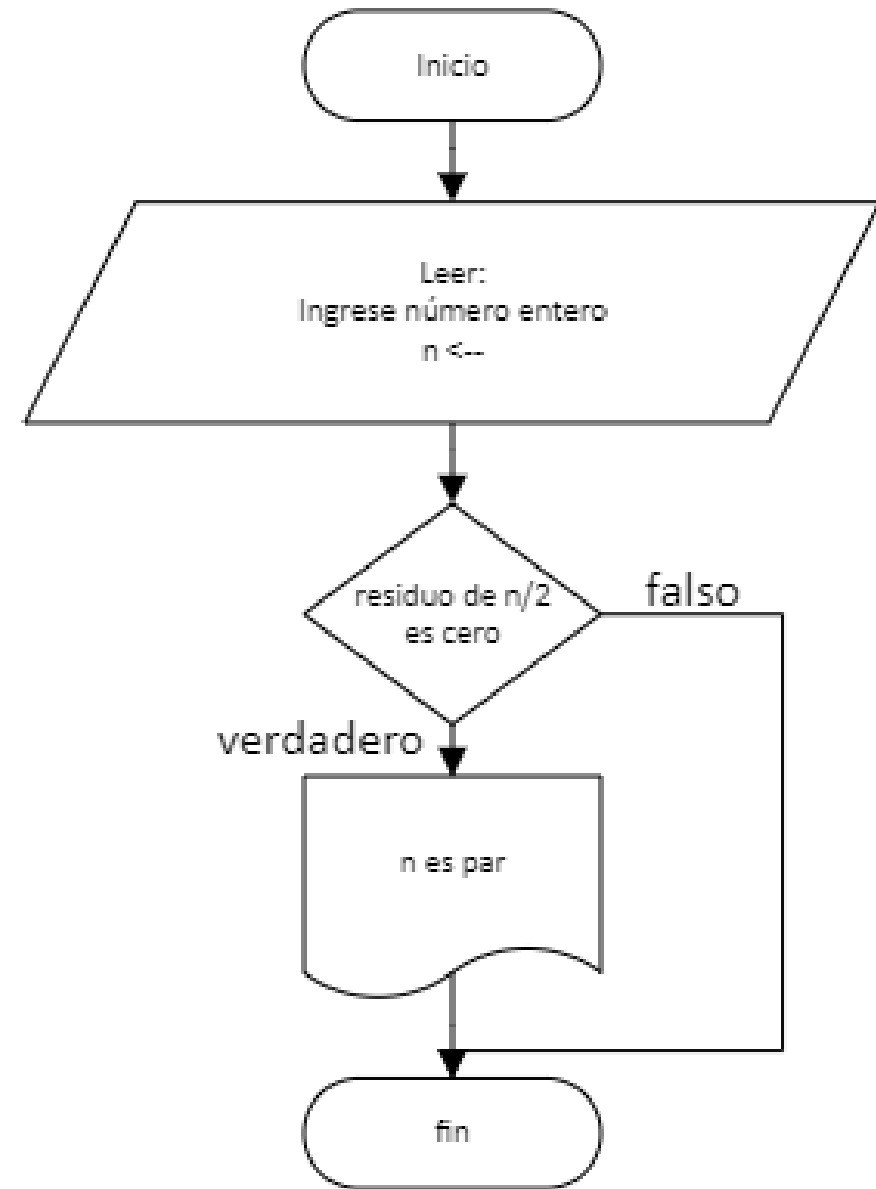


Ejemplo 1 – Condicional simple - Código

```
#include <iostream>
// Ejemplo de condición simple
int main()
{
    int y;
    std :: cout << "Ingrese un numero negativo: ";
    std :: cin >> y;
    if (y<0)
    {
        std :: cout << "Correcto " << y << "es numero negativo\n";
    }
    return 0;
}
```

Ejemplo 2 – Condicional simple

- Haga un programa que indique si un número entero es par



Ejemplo 2 – condicional simple – número par

```
#include <iostream>
```

```
int main()
```

```
{
```

```
    int n;
```

```
    std :: cout << "Ingrese un numero entero: ";
```

```
    std :: cin >> n;
```

```
    if (n % 2 == 0)
```

```
    {
```

```
        std :: cout << "Correcto " << n << " es numero par\n";
```

```
    }
```

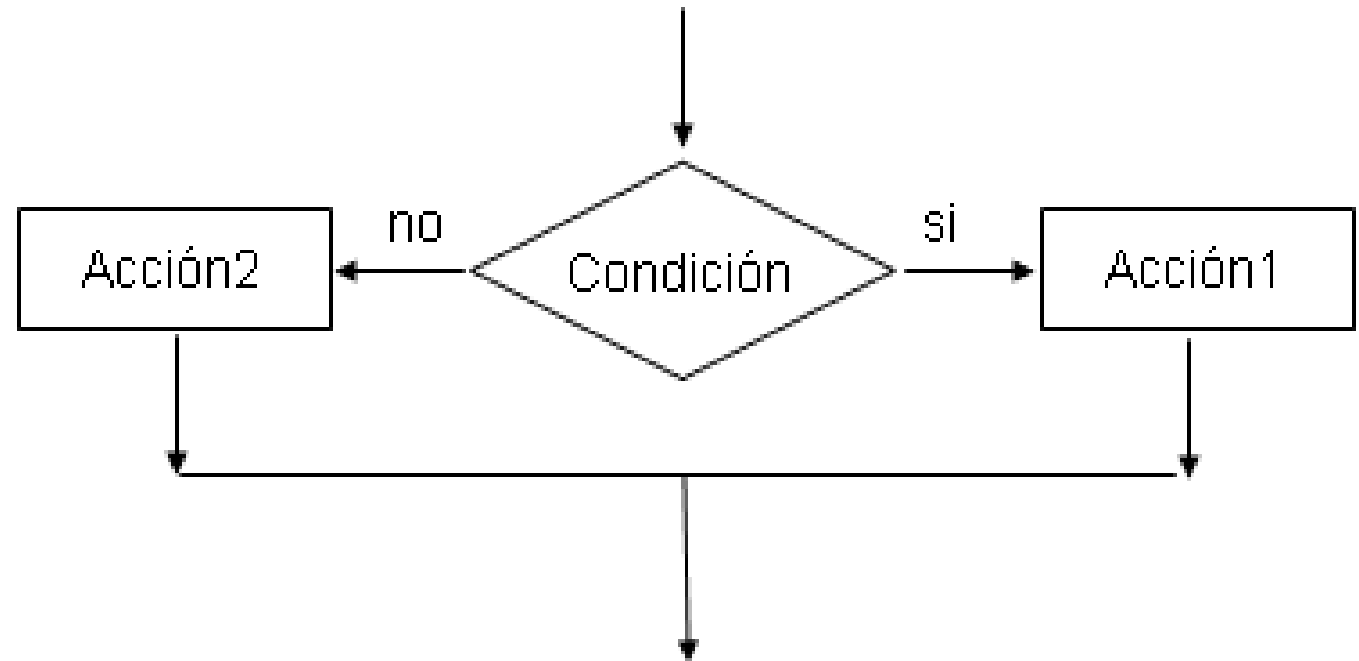
```
    return 0;
```

```
}
```

Estructuras Selectivas – conditional if / else

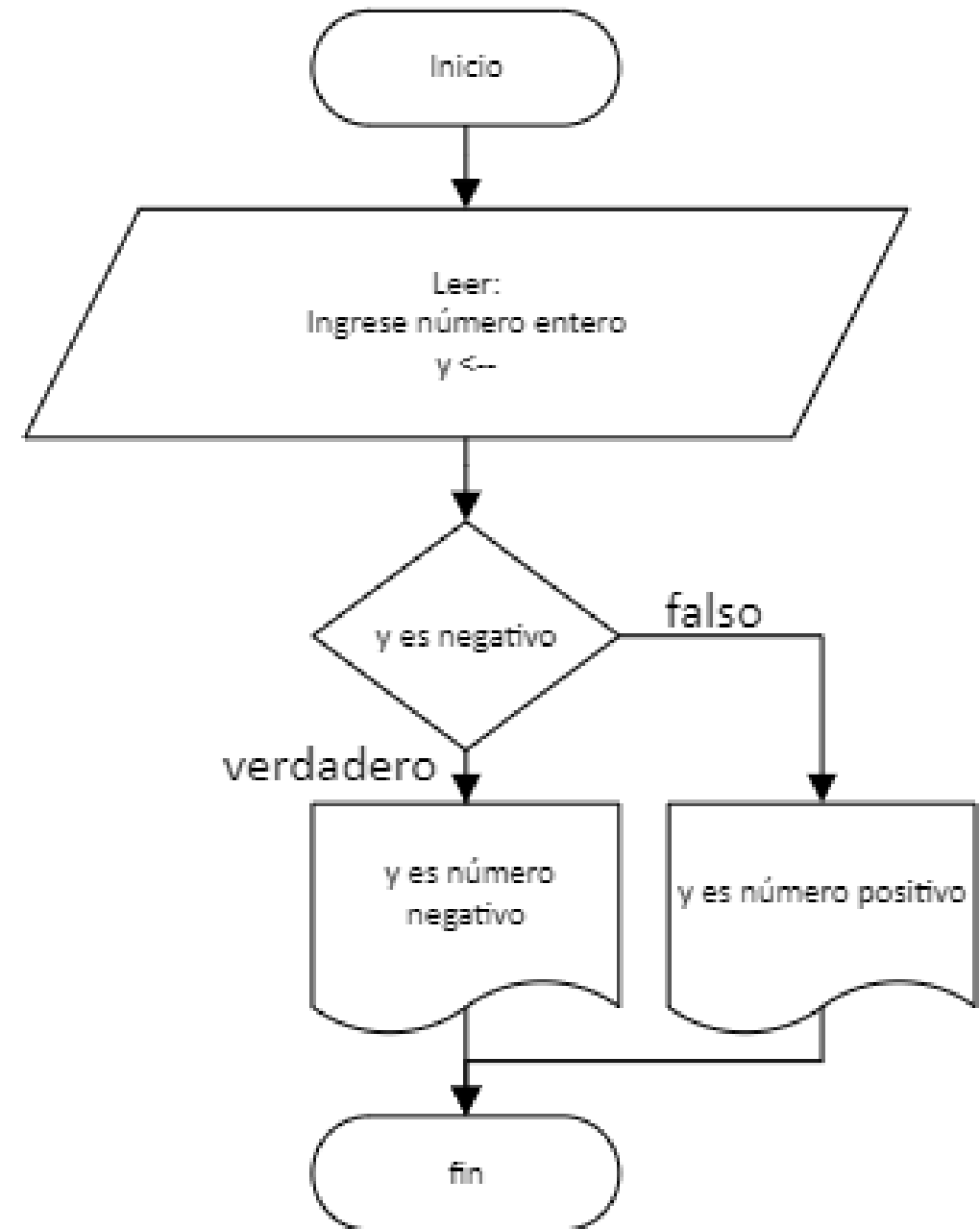
- Condicional if/else

```
if (condición)
{
    acción1
}
else
{
    acción2
}
```



Ejemplo 3 – Condicional if/else

- Haga un programa que le solicite al usuario un número entero.
- Su programa debe informar si el número ingresado es negativo o positivo.

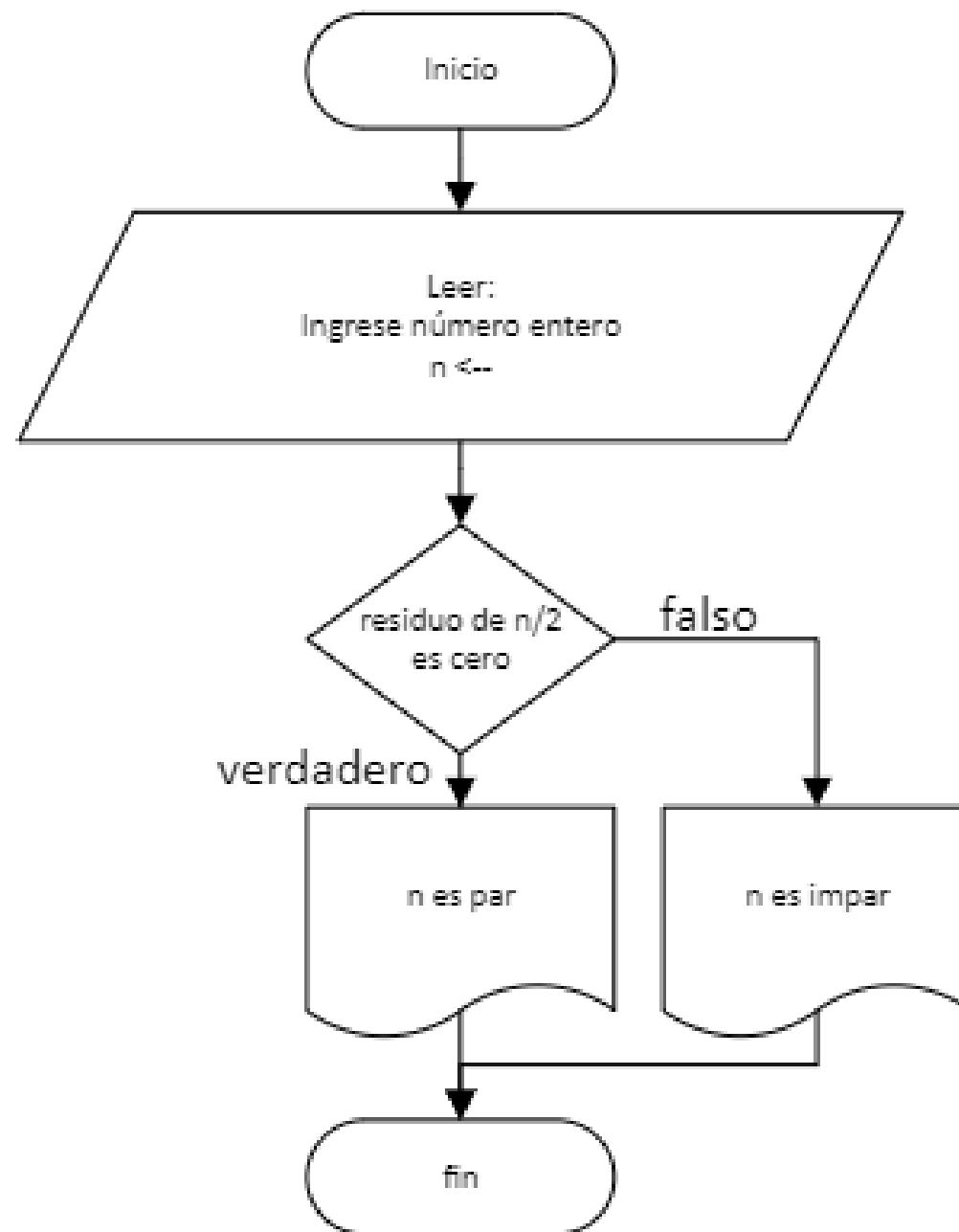


Ejemplo 3 – Condicional if/else - Código

```
#include <iostream>
// Ejemplo de condición doble
int main()
{
    int y;
    std :: cout << "Ingrese un numero entero: ";
    std :: cin >> y;
    if (y<0)
    {
        std :: cout << y << " es numero negativo\n";
    }
    else
    {
        std :: cout << y << " es numero positivo";
    }
    return 0;
}
```

Ejemplo 4 – Condicional if/else

- Haga un programa que indique si un número entero es par o impar.



Ejemplo 2 – condicional if/else – número par/impar

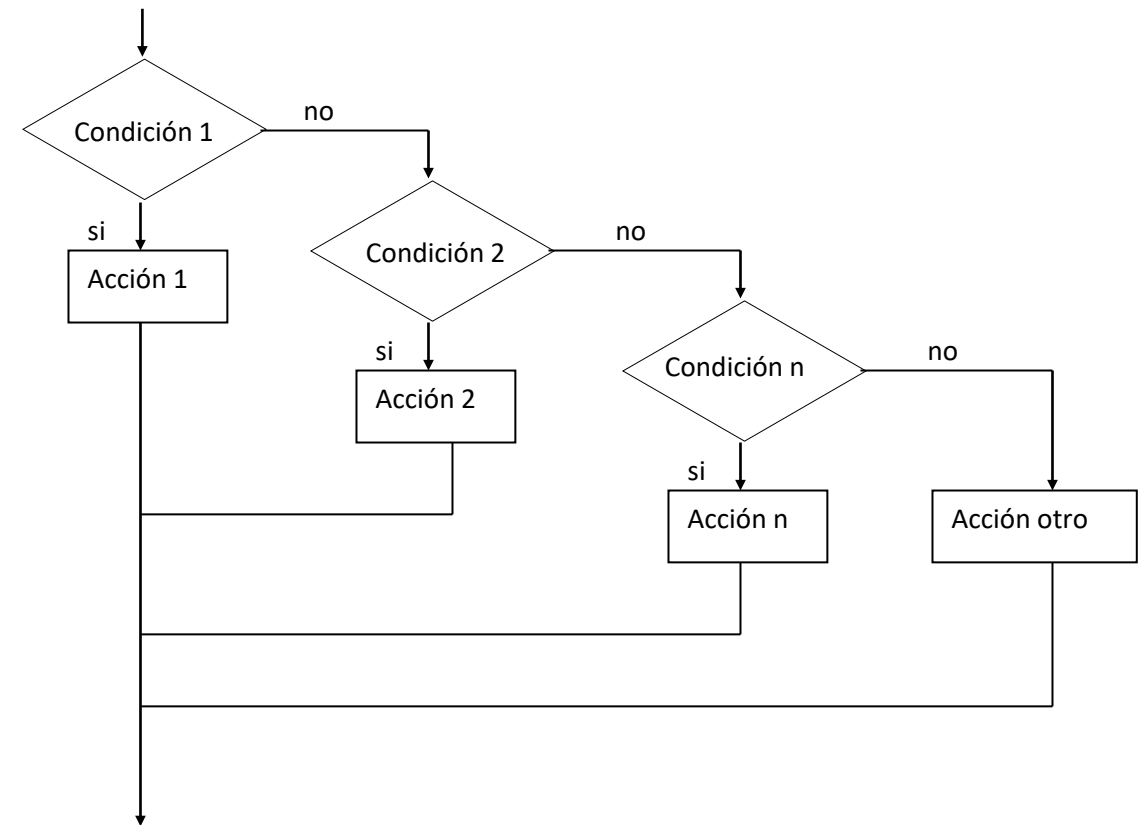
```
#include <iostream>

int main()
{
    int n;
    std :: cout << "Ingrese un numero entero: ";
    std :: cin >> n;
    if (n % 2 == 0)
    {
        std :: cout << n << " es numero par\n";
    }
    else
    {
        std :: cout << n << " es numero impar\n";
    }
    return 0;
}
```

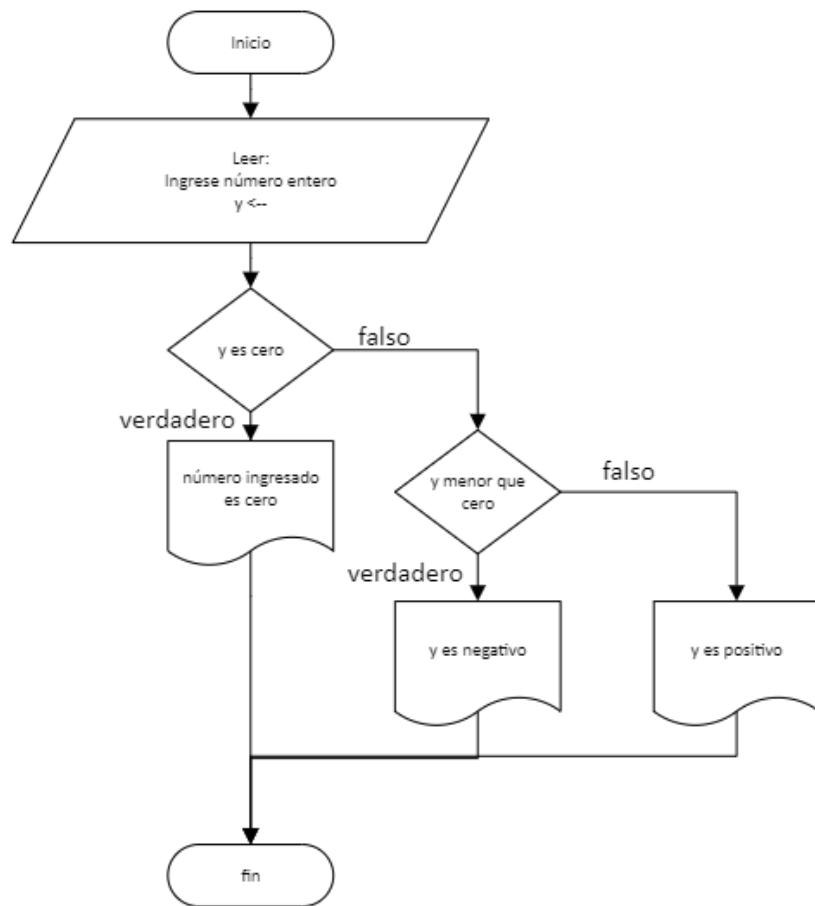
Estructuras Selectivas – if/elseif

Selección Múltiple

```
if (condición 1)
{
    accion 1
}
else if (condición 2)
{
    accion 2
}
elseif (condición n)
{
    accion n
}
else
{
    otra acción
}
```



Ejemplo 5 – Condición if/elseif



- Haga un programa que diga si un número entero es positivo, negativo o cero.

Ejemplo 5 – condición if/elseif – negativo/positivo/cero

```
#include <iostream>

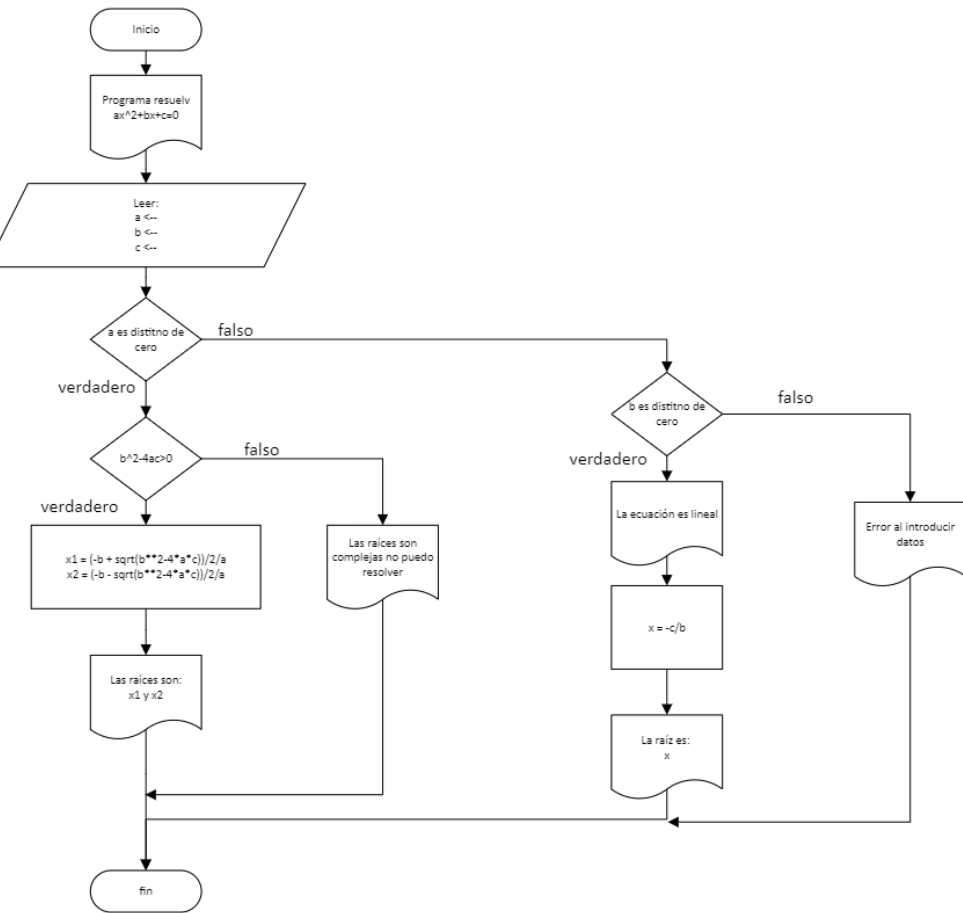
int main()
{
    int y;
    std :: cout << "ingrese un numero entero: ";
    std :: cin >> y;

    if (y == 0)
    {
        std :: cout << "numero ingresado es cero\n";
    }
    else if (y < 0)
    {
        std :: cout << y << " es negativo\n";
    }
    else
    {
        std :: cout << y << " es positivo\n";
    }
    return 0;
}
```

Ejemplo 6 - condicionales

- Haga un programa para encontrar las raíces de una ecuación de segundo orden. La solución debe considerar lo siguiente:
- Entradas: a, b, c
- Salidas, raíces de la ecuación.
- Proceso:
 - Debe considerar el caso general de una cuadrática, recuerde que las raíces se calculan de la siguiente manera:
$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \text{ y } x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$
 - Si $b^2 - 4ac < 0$ debe enviar un mensaje al usuario y decirle que no puede realizarse el cálculo debido a que hay raíces complejas conjugadas.
 - Si $a = 0$, debe enviar un mensaje notificando que la ecuación no es cuadrática, que es una ecuación lineal. Pero si informará el valor de la raíz, recuerde $x = -c/b$

Ejemplo 6 – if/else – cuadrática



```
#include <iostream>
#include <cmath>

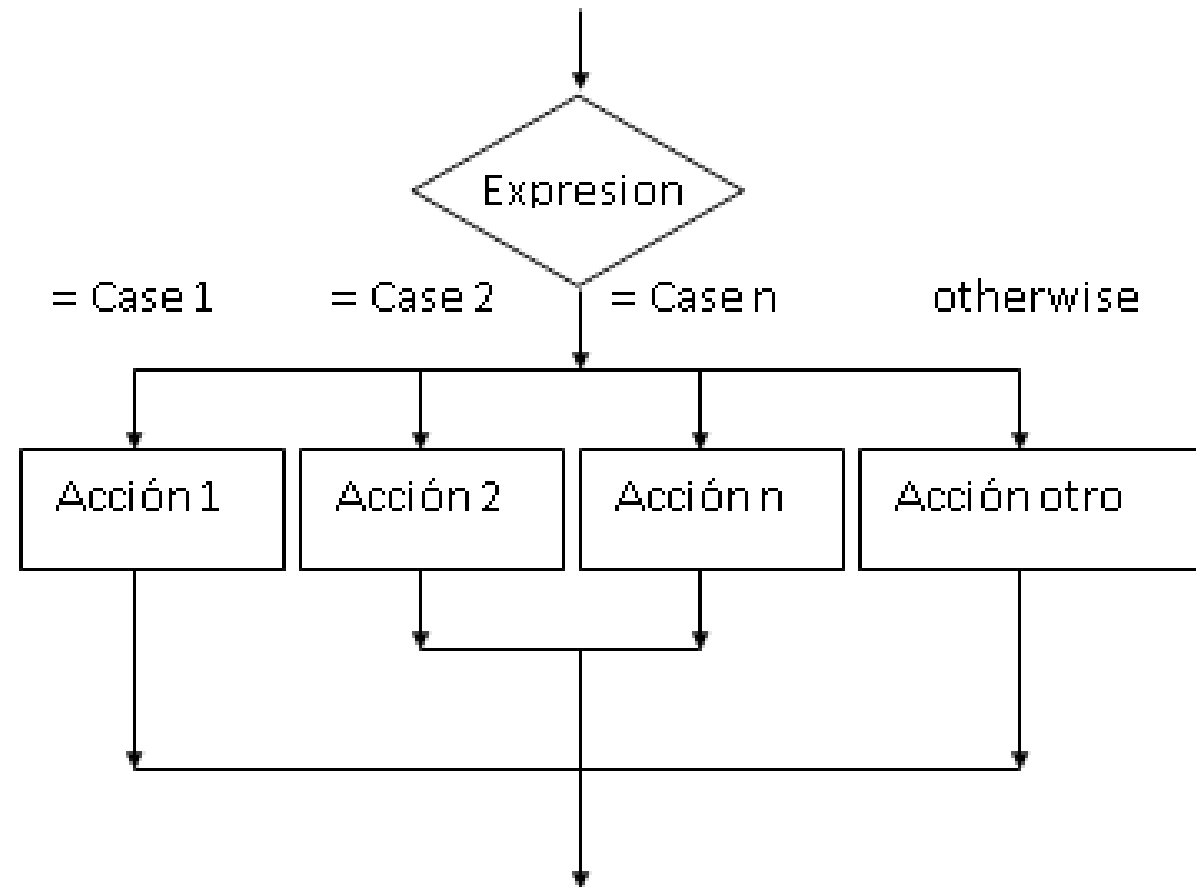
int main()
{
    float a,b,c,x1,x2,r;

    std :: cout << "El programa resuelve las raices de ax^2+bx+c=0\n";
    std :: cout << "Ingrese a= ";
    std :: cin >> a;
    std :: cout << "Ingrese b= ";
    std :: cin >> b;
    std :: cout << "Ingrese c= ";
    std :: cin >> c;

    if (a != 0)
    {
        r = pow(b,2)-4*a*c;
        if (r >= 0)
        {
            x1 = (-b+sqrt(r))/2/a;
            x2 = (-b-sqrt(r))/2/a;
            std :: cout << "Las raices de la ecuacion son: \nx1= " << x1 << "\nx2= " << x2 << "\n";
        }
        else
        {
            std :: cout << "Las raices son numeros complejos, no puedo resolver\n";
        }
    }
    else if (b != 0)
    {
        std :: cout << "La ecuacion es lineal, tiene una raiz\n";
        x1 = -c/b;
        std :: cout << "La raiz es: " << x1 << "\n";
    }
    else
    {
        std :: cout << "Datos ingresado son erroneos";
    }
    return 0;
}
```


Estructuras Selectivas - switch

```
switch (expresión)
{
case caso 1:
    accion 1
break;
case caso 2:
    accion 2
break;
...
case caso n:
    accion n
break;
default:
    otra accion
}
```



Ejemplo 7 - Estructuras Selectivas – switch – par/impar

```
#include <iostream>
using namespace std;
int main()
{
    int res, n;
    cout << "Ingrese número entero: ";
    cin >> n;
    res = n % 2;
    switch (res)
    {
        case 1:
            cout << n << " es impar\n";
            break;
        case 0:
            cout << n << " es par\n";
            break;
        default:
            cout << "Datos incorrectos\n";
    }
    return 0;
}
```