

Programación II

Trabajo Práctico 1 – TDAs: Conceptos Básicos. Definición, implementación y utilización. Cálculo de Costos.

TDA Pila

- 1) Definir el TDA Pila, listando las operaciones asociadas y establecer sus precondiciones.
- 2) A partir del TDA Pila definido, escribir distintos métodos que permitan
 - a) Pasar una Pila a otra (dejándola en orden inverso)
 - b) Copiar una Pila en otra (dejándola en el mismo orden que la original)
 - c) Invertir el contenido de una Pila.
 - d) Contar los elementos de una Pila
 - e) Sumar los elementos de una Pila
 - f) Calcular el promedio de los elementos de una Pila

TDA Cola

- 3) Definir el TDA Cola, listando las operaciones asociadas y establecer sus precondiciones.
- 4) A partir del TDA Cola definido, escribir distintos métodos que permitan
 - a) Pasar una Cola a otra
 - b) Invertir el contenido de una Cola (pueden usarse Pilas auxiliares)
 - c) Invertir el contenido de una Cola (NO pueden usarse Pilas auxiliares)
 - d) Determinar si los elementos finales de la Cola C1 coinciden o no con los de la Cola C2.
 - e) Determinar si una Cola es capicúa o no. Para ser capicúa debe cumplir que el primer elemento es igual al último, el segundo igual al penúltimo, etc.
 - f) Determinar si la Cola C1 es la inversa de la Cola C2. Dos Colas serán inversas, si tienen los mismos elementos, pero en orden inverso.

TDA Cola con Prioridades

- 5) Definir el TDA Cola con prioridades, listando las operaciones asociadas y establecer sus precondiciones.
- 6) A partir del TDA Cola con prioridades definido, escribir un método que permita
 - a) Combinar dos colas con prioridades CP1 y CP2, generando una nueva cola con prioridades. Considerar que a igual prioridad, los elementos de la CP1 son más prioritarios que los de la CP2.
 - b) Determinar si dos Colas con prioridad son idénticas.

Implementaciones y Costos

- 7) Escribir al menos dos implementaciones distintas (basadas en arreglos) del TDA Pila definido en 1). Comparar los costos de cada una de las operaciones.
- 8) Escribir al menos dos implementaciones distintas (basadas en arreglos) del TDA Cola definido en 3). Comparar los costos de cada una de las operaciones.
- 9) Escribir al menos dos implementaciones distintas (basadas en arreglos) del TDA Cola con prioridades definido en 5). Comparar los costos de cada una de las operaciones.

Trabajo Práctico 2 TDAs: Implementación utilizando estructuras dinámicas. Costos. TDA Conjunto. TDA Diccionario.

TDA Pila, TDA Cola y TDA Cola con Prioridades

1. Implementar los TDA Pila, TDA Cola y TDA Cola con Prioridades definidos en los ejercicios número 1), 3) y 5) del TP1 (respectivamente) con listas dinámicas.
2. Calcular y comparar los costos de las operaciones típicas de cada uno de los TDAs anteriores para las implementaciones basadas en arreglos del TP1 y la propuesta en el ejercicio anterior.

TDA Conjunto

1. Definir el TDA Conjunto, listando las operaciones asociadas.
2. Implementar el TDA Conjunto con las siguientes restricciones
 - 2.1. Tamaño máximo acotado
 - 2.2. Tamaño máximo no acotado
 - 2.3. Universo acotado. Considerar por ejemplo el Universo de los números enteros entre 0 y N.

En todos los casos, dar al menos dos implementaciones utilizando arreglos y con listas dinámicas.

3. Comparar los costos de las operaciones definidas en el TDA Conjunto según las implementaciones del ejercicio anterior.
4. Escribir los métodos externos al TDA que implementan las operaciones intersección, unión y diferencia.

TDA Diccionario

5. Definir el TDA Diccionario, listando las operaciones asociadas. Considerar los dos casos vistos en clase: a) cada clave está asociada a un único valor, y b) cada clave está asociada a un conjunto de valores.
6. Implementar el TDA Diccionario, considerando las dos alternativas del ejercicio anterior
 - 6.1. cada clave está asociada a un único valor
 - 6.2. cada clave está asociada a un conjunto de valores.

En ambos casos, dar al menos una implementación utilizando arreglos y una con listas dinámicas.

7. Comparar los costos de las operaciones definidas en el TDA Diccionario según las implementaciones del ejercicio anterior.

Trabajo Práctico 3 Utilización de TDAs.

(En todos los ejercicios siguientes calcular el costo espacial y temporal de los métodos escritos)

1. A partir del **TDA Pila**, escribir distintos métodos externos que permitan:
 - 1.1. Comprobar si una Pila P es capicúa (el elemento del tope es igual al de la base, el segundo igual al anteúltimo, etc.)
 - 1.2. Eliminar de una Pila P las repeticiones de elementos, dejando un representante de cada uno de los elementos presentes originalmente. Se deberá respetar el orden original de los elementos, y en el caso de los repetidos se conservará el primero que haya ingresado en P.
 - 1.3. Repartir una Pila P en dos mitades M1 y M2 de elementos consecutivos, respetando el orden. Asumir que la Pila P contiene un número par de elementos.
 - 1.4. Generar el conjunto de elementos que se repiten en una Pila.
2. A partir del **TDA Cola**, escribir distintos métodos externos que permitan
 - 2.1. Eliminar de una Cola C las repeticiones de elementos, dejando un representante de cada uno de los elementos presentes originalmente. Se deberá respetar el orden original de los elementos, y en el caso de los repetidos se conservará el primero que haya ingresado en C.
 - 2.2. Repartir una Cola C en dos mitades M1 y M2 de elementos consecutivos, respetando el orden. Asumir que la cantidad de elementos de C es par.
 - 2.3. Generar el conjunto de elementos que se repiten en una Cola,
3. A partir del **TDA Conjunto**, escribir distintos métodos externos que permitan
 - 3.1. Calcular la diferencia simétrica entre dos conjuntos A y B (definido en clase).
 - 3.1.1. Sin utilizar las operaciones unión, intersección y diferencia.
 - 3.1.2. Utilizando las operaciones unión, intersección y diferencia.
 - 3.2. Determinar si dos conjuntos son iguales.
 - 3.3. Calcular la cardinalidad (cantidad de elementos) de un conjunto.
 - 3.4. Generar el conjunto de elementos que están tanto en la Pila P y en la Cola C.
 - 3.5. Determinar si los elementos de una Pila P son los mismos que los de una Cola C. No interesa el orden ni si están repetidos o no.
4. A partir del **TDA ColaPrioridad**
 - a) Escribir un método externo que permita generar un Diccionario Múltiple que permita, para cada valor presente en la ColaPrioridad C recuperar todas las prioridades que tiene asociadas en C.
5. A partir del **TDA Diccionario**, escribir distintos métodos externos que dados dos DiccionarioMultipleTDA D1 y D2, permitan generar un DiccionarioMultipleTDA que contenga:
 - 5.1. las claves presentes en D1 y D2, con todos los valores asociados a cada clave.
 - 5.2. las claves presentes en D1 y D2 con sus elementos asociados, pero en el caso que las claves coincidan, solo con los valores comunes de las claves.
 - 5.3. las claves comunes de D1 y D2, con todos los valores asociados a cada clave.
 - 5.4. las claves comunes de D1 y D2, con todos los valores comunes a las claves coincidentes en ambos.
6. Dado un Diccionario Simple D, que representa el concepto clásico de diccionario: la clave representa una palabra y el valor su significado. Generar un Diccionario Múltiple DS que a partir de un significado s, vincule todas las palabras que tienen dicho significado, es decir que son sinónimos. Cada clave s será un significado y los valores asociados (sinónimos) aquellas claves de D que tenían asociado el valor s.

Trabajo Práctico 4 TDA Árbol Binario de Búsqueda (ABB). Uso de la Recursión.

(En todos los ejercicios siguientes calcular el costo espacial y temporal de los métodos escritos)

TDA ABB

- 1) Definir el TDA ABB, con las siguientes operaciones asociadas (según lo visto en clase):
 - a) Raiz
 - b) HijoIzq
 - c) HijoDer
 - d) ArbolVacio
 - e) InicializarArbol
 - f) AgregarElem
 - g) EliminarElem
- 2) Implementar el TDA ABB definido en el ejercicio anterior, utilizando estructuras dinámicas.

Utilización del TDA ABB / Uso de la Recursión.

- 3) A partir del TDA ABB, escribir métodos externos que resuelvan los siguientes problemas. En caso de ser posible, escribir la versión iterativa y la versión recursiva de los métodos.
 - a) Dado un elemento, determinar si está o no en un ABB.
 - b) Dado un elemento, determinar si es una hoja de un ABB.
 - c) Dado un elemento, calcular su profundidad en el ABB.
 - d) Obtener el valor del menor elemento de un ABB.
 - e) Calcular la cantidad de elementos que contiene un ABB.
 - f) Calcular la suma de los elementos que contiene un ABB.
 - g) Calcular la cantidad de hojas de un ABB
 - h) Calcular la altura de un ABB.
 - i) Comprobar si dos ABBs tienen la misma forma.
 - j) Comprobar si dos ABBs son iguales.
 - k) Contar la cantidad de elementos que están en un cierto nivel N.
 - l) Mostrar por pantalla todos los elementos que contiene un ABB en:
 - i. In-orden
 - ii. Pre-orden
 - iii. Post-orden
 - m) Dado un valor k, arme un conjunto con todos los elementos del ABB que son mayores que k.
 - n) Dado un elemento de valor v (que está presente en el ABB), obtener el elemento del árbol que es inmediatamente anterior (en valor).

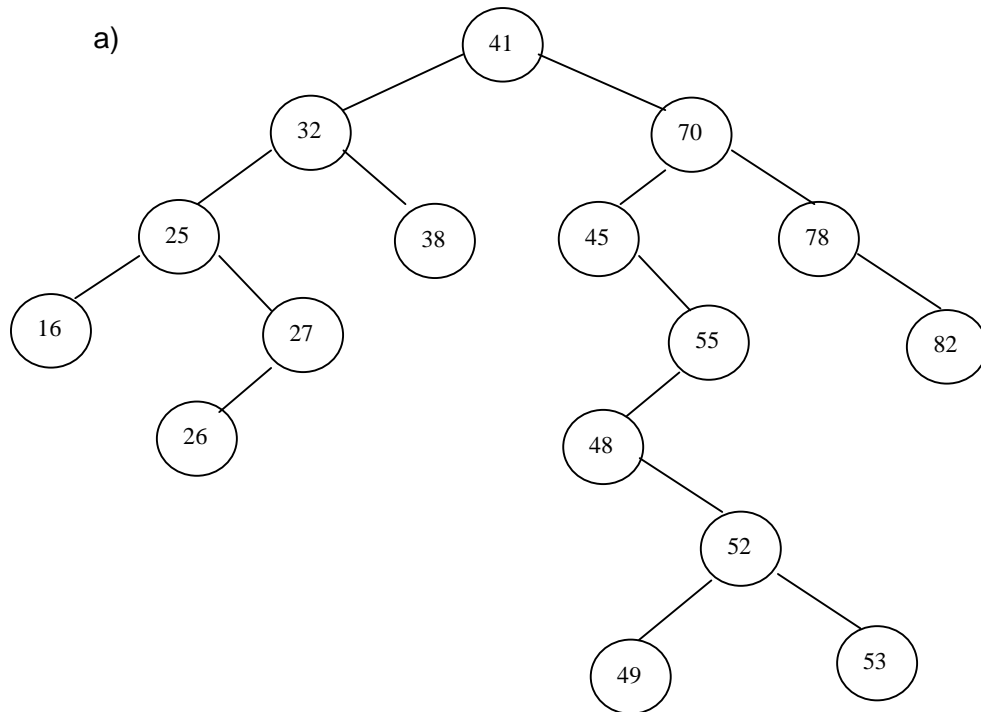
Trabajo Práctico 5 –Árbol Binario de Búsqueda Balanceado (AVL) y Árbol B.

(En todos los ejercicios siguientes calcular el costo espacial y temporal de los métodos escritos)

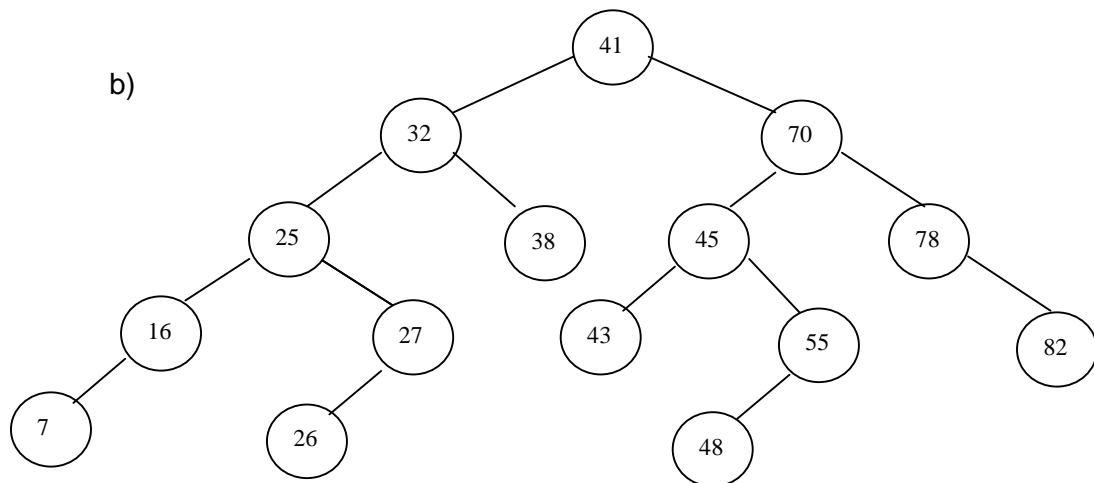
Árbol AVL

- 1) Indicar si los siguientes árboles binarios de búsqueda cumplen con la propiedad de AVL y justificar la respuesta. En caso negativo, indicar si se puede balancear con rotaciones a izquierda o derecha, simples o dobles. En aquellos que sea posible el balanceo llevar a cabo el mismo mostrando la secuencia de pasos correspondientes:

a)

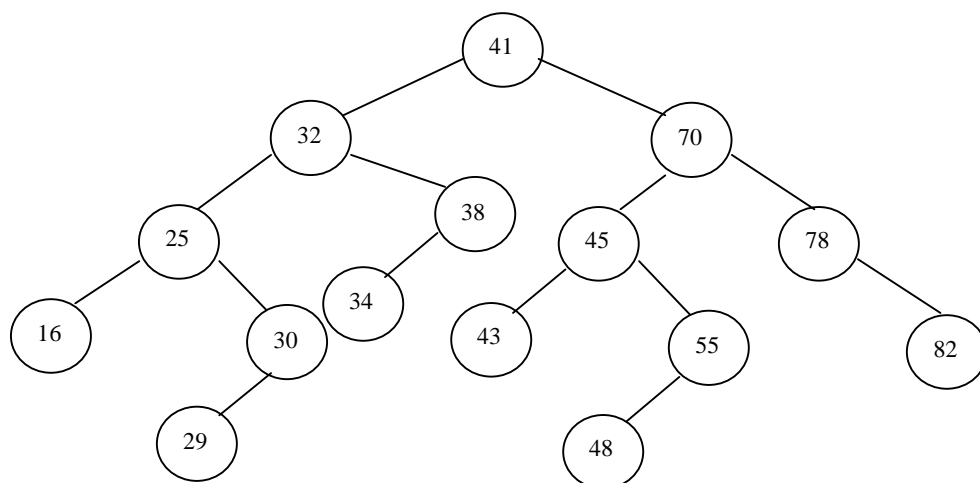


b)



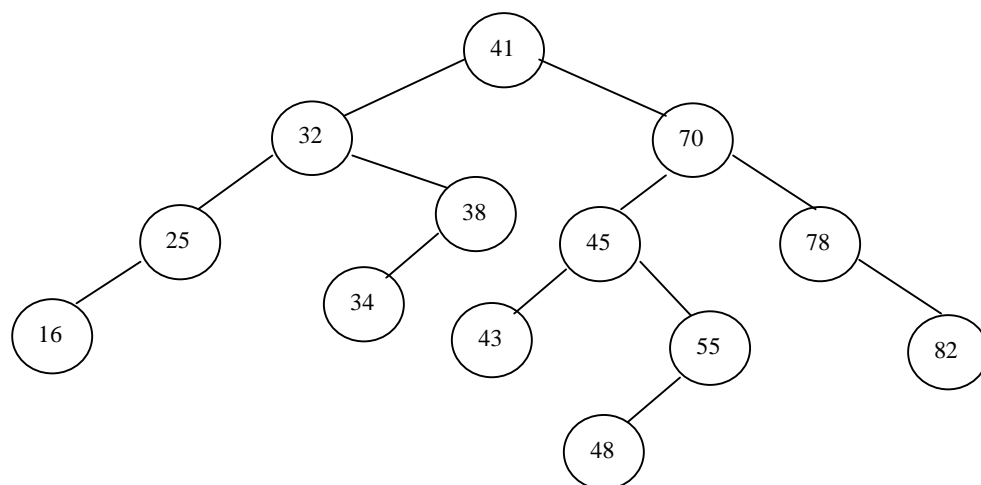
- 2) Dados los siguientes árboles AVL, insertar los valores que se indican. En caso de que el árbol no cumpla la propiedad de AVL, mostrar la secuencia de pasos que se deberían llevar a cabo para que el árbol vuelva a ser un AVL.

a) Insertar el valor 28.



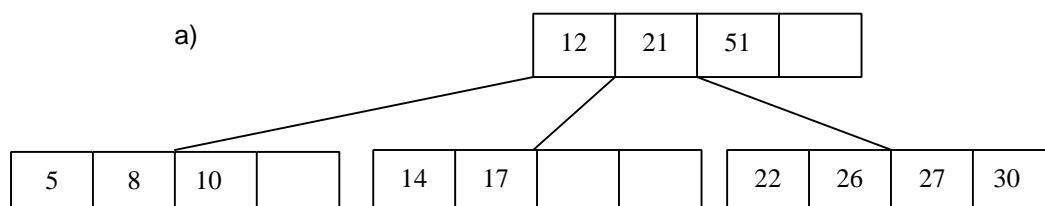
b) Al árbol resultante de la inserción anterior, agregar el valor 47.

c) Insertar el valor 84

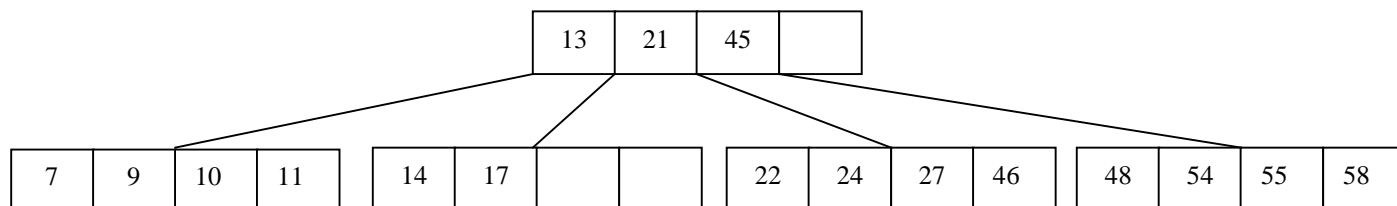


Árbol B

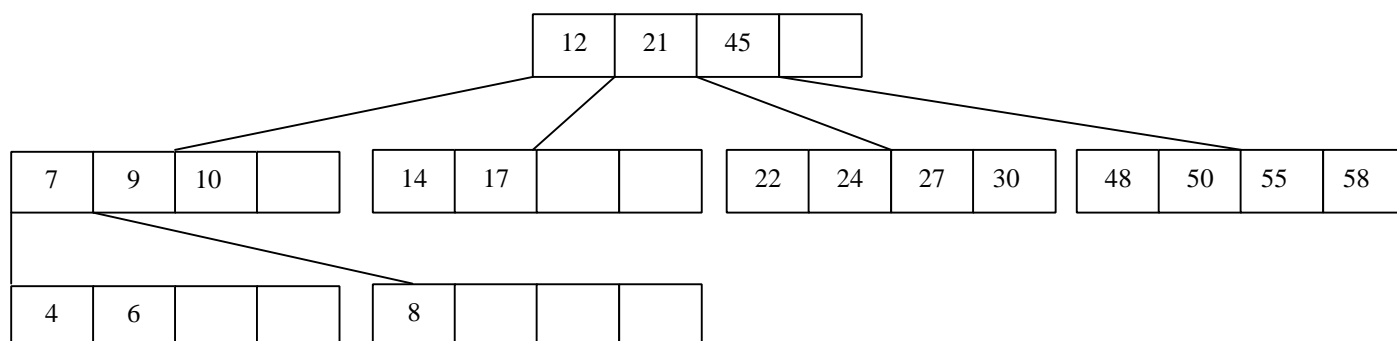
3) Dados los siguientes árboles, indicar si los mismos son árboles B, y en caso negativo indicar por qué.



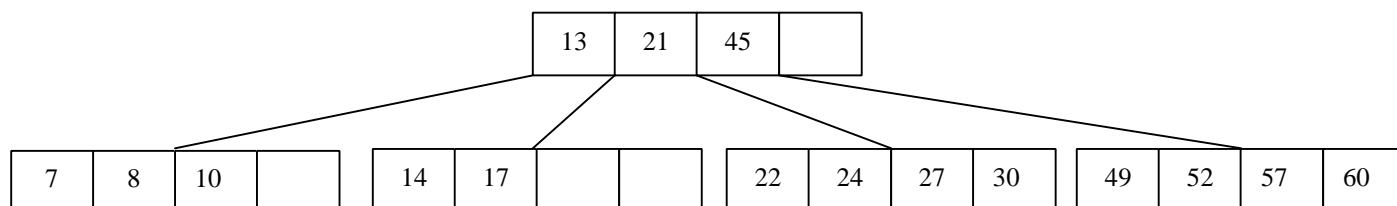
b)



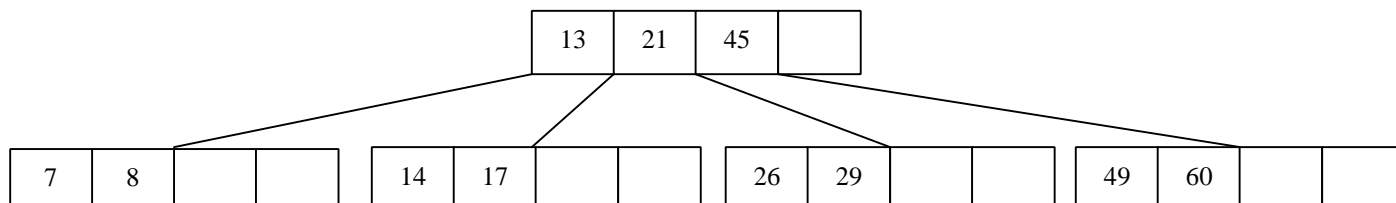
c)



- 4) Dado el siguiente árbol B, insertar el valor 67 en primer lugar. Al árbol resultante insertar el valor 32. En caso de requerir reestructuración del árbol, mostrar la secuencia de la misma.



- 5) Dado el siguiente árbol B, eliminar el valor 45 en primer lugar.



Trabajo Práctico 6 – Grafos

(En todos los ejercicios siguientes calcular el costo espacial y temporal de los métodos escritos)

TDA Grafo

- 1) Definir la interface del TDA Grafo, con las operaciones asociadas (según lo visto en clase):
- 2) Implementar el TDA Grafo definido en el ejercicio anterior, utilizando matriz de adyacencia.
- 3) Implementar el TDA Grafo definido en el ejercicio anterior, utilizando listas de adyacencia.

Utilización del TDA Grafo

- 4) Dado un Grafo G y un vértice v , calcular el conjunto de vértices *AdyacentesDobles* de v . Se define que un vértice w es *adyacente doble* de un vértice v , si existe otro vértice x y hay una arista que comienza en v y termina en x y otra que comienza en x y termina en w .
- 5) Dado un vértice v de un grafo, calcular el mayor de los costos de las aristas salientes.
- 6) Dado un Grafo G y un vértice v , escribir un método que permita obtener el conjunto de los *Predecesores* del vértice v en G .
Se define que un vértice o es *predecesor* de otro vértice d , si hay una arista que comienza en o y termina en d .
- 7) Dado un Grafo G escribir un método que permita obtener el conjunto de los *vértices aislados* en G .
Se define que un vértice v es *aislado* si v no tiene aristas entrantes ni salientes.
- 8) Dado un Grafo G y dos vértices v_1 y v_2 , escribir un método que permita obtener el conjunto de todos los vértices *punte* entre v_1 y v_2 .
Se define que un vértice p es *punte* entre dos vértices o y d , si hay una arista que comienza en o y termina en p y otra que comienza en p y termina en d .
- 9) Dado un Grafo G y un vértice v , calcular el grado de v .
Se define el grado de un vértice v como el entero que es igual a la resta entre la cantidad de aristas que salen de v menos la cantidad de aristas que llegan a v .