

```

let x =1; //declara una variable y asigna valor
let total = 0; //declara otra variable para total
let pares = 0; //declara una variable para pares
let impares = 0; //declara una variable para impares

do {
    total += x; //operación de suma acumulativa
    if (x % 2 === 0) { // verifica si x es un número par
        total -= 1; //resta 1 al valor de total
        pares++; //incrementa el valor de pares en 1
    }
    else {
        impares++; //incrementa el valor de impares en 1
    }
    if (total > 15) { // verifica si el total es mayor que 15
        console.log("Límite superado"); //imprime el texto "Límite superado", "alert"
        //muestra en pantalla si esta definido
        break; // Detiene el ciclo
    }

    x++; //aumenta x en 1
}
while (x <=6) // verifica que x llegue hasta el 6

console.log("Total:",total); // imprime el valor de la variable de total
console.log("Pares:",pares); // imprime el valor de la variable de pares
console.log("Impares:",impares); // imprime el valor de la variable de impares

```

Punto 1:

Analiza cuidadosamente el comportamiento del algoritmo

En el algoritmo completo podemos analizar que hace y cuál es su comportamiento, veo que para empezar declara unas variables, luego se usa para la ejecución del ciclo `do` y `while`, luego acumulamos valores, clasificamos los valores de pares e impares, luego usamos un límite mayor a 15 en total, si se supera nos dará un mensaje y se detendrá el ciclo con `break`, por último, usamos una condición final que esto lo que hará será que el ciclo se ejecute hasta el numero limite que es 6.

Describe con tus palabras qué problema podría estar resolviendo este algoritmo

Mi algoritmo resuelve el problema de controlar una acumulación de valores y puede clasificar números pares e impares. También hace una condición límite para evitar que la suma acumulada se pase de 15 y detiene el proceso del algoritmo.

Identifica los requerimientos que derivan del análisis

Requerimientos funcionales: Primero usamos 4 variables para ingresar valores, el sistema debe acumular valores del 1 al 6, también debe acumular valores de pares e impares y si el total supera el numero 15, dar una alerta con un mensaje que nos diga "límite superado".

Entradas y salidas esperadas: Colocamos 4 variables:

```
let x =1;
```

```
let total = 0;
```

```
let pares = 0;
```

```
let impares = 0;
```

Luego usamos las condiciones `do`, `while`, `else`, `if` para ingresar operaciones, comparar y poder tener los resultados.

Para nuestras salidas usamos

```
console.log ("límite superado")
```

```
console.log ("Total:", total);
```

```
console.log ("Pares:", pares);
```

```
console.log ("Impares:", impares);
```

Escenarios de uso:

```
let x =2;
```

```
let total = 0;
```

```
let pares = 0;
```

```
let impares = 0;
```

Nos dará como resultado

Límite superado

Total: 17

Pares: 3

Impares: 2

```
let x =5;
```

```
let total = 0;
```

```
let pares = 0;
```

```
let impares = 0;
```

Nos dará como resultado

Total: 10

Pares: 1

Impares: 1

```
let x =3;
```

```
let total = 0;
```

```
let pares = 0;
```

```
let impares = 0;
```

Nos dará como resultado

Límite superado

Total: 16

Pares: 2

Impares: 2

condiciones y restricciones

Usamos **do**, adentro colocamos la condición de operación de suma acumulativa, luego usamos **if** para ingresar una operación de suma acumulativa, también ingresamos la operación para restarle 1 al valor de total e incrementamos el valor de pares en 1, luego usamos **else** para incrementar el valor de impares en 1, volvemos a usar otro **if** para hacer la operación para verificar si nuestro total es

mayor que 15 y agregamos break para detener el ciclo, usamos x++ para aumentar x en 1 y para finalizar usamos while para hacer la operación que verifique que x llegue a 6.

Uso de operadores lógicos y relacionales

(>) lo use para verificar si total pasa de 15 y de tiene el ciclo con break.

(==) se usó para determinar si x es par.

(<=) se usó para que el ciclo llegue hasta 6.

