**Descripción del programa**

1. **Definición del tipo booleano**:
   * Se define un tipo booleano llamado bool con los valores false (0) y true (1).
2. **Función booleana** esPalindromo:
   * **Propósito**: Verifica si un número entero x es un palíndromo.
   * **Operaciones**:
     + Si x es menor que 0, devuelve false.
     + Si no, almacena el valor original de x en original y establece invertido en 0.
     + Utiliza un bucle while para invertir los dígitos de x y construir el número invertido en invertido.
     + Al final, compara original con invertido y devuelve true si son iguales, y false si no.
3. **Función principal** main:
   * Solicita al usuario que ingrese un número entero.
   * Comprueba si el número ingresado está dentro del rango permitido (-2^31 a 2^31 - 1).
   * Si está dentro del rango, llama a la función esPalindromo para determinar si el número es un palíndromo.
   * Imprime si el número es o no un palíndromo.
   * Si el número no está dentro del rango, imprime un mensaje indicando que el número no está dentro de los límites permitidos.

**Dentro de la función esPalindromo:**

* Asignaciones:
  + original = x
  + invertido = 0
  + digito = x % 10 (dentro del bucle)
  + invertido = invertido \* 10 + digito (dentro del bucle)
  + x /= 10 (dentro del bucle)
* Comparaciones:
  + x < 0
  + x != 0 (dentro del bucle)
  + original == invertido
* Operaciones aritméticas:
  + x % 10 (dentro del bucle)
  + invertido \* 10 + digito (dentro del bucle)
  + x /= 10 (dentro del bucle)

En un número con n dígitos, el bucle while se ejecutará n veces. Por lo tanto, para cada ciclo del bucle, hay 3 asignaciones, 1 comparación y 2 operaciones aritméticas.

**Dentro de la función main:**

* Asignaciones:
  + x
* Comparaciones:
  + x >= pow(-2, 31)
  + x <= pow(2, 31) - 1
  + esPalindromo(x)
* Operaciones aritméticas:
  + pow(-2, 31)
  + pow(2, 31) - 1

**Resumen de Operaciones:**

Función esPalindromo:

- Asignaciones: n \* 3

- Comparaciones: 2 + n

- Operaciones aritméticas: n \* 2

Función main:

- Asignaciones: 1

- Comparaciones: 3

- Operaciones aritméticas: 2

El total de operaciones simples dependerá del número de dígitos de x. ¿Hay algo más en lo que pueda ayudarte?

CODIGO EN C:

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

// Definir un tipo booleano

typedef enum {

false = 0,

true = 1

} bool;

// Función booleana que verifica si un número es par

bool esPalindromo(int x) {

if(x<0){

return false;

}else{

int original = x;

int invertido=0;

while(x!=0){

int digito = x % 10;

invertido = invertido\*10 + digito;

x/=10;

}

return original == invertido;

}

}

int main(int argc, char \*argv[]) {

int x;

printf("Por favor ingrese un numero entero mayor o igual a -2^31 y menor o igual a 2^31 - 1\n");

scanf("%d", &x);

if(x>= pow(-2, 31) && x<= pow(2, 31)-1){

if(esPalindromo(x)){

printf("El numero %d es palindromo", x);

}else{

printf("El numero %d NO es palindromo", x);

}

}else{

printf("El numero proporcionado no esta dentro de los limites");

}

return 0;

}