INSTITUTO POLITECNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

ANÁLISIS Y DISEÑO DE ALGORITMOS

PRÁCTICA 02

Rodrigo García Mayorga

2012630554

Introducción

La división es una de las operaciones aritméticas fundamentales en programación, pero su implementación manual puede ser un reto interesante. En lenguajes como C, la división entre enteros suele devolver el cociente truncado, lo que significa que se ignora la parte decimal del resultado.

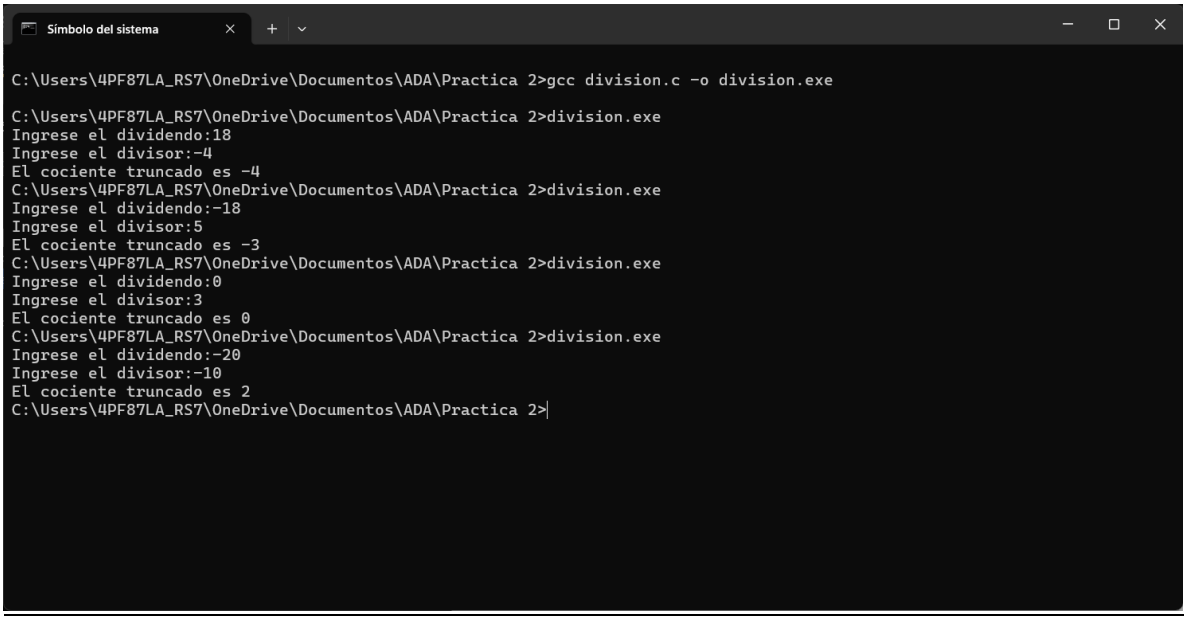
El código presentado implementa la división entera utilizando **restas sucesivas**, un método que consiste en restar repetidamente el divisor del dividendo hasta que el residuo sea menor que el divisor. Este enfoque, aunque simple, tiene implicaciones en términos de **complejidad algorítmica**.

Desde un punto de vista computacional, este algoritmo tiene una complejidad de **O(n)**, donde n representa el valor del cociente. En el peor de los casos, cuando el divisor es 1, el número de iteraciones es igual al valor absoluto del dividendo. Esto lo hace ineficiente para valores grandes, en comparación con algoritmos más avanzados como la **división binaria**, que tiene una complejidad de **O(\log n)**.

A pesar de su ineficiencia para números grandes, este método es útil en entornos con recursos limitados, como **microcontroladores o procesadores sin unidad de división**, donde se busca implementar operaciones básicas utilizando solo sumas y restas.

Capturas

```c
//Rodrigo García Mayorga
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[]){
    int dividendo, divisor, residuo, signo=1, cociente=0;
    printf("Ingrese el dividendo:");
    scanf("%d",&dividendo);
    printf("Ingrese el divisor:");
    scanf("%d",&divisor);

    if(divisor == 0){
        printf("No se puede dividir por cero\n");
        return 0;
    }

    if( (dividendo<0&&divisor>0) || (dividendo>0&&divisor<0) )//si divisor o dividendo son negativos, el signo será negativo
        signo = -1;

    dividendo=abs(dividendo);//Los hacemos positivos a ambos
    divisor=abs(divisor);
    residuo=dividendo;
    while(residuo>=divisor){//hasta que la resta no sea menor que el divisor se seguirá haciendo
        residuo -= divisor;
        cociente++;
    }

    cociente*=signo;
    printf("El cociente truncado es %d", cociente);
    return 0;
}
```



Enlace del código (github):

https://github.com/LowisN/ADA_Practica2.git