

**Ejercicios Propuestos de programación - Semana 1**

**Semana 1**

Escriba un algoritmo correspondiente a cada problema usando el lenguaje Java. Indique las especificaciones de cada algoritmo

1. Dado un número entero N, determine la cantidad de dígitos del número N.
2. Dado un número entero N, determine el número invertido del número N.
3. Dado un número entero N, hallar su valor en base binaria.
4. Dado un número natural N, calcular la raíz digital de N de la siguiente manera:

* Se suman los dígitos que lo componen; el proceso se repite sobre el nuevo número hasta que el resultado obtenido tenga un solo dígito.
* Este último número es la raíz digital del número N.

**Ejemplo:**

347 --> 3 + 4 + 7 = 14

14 --> 1 + 4 = 5

Entonces la raíz digital de 347 es 5.

1. Dado un número N, entero mayor que cero, escriba un algoritmo que verifique si el N es primo o no.
2. Dado un número entero N, y una posición i, determine el dígito que ocupa la posición i del número N contando de derecha a izquierda.
3. Escriba un algoritmo que muestre la tabla de multiplicar de cualquier número del 1 al 12.
4. Se tiene una relación de N estudiantes de Algorítmica I, para cada estudiante se tiene 3 notas. Escribir un algoritmo que calcule el promedio de cada alumno.
5. Escribir un algoritmo que evalúe la siguiente expresión:

12 – 22 + 32 – 42 + 52 – 62 + … + 992 – 1002

1. Escribir un algoritmo para determinar el mayor de N números enteros positivos.
2. Se tiene la relación de promedios finales de los estudiantes de EDA, Escribir un programa que determine: El Promedio de la clase, La nota mayor, La nota menor,

Número de estudiantes en las siguientes categorías:

1. 15 <= Nota
2. 10 <= Nota < 15
3. 5 <= Nota < 10
4. Nota < 5
5. Dos números primos (p, q) son números primos gemelos si, siendo q > p, se cumple q -p = 2. Por ejemplo 11 y 13. Escriba un algoritmo para imprimir los n primeros pares de primos gemelos.

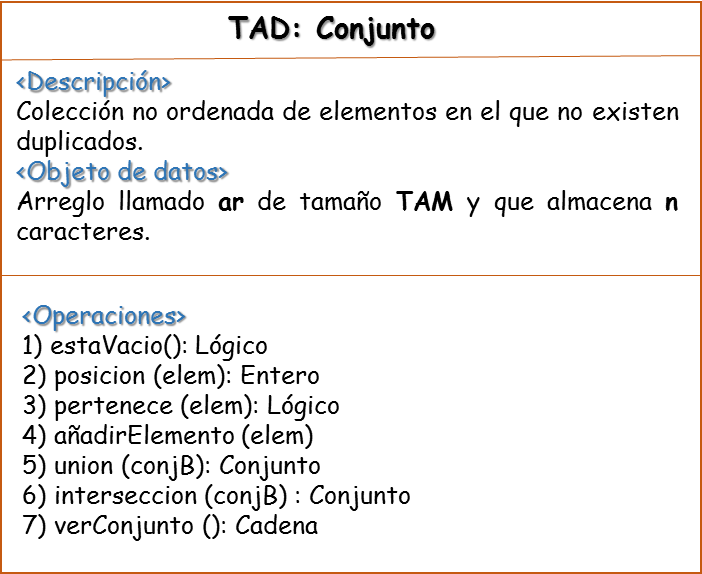
**Ejercicios de Programación Orientada a Objetos y TDA**

Para cada uno de los siguientes TDAs proporcione las especificaciones correspondientes. Implemente uno de ellos usando el lenguaje Java.

1. Diseñe el **TDA Conjunto**, que incluye las siguientes operaciones:
   1. **EstaVacio**, operación que retorna **true** si el conjunto está vacío, **false** en caso contrario.
   2. **Posición,** operación que retorna la posición del elemento, si se encuentra, en caso contrario retorna -1.
   3. **Pertenece**, operación que retorna un valor lógico **true**, si el elemento está en el conjunto, **false** en caso contrario.
   4. **AñadirElemento**, operación que inserta en el conjunto un nuevo elemento; siempre que no se encuentre.
   5. **Unión**, operación que retorna el conjunto que representa la unión de dos conjuntos A y B
   6. **Interseccion**, operación que retorna el conjunto que representa la intersección de dos conjuntos A y B (elementos comunes a ambos)
   7. **verConjunto**, operación que retorna en forma de cadena la estructura del conjunto.

**Especificación en Notación Algorítimica:**

* Esquema del TDA



1. Diseñe el **TDA Fraccion**, que incluya las siguientes operaciones:
2. ***NumeradorCanónico****,* operación que devuelve el numerador del término canónico asociado.
3. ***DenominadorCanónico****,* operación que devuelve el denominador del término canónico asociado.
4. ***EsEntero****,* operación booleana que decide si una fracción es un número entero.
5. ***EsFraccionNula****,* operación booleana que decide si una fracción es nula.
6. ***Menor****,* operación booleana que recibe dos fracciones y decide si la primera es menor que la segunda.
7. ***Suma****,* ***Resta****,* ***Multiplicación***y ***División****,* operaciones que permiten realizar operaciones aritméticas entre fracciones. Téngase en cuenta la posible parcialidad de alguna de estas operaciones

**Nota**: en la especificación anterior se supondrán disponibles las siguientes operaciones para el tipo de dato *Entero*: ***abs***(valor absoluto), ***mcd***(máximo común divisor para enteros positivos), y ***Signo***(que devuelve –1 si el entero pasado como parámetro es negativo y 1 en caso contrario).

1. Diseñe el **TDA *Triángulo*** en el plano, representado por tres vértices en R x R, que incluya las siguientes operaciones:
   1. ***Área:*** que devuelva el área del triángulo.
   2. ***Perímetro:*** devuelve el perímetro del triángulo.
   3. ***TipoTriangulo***: calcule y devuelva un texto que indique si el triángulo es equilátero, o isósceles o escaleno.
   4. ***TipoAngularidad:*** calcule y devuelva un texto que indique si el triángulo es rectángulo, acutángulo u obtusángulo,
2. Diseñe el **TDA Complejo**, que incluya las operaciones siguientes:
3. ***EsImaginarioPuro****,* operación booleana que decide si un número complejo es imaginario puro (es decir, no tiene parte real).
4. ***EsReal****,* operación booleana que decide si un número complejo es un número real.
5. ***EsNulo****,* operación booleana que decide si un número complejo es nulo.
6. ***Argumento****,* operación que recibe como entrada un número complejo no nulo y devuelve su argumento. Se recuerda a continuación el valor del argumento de un número complejo *a + bi* no nulo (número real comprendido entre –Pi y Pi):

| argumento | condición |
| --- | --- |
| arctg (b/a) | Si a > 0 |
| arctg (b/a) + pi | Si a < 0 y b >= 0 |
| arctg (b/a) - pi | Si a < 0 y b < 0 |
| pi / 2 | Si a + 0 y b > 0 |
| -pi / 2 | Si a = 0 y b < 0 |

1. El **TDA Polinomio**, el cual representa al polinomio de una sola variable, P(x). Incluye las siguientes operaciones:
   1. **Suma, Resta, y Multiplicación,** operaciones que calculan la suma, resta, multiplicación y división del par de polinomios P(x) y Q(x).
   2. **Evaluación**: operación que evalúa el polinomio P(x) para un valor de la variable x.
   3. **Diferenciación**: operación que determina la derivada del polinomio P(x).
   4. **IntegracionDefinida**: operación que determina la integral de P(x), cuyos límites de integración son a y b. (a<b).