

## Bài 8.5: Băm lại-rehash

---

- ✓ Các bước thực hiện
- ✓ Triển khai
- ✓ Ví dụ minh họa
- ✓ Bài tập thực hành

# Các bước thực hiện

- B1: Lưu lại kích thước cũ, bảng cũ.
- B2: Tạo kích thước mới gấp đôi kích thước cũ.
- B3: Nếu kích thước mới > số phần tử tối đa
  - B3.1: Nếu kích thước cũ bằng kích thước tối đa thì kết thúc.
  - B3.2: Ngược lại, gán kích thước mới bằng MAX\_SIZE.
- B4: Cấp phát mảng với kích thước mới.
- B5: Tăng số lần sửa đổi bảng lên 1.
- B6: Tính toán lại ngưỡng thực hiện băm lại bảng băm.
- B7: Cho bảng băm tham chiếu đến mảng mới.
- B8: Lặp tới hết bảng cũ thực hiện các bước sau
  - B8.1: Lấy từng phần tử x tại vị trí đang xét trong bảng cũ.
  - B8.2: Tính toán lại chỉ số của phần tử đó trong mảng mới.
  - B8.3: Gán phần tử x vào vị trí mới có được ở bước B8.2.

# Bổ sung các thuộc tính

```
// lớp template mô tả thông tin và các hành động của bảng băm
template<class V> class HashTable {
private:
    size_t capacity; // số phần tử tối đa của bảng băm
    list<Entry<V>>* table; // dữ liệu bảng băm
    const int MAX_SIZE = INT_MAX - 8;
    const float loadFactor = 0.75f;
    int modCount; // số lần sửa đổi cấu trúc bảng băm
    size_t threshold; // ngưỡng băm lại bảng
    size_t count; // số phần tử của bảng băm

public:
    HashTable(int capacity = 10) {
        if (capacity <= 0) {
            capacity = 10;
        }
        this->capacity = capacity;
        this->table = new list<Entry<V>>[capacity];
        modCount = 0;
        threshold = (size_t)(capacity * loadFactor);
    }
}
```

# Mã thật

```
void rehash() {
    int oldCapacity = capacity;
    list<Entry<V>>* oldMap = table;
    int newCapacity = (oldCapacity << 1) + 1;
    if (newCapacity - MAX_SIZE > 0) {
        if (oldCapacity >= MAX_SIZE) {
            return;
        }
        newCapacity = MAX_SIZE;
    }
    list<Entry<V>>* newMap = new list<Entry<V>>[newCapacity];
    modCount++;
    threshold = min((size_t)(newCapacity * loadFactor), MAX_SIZE + 1);
    table = newMap;
    capacity = newCapacity;
    // copy các phần tử từ bảng cũ sang bảng mới
    for (int i = 0; i < oldCapacity; i++)
    {
        for (auto x : oldMap[i])
        {
            size_t index = hashCode(x.hash);
            newMap[index].push_back(x);
        }
    }
}
```

# Nội dung tiếp theo

**Lấy danh sách key, value trong bảng băm**