UNIVERSIDAD AUSTRAL – FACULTAD DE INGENIERÍA

INGENIERÍA INFORMÁTICA **ÁLGEBRA III**

GUÍA N° 1

Algoritmos iterativos y algoritmos recursivos

Para cada uno de los siguientes problemas diseñar y probar, utilizando cualquier lenguaje de programación, algoritmos iterativos y algoritmos recursivos que los resuelvan.

Ejercicio 1

Calcular las siguientes sumas.

a) $1+2+3+\cdots+n$

b) $1^2+2^2+3^2+\cdots+n^2$

c) $1+p+p^2+p^3+\cdots+p^n$

d) $1+\ 3+5+\cdots+2n-1$

e) $1*2+2*3+3*4+\cdots+n*(n+1)$

f) $1^3+2^3+3^3+\cdots+n^3$

Ejercicio 2

Calcular:

a) $n!$

b) 2^n

c) El término n-ésimo de la sucesión de Fibonaci

Ejercicio 3

Determinar la cantidad de ceros que tienen un número natural. Por ejemplo si n = 93020, la respuesta es 2.

Ejercicio 4

Sea $A_n = \{a_1, a_2, ..., a_n\}$, verificar si el conjunto es capicúa, es decir $a_1 = a_n$; $a_2 = a_{n-1}$, etc.

Ejercicio 5

Dado un conjunto de números enteros verificar si un número dado pertenece al conjunto.

Ejercicio 6

Definición: Si b divide exactamente a a (o sea $r = 0$) entonces b es un divisor de a. Todo número entero tiene como divisores al número 1 y a sí mismo. Cuando un número tiene sólo esos dos divisores se dice que el número es *primo*.

Propiedad: Sean n, a y b enteros tal que $n = a.\ b$ y $1 < a < n$ y $1 < b < n$ entonces

$a \leq \sqrt{n}$ ó $b \leq \sqrt{n}$

Esta propiedad nos dice que si un número n tiene algún divisor diferente al número 1 y a sí mismo entonces este divisor debe ser menor o igual a la raíz de n.

a) Demostrar la propiedad anterior.

b) Escribe algoritmos para:

   i) Determinar si un entero positivo, n, es primo.

   ii) Dado un número entero hallar el primer número primo mayor o igual al número dado.

   iii) Determinar la cantidad de factores primos que tiene un número entero.

   iv) Determinar los factores primos que tiene un número entero.

Ejercicio 7

Definición: Sean a y b enteros no nulos, se denomina *máximo común divisor* al mayor entero d tal que d divide exactamente a a y a b.
Notación: El máximo común divisor se denota como *MCD(a, b).*

a) Calcula el MCD de los siguientes pares de enteros.
   i) 123 y 277
   ii) 111 y 201
   iii) 14039 y 1529
b) El algoritmo de Euclides es un método antiguo y eficaz para calcular el máximo común divisor entre dos números. Fue originalmente descrito por Euclides en su obra Elementos. Este algoritmo tiene aplicaciones en diversas áreas como álgebra, teoría de números y ciencias de la computación entre otras. Con unas ligeras modificaciones suele ser utilizado en computadoras electrónicas debido a su gran eficiencia. Investiga el funcionamiento del algoritmo de Euclides y escribe una versión iterativa y una recursiva.

Ejercicio 8

La **regla de Horner** es una forma muy eficiente para evaluar polinomios donde se minimiza el número de operaciones. La regla se puede programar de manera iterativa o recursiva.
Supongamos que se conocen los n coeficientes de un polinomio de grado n, y se lo desea evaluar en x. La regla de Horner es:

$$f(x) = (\cdots((a_n x + a_{n-1})x + a_{n-2})x + \cdots + a_1)x + a_0$$

Ejemplo:
Se desea evaluar el siguiente polinomio de grado 3

$$f(x) = 10x^3 + 5x^2 + 8x + 1$$

usando la Regla de Horner

$$f(x) = (((10x + 5)x + 8)x + 1$$

Escribir una versión iterativa y una recursiva de la regla de Horner. Comparar la cantidad de multiplicaciones y sumas con respecto a la forma tradicional de evaluar un polinomio.