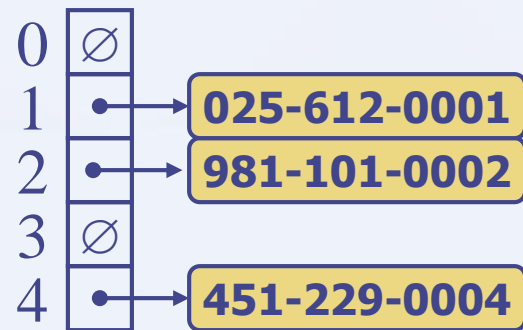


# 3. Tìm kiếm theo địa chỉ

\*\*\*

## Bảng băm - Hash Tables



# I. Hàm băm

## ◆ Cấu trúc hàm băm

- Hàm băm có dạng như sau:

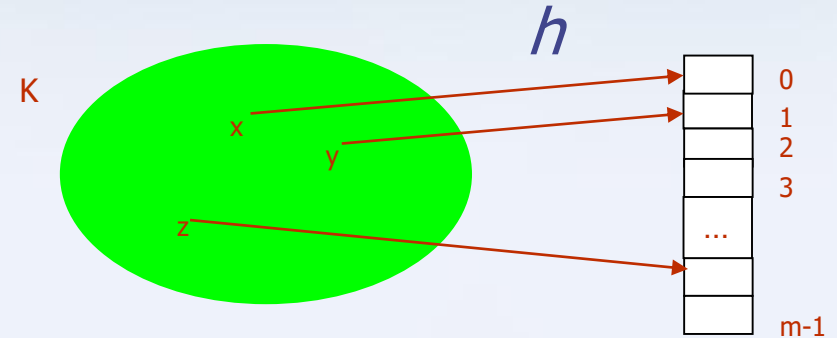
$$h : K \rightarrow 0..m-1$$

- Trong đó:

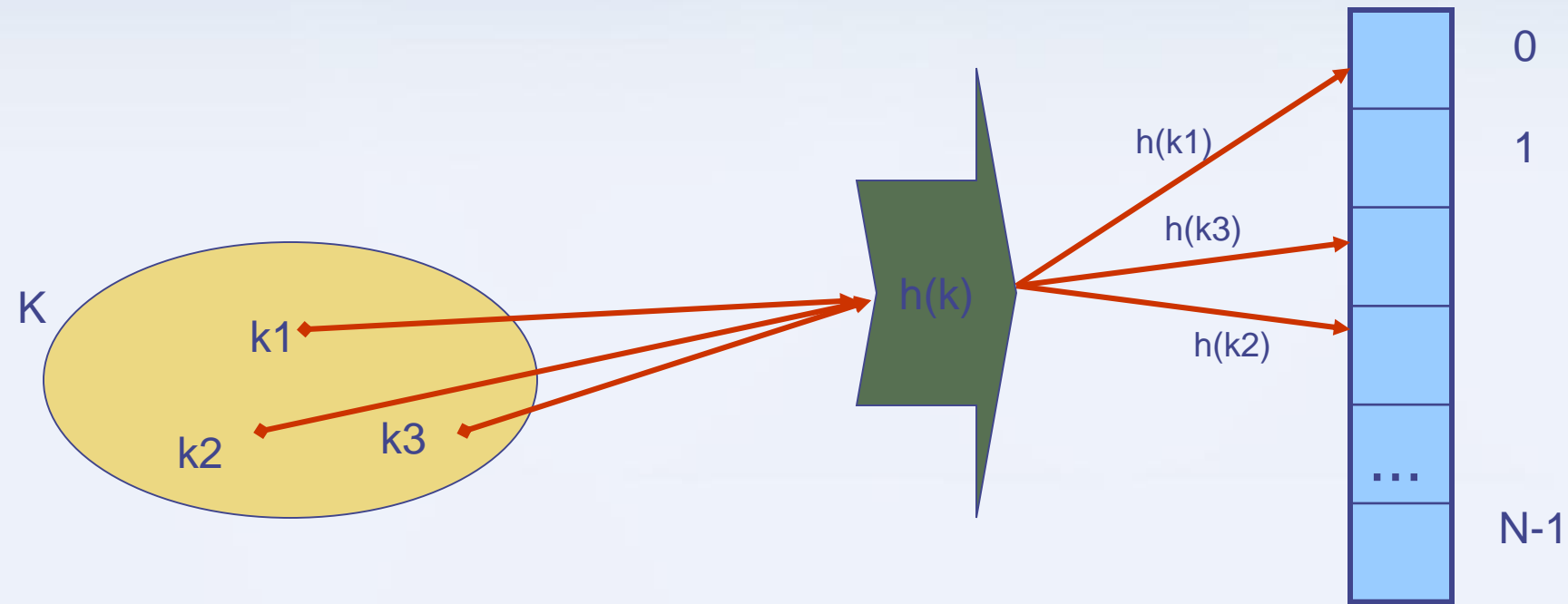
- $h$  được gọi là hàm băm (hash function)
- $K$  là tập giá trị khóa
- $0..m-1$  là bảng địa chỉ (là các số nguyên)
- $m$  là kích thước của bảng

- Yêu cầu khi xây dựng hàm băm:

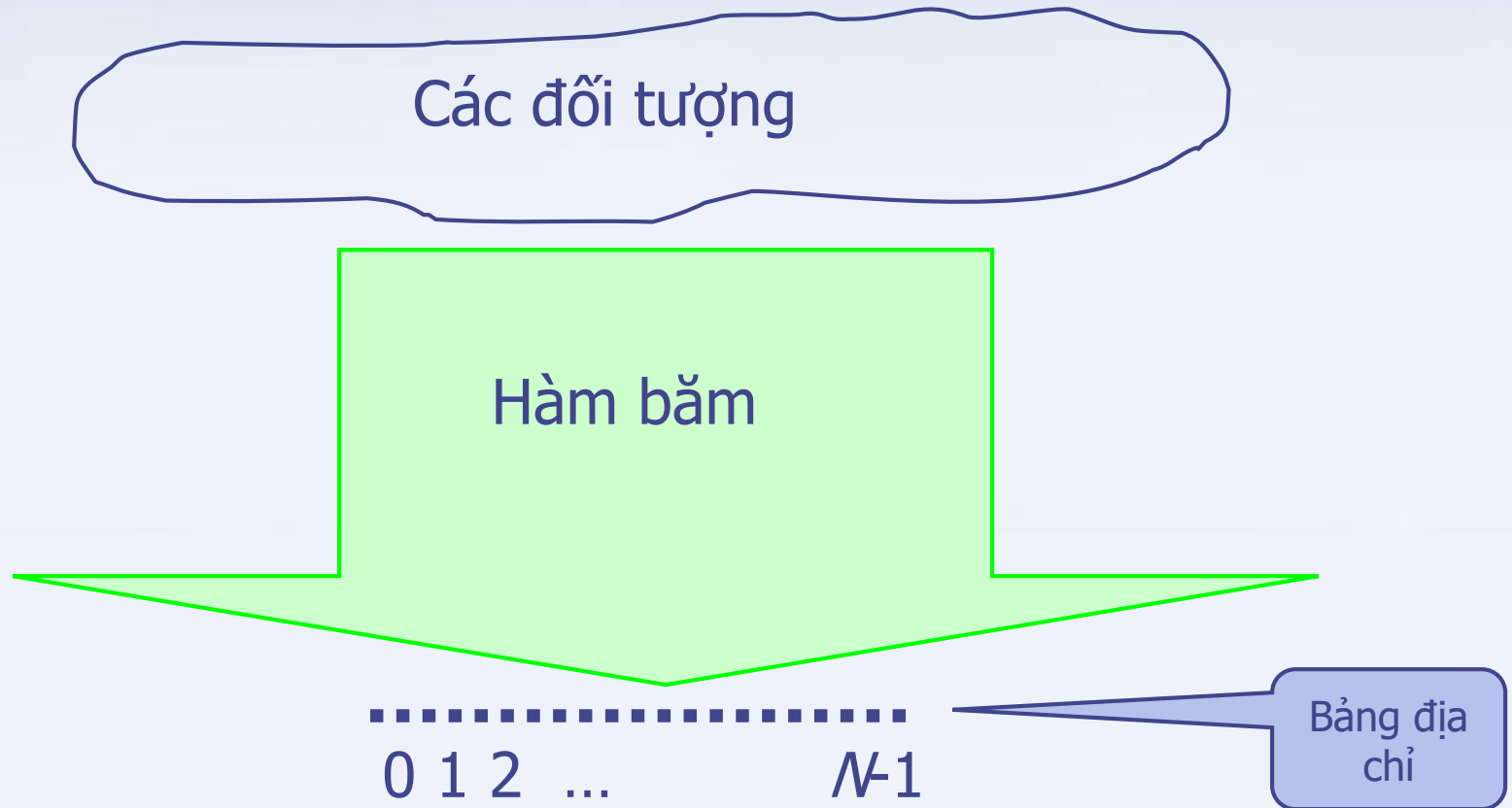
- ◆ Hàm phải rải đều các địa chỉ trên bảng địa chỉ
- ◆ Hàm băm phải được tính toán đơn giản.



# Hình ảnh hàm băm



# Hàm băm



Một hàm băm ánh xạ các đối tượng vào tập các địa chỉ  $[0, N-1]$

## II. Một số phương pháp xây dựng hàm băm

### 1. Phương pháp chia

- Để tính địa chỉ dải của đối tượng ta lấy giá trị khóa chia cho kích thước của bảng. Địa chỉ dải là phần dư của phép chia đó.

$$h(k) = k \% m$$

- Yêu cầu:
  - ◆ hàm  $h$  phải dải đều các đối tượng trên bảng một cách ngẫu nhiên. Để có được điều đó  $h$  phải phụ thuộc vào  $m$ .
  - ◆ Phụ thuộc vào  $m$
  - ◆ Thông thường người ta chọn  $m$  là một số nguyên tố nhỏ hơn gần với (10, 100, 1000, ...) nhất.

## 2. Phương pháp nhân

- ◆ Giá trị khóa được nhân với chính nó sau đó lấy con số bao gồm một số chữ số ở “giữa” kết quả để làm “địa chỉ rải”.
- ◆ Ví dụ:

k	$k^2$	h(k) gồm 3 chữ số
5402	29181604	181 hoặc 816
0367	00134689	134 hoặc 346
1246	01552516	552 hoặc 525

- ◆ Rõ ràng các chữ số ở giữa phụ thuộc vào mọi chữ số của khóa do đó nếu khóa có khác nhau chút ít thì địa chỉ rải vẫn khác nhau nhiều.

### 3. Phương pháp phân đoạn

- ◆ Giá trị khóa được phân ra thành nhiều đoạn bằng nhau
  - Người ta sử dụng hai kỹ thuật phân đoạn sau đây:
  - Tách: Tách các đoạn ra và mỗi đoạn được xếp thành một hàng, dóng lề trái hoặc lề phải.
  - Gấp: Gấp các đoạn lại theo đường biên tương tự như gấp giấy, các chữ rơi vào cùng một chỗ được đặt thành hàng thẳng nhau.

# Ví dụ:

◆ Tách: giả sử có khóa là  $k = 17046329$

$$\begin{array}{r} 329 \\ + 046 \\ \hline 017 \\ \hline 392 \end{array}$$

◆ Gấp:

$$\begin{array}{r} 046 \\ + 923 \\ \hline 710 \\ \hline 1679 \end{array}$$

Chọn 167  
hoặc 679

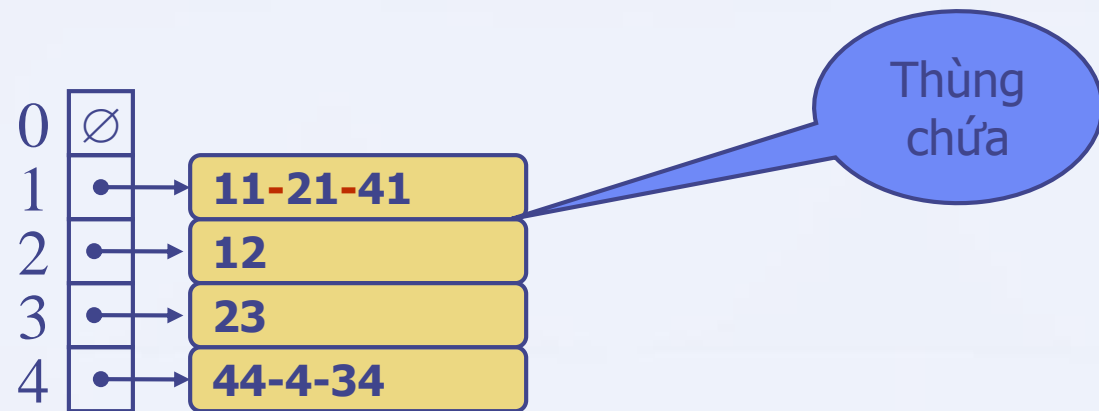


# III. Bảng băm - Hash table

- ◆ Một bảng băm là một cấu trúc dựa trên mảng để lưu trữ các phần tử, mỗi phần tử là một cặp Khóa-Giá trị (key-value)
- ◆ Các thành phần cấu thành lên bảng băm
  - Mảng chứa
  - Mỗi phần tử mảng quản lý một danh sách các phần tử có khóa qua ánh xạ h cho cùng một địa chỉ.
  - Hàm băm  $h(k)$  - Hash function,  $h(k)$ 
    - ◆ Mã băm

Giả sử có hàm  $h(k) = k \% 5$

Có các giá trị: 11, 21, 44, 23, 41, 4, 34, 12



# Độ phức tạp về thời gian

- ◆ Độ phức tạp về thời gian khi đưa một phần tử vào bảng và tìm kiếm một phần tử trong bảng
- ◆ Độ phức tạp về thời gian
  - Trường hợp xấu nhất là  $O(n)$
  - Trường hợp tốt nhất  $O(1)$

# Cấu trúc dữ liệu bảng băm

## ◆ Thuộc tính

- Mảng (mỗi phần tử mảng lưu một danh sách các phần tử)
- N: kích thước mảng

## ◆ Các phương thức

- **Node** \*Add(Key, Object)
- **void** Remove(Key)
- **Node** \*Find(Key)
- **bool** Contains(Key)
- **int** Count()

# Relax

