**Números Perfectos**

Programación II

**Números Perfectos**

Un número perfecto es un entero positivo que es equivalente a la suma de todos sus divisores excluyendo al número en si. Por ejemplo, el número 6 es perfecto debido a que sus divisores son 1,2,3 (excluyendo al número) y la suma de ellos es 1+2+3=6.

**Ejercicio**

Diseñe un programa que solicite un número al usuario e indique si dicho numero es perfecto o no.

import java.util.Scanner;

public class nperfectos {

public static void main(String[] args) {

//Verificar si el numero ingresado es perfecto

Scanner input=new Scanner(System.in);

System.out.println("Ingresa un numero");

int a=Integer.parseInt(input.next());

int cont=0;

for(int i=1; i<a; i++){

if(a%i==0){

cont+=i;

}

}

if(cont==a){

System.out.println("Es perfecto");

}else{

System.out.println("No es perfecto");

}

}

}

**Explorando Ideas**

1. Modifique su programa para que ahora encuentre los números perfectos menores al numero ingresado. Por ejemplo, si el usuario ingresa el numero 100, entonces el programa debe mostrar en pantalla que ha encontrado dos números perfectos: 6 y 28.

import java.util.Scanner;

public class nperfectos2 {

public static void main(String[] args) {

//Sacar numeros perfectos menores al numero ingresado

Scanner input=new Scanner(System.in);

System.out.println("Ingresa un numero");

long b=Long.parseLong(input.next());

long cont=0;

boolean p=false;

for(long i=1; i<b; i++){

for(long j=1; j<i; j++){

if(i%j==0){

cont+=j;

}

}

if(cont==i){

System.out.println(i+" es perfecto");

p=true;

}

cont=0;

}

if(p==false){

System.out.println("No hay perfectos");

}

}

}

1. ¿Qué sucede cuando el numero ingresado es muy grande?, por ejemplo 1,000,000

Si se están utilizando variables tipo int el programa muestra un error ya que no soporta el número ingresado. Se pueden ingresar números más grandes con la variable tipo long, pero eso no significa que soporte números como por ejemplo 1,000,000,000,000 debido al soporte de RAM.

1. ¿Cuál es el orden del algoritmo implementado? (Es más, ¿Qué es el orden de un algoritmo?)

El orden de un algoritmo pretende medir cuan rápidamente aumenta el tiempo de ejecución de un algoritmo cuando aumenten los datos de entrada.

El orden del logaritmo es lineal por la relación que tiene con el tiempo para calcular los datos.

1. ¿Qué características del programa generan los comportamientos experimentados?

Los tipos de dato y manipulación de las variables dentro de las sentencias, así como la modalidad de emitir el output.

1. Actualmente se están destinando esfuerzos para el desarrollo de Quantum Computing, si implementáramos una solución como un algoritmo en Quantum Computing, ¿Podría mejorar la ejecución del programa?

Definitivamente mejoraría la ejecución del programa, ya que ésta se basa en el uso de cúbits, lo que le permite reducir el margen de error pues utiliza menos recursos y abarca más probabilidades en menor tiempo que un ordenador común.