
C++

EJERCICIOS
HASHING

Colisiones: Open adresing

- Si se produce una colisión se busca una nueva posición tomando alguna nueva función que prueba en primer lugar con el valor 1, luego con el 2, etc.
- Si $h(k)$ está ocupada prueba con $h(k) + p(1)$, y si también lo está intenta con $h(k) + p(2)$, etc., donde p es una nueva función.
- Al finalizar siempre se aplica la función módulo.

Colisiones: Open addressing

Ejercicio 1: Si se usa módulo de 11 en una tabla de tamaño 11 y la solución de colisiones es direccionamiento abierto, indicar cómo funciona cada operación y cómo va quedando la tabla en cada paso: insertar(37), insertar(12), insertar(8), insertar(26), eliminar(37), insertar(34), buscar(26).

Colisiones: Bucket addressing (Extensible)

Ejercicio 2: (Hashing extensible). Suponiendo la siguiente función de hashing: $h(k)=k \bmod 16$, usando como tratamiento de colisiones el bucket addressing, con buckets de 3 registros, insertar los siguientes registros: 2369, 3760, 4642, 4871, 5657, 1821, 1074, 7115, 1620, 2428, 3543, 4750.

$$H(2369)=2369\%16=1=0001$$

$$H(3760)=3760\%16=0000$$

$$H(4692)=4692\%16=0100$$

$$H(4871)=4871\%16=0111$$

$$H(5659)=5659\%16=1011$$

$$H(1821)=1821\%16=1101$$

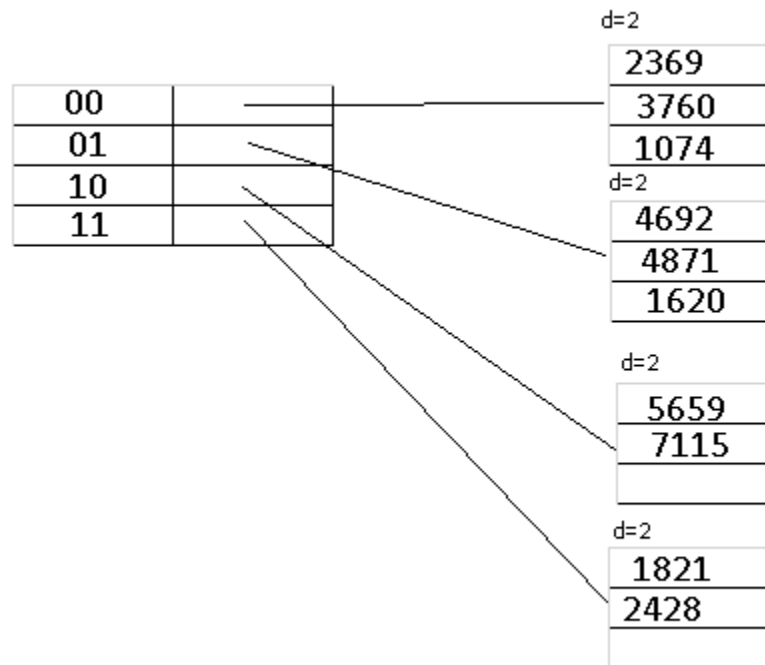
$$H(1074)=1074\%16=0010$$

$$H(7115)=7115\%16=1011$$

$$H(1620)=1620\%16=0100$$

$$H(2428)=2428\%16=1100$$

Colisiones: Bucket addressing (Extensible)



$$H(2369) = 2369 \% 16 = 1 = 0001$$

$$H(3760) = 3760 \% 16 = 0000$$

$$H(4692) = 4692 \% 16 = 0100$$

$$H(4871) = 4871 \% 16 = 0111$$

$$H(5659) = 5659 \% 16 = 1011$$

$$H(1821) = 1821 \% 16 = 1101$$

$$H(1074) = 1074 \% 16 = 0010$$

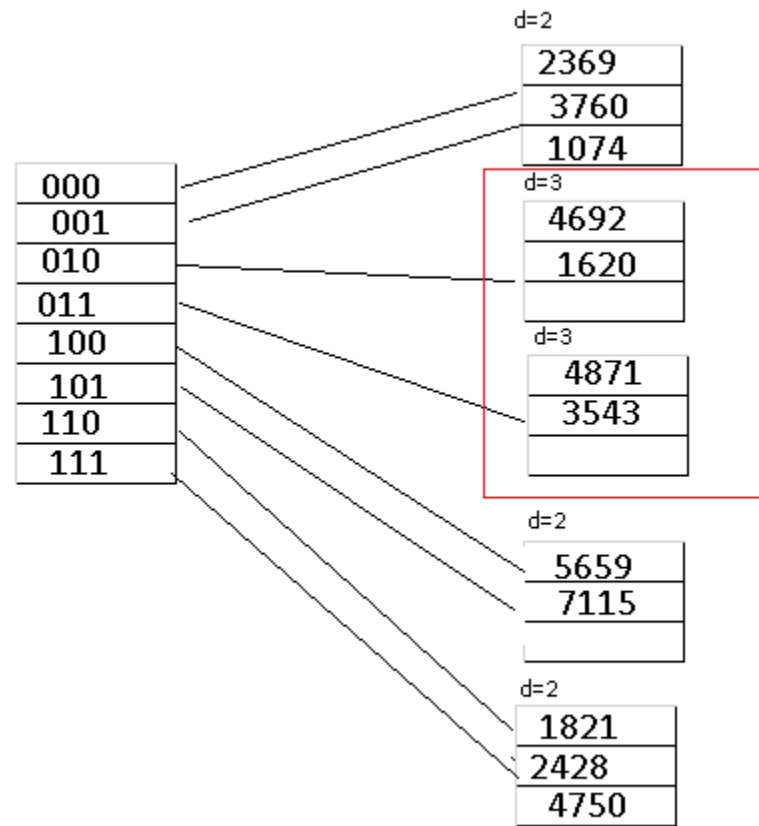
$$H(7115) = 7115 \% 16 = 1011$$

$$H(1620) = 1620 \% 16 = 0100$$

$$H(2428) = 2428 \% 16 = 1100$$

$$H(3543) = 3543 \% 16 = 0111 \text{ overflow bucket 2!}$$

Colisiones: Bucket addressing (Extensible)



Fin
