

Resumen de las Funciones

]Lectura del Número de Dígitos

1.

```
cout << "Ingrese el número de dígitos: ";
cin >> n;
```

- **Función:** Lee el número total de dígitos n que se introducirán. Este valor define el tamaño del vector `digits`.

2. Inicialización del Vector

```
vector<int> digits(n);
```

- **Función:** Crea un vector de enteros de tamaño n para almacenar los dígitos introducidos por el usuario.

3. Lectura de los Dígitos

```
for (int i = 0; i < n; i++) {
    cout << "Ingrese el dígito " << i + 1 << ": ";
    cin >> digits[i];
}
```

- **Función:** Un bucle que itera n veces, solicitando al usuario que introduzca cada dígito uno por uno. Cada dígito se almacena en el vector `digits`.

4. Comprobación de Paridad

```
if (n % 2 != 0) {
    cout << "El número de dígitos debe ser par." << endl;
    return 0;
}
```

- **Función:** Verifica si el número total de dígitos n es impar. Si es impar, muestra un mensaje de error y termina la ejecución del programa. Esto asegura que el número de dígitos sea par para poder dividirlo en dos mitades iguales.

5. Cálculo del Índice Medio

```
int mid = n / 2;
```

- **Función:** Calcula el punto medio del vector `digits`, que se usará para dividir el vector en dos mitades.

6. Suma de la Primera Mitad

```
int sum_first_half = 0;
for (int i = 0; i < mid; i++) {
    sum_first_half += digits[i];
}
```

- **Función:** Un bucle que suma los valores de los primeros *mid* dígitos del vector. El resultado se almacena en `sumfirst_half`.

7. Suma de la Segunda Mitad

```
int sum_second_half = 0;
for (int i = mid; i < n; i++) {
    sum_second_half += digits[i];
}
```

- **Función:** Un bucle que suma los valores de los dígitos desde *mid* hasta el final del vector. El resultado se almacena en `sumsecond_half`.

8. Comprobación de la Validez del Boleto

```
if (sum_first_half == sum_second_half) {
    cout << sum_first_half << " 1" << endl;
} else {
    cout << (sum_first_half + sum_second_half) << " 0" << endl;
}
```

- **Función:** Compara las sumas de las dos mitades del vector. Si las sumas son iguales, el boleto es válido y se muestra la suma junto con un "1". Si las sumas son diferentes, el boleto es inválido y se muestra la suma total de ambas mitades junto con un "0".

Código en C++

```
1 #include <iostream>
2 #include <vector>
3
4 using namespace std;
5
6 int main() {
7     int n; // N mero de d gitos
```

```

8      cout << "Ingrese el n mero de d gitos: ";
9      cin >> n;
10
11     vector<int> digits(n);
12
13     // Leer los d gitos del boleto
14     for (int i = 0; i < n; i++) {
15         cout << "Ingrese el d gito " << i + 1 << ": ";
16         cin >> digits[i];
17     }
18
19     // Comprobar si el n mero de d gitos es par
20     if (n % 2 != 0) {
21         cout << "El n mero de d gitos debe ser par." << endl;
22         return 0;
23     }
24
25     int mid = n / 2;
26     int sum_first_half = 0, sum_second_half = 0;
27
28     // Sumar la primera mitad
29     for (int i = 0; i < mid; i++) {
30         sum_first_half += digits[i];
31     }
32
33     // Sumar la segunda mitad
34     for (int i = mid; i < n; i++) {
35         sum_second_half += digits[i];
36     }
37
38     // Comprobar si el boleto es v lido
39     if (sum_first_half == sum_second_half) {
40         cout << sum_first_half << "1" << endl; // Boleto v lido
41     } else {
42         cout << (sum_first_half + sum_second_half) << "0" << endl;
43         // Boleto inv lido
44     }
45
46     return 0;
47 }

```