**ESTRUCTURAS DE DATOS Y ALGORITMOS**

**ACTIVIDAD EN AULA**

**12.04.2019**

**TEMA:** Tablas Hash, lo básico

En parejas, miren el video sobre Tablas Hash, Parte 1 y luego desarrollen las siguientes actividades:

1. Escribe el seudo-código de la función hash(n) que recibe como argumento de entrada un número y retorna el resultado de aplicar la función de hash. Usa la función mostrada como ejemplo en el video: (2n+3)mod 9

function h(n):

return (2\*n + 3) mod 9

1. Escribe el seudo-código de la función BUSCAR\_HASH(A,k), que recibe como argumento de entrada el arreglo (A) donde se buscará el número y el número (k) a buscar. Puedes usar la función anterior para implementar esta.

function BUSCAR\_HASH(A,k):

i 🡨 h(k)

if (A[i] == -1):

print(“El número no se encuentra”)

else:

print(“El número se encuentra en la posición %d” %(A[i]))

1. Realiza el análisis asintótico de las dos funciones anteriores (O(), Ω() y Θ()).

Como la función h(n) consiste en 3 operaciones aritméticas, el costo es una constante, por lo es de Θ(1). [Peor caso y mejor caso tienen el mismo costo]

La función BUSCAR\_HASH(A,k) tiene una asignación, una comparación y un print en cada caso de la condición. Por lo que también tiene un costo constante y podemos concluir que es Θ(1). [Peor caso y mejor caso tienen el mismo costo]

1. Escribe el programa más simple que puedas para:
   1. Crear un arreglo de 9 posiciones, inicialmente lleno de -1s.
   2. Usando la función hash de ejemplo del video, ingresa los siguientes números al arreglo: {13, 5, 28, 9, 43, 39, 23, 27, 17}
   3. Ahora registra los resultados de buscar los siguientes números en el arreglo: {18, 9, 17, 5}. ¿Detectan algún problema?
   4. Describan el problema que ocurrió en el ejercicio anterior e inventen (OJO: “inventen”, no “busquen en Internet”) al menos dos maneras de resolverlo.

Ocurrió este problema

[39, 17, 13, 27, 23, 28, -1, -1, 43]

El número 18 no se encuentra en el arreglo

El número 9 no se encuentra en el arreglo

El número 17 se encuentra en la posición 1

El número 5 no se encuentra en el arreglo

Primera solución:

* Guardar los datos como string y concatenándolos en cierto formato entendible.

Segunda solución:

* Crear arreglos bidimensionales.
  1. Implementa las soluciones propuestas y vuelve a buscar los mismos números.

def h(n):

return (2\*n + 3 ) % 9

def ingresar(A, k):

i = h(k)

j= 0

if A[i][j] == -1:

A[i][j] = k

else:

A[i].append(k)

return (A)

def BUSCAR\_HASH (A,k):

i = h(k)

esta = False

for s in (A[i]):

if (k == s):

esta = True

if esta:

print("el número %s está en el arreglo" %(k))

else:

print("el número %s no está" %(k))

return

A = []

for i in range(9):

A.append([-1])

print(A)

for i in [13, 5, 28, 9, 43, 39, 23, 27, 17]:

A = ingresar(A, i)

print(A)

for i in [18, 9, 17, 5]:

BUSCAR\_HASH(A,i)

* 1. ¿Se resolvió el problema? ¿Podría volver a haber problemas incluso con la solución implementada?

SIII la solución 2 sí funcionó

[[-1], [-1], [-1], [-1], [-1], [-1], [-1], [-1], [-1]]

[[39], [17], [13], [9, 27], [5, 23], [28], [-1], [-1], [43]]

el número 18 no está

el número 9 está en el arreglo

el número 17 está en el arreglo

el número 5 está en el arreglo