



TRƯỜNG ĐẠI HỌC THỦY LỢI
KHOA CÔNG NGHỆ THÔNG TIN

Báo cáo bài tập lớn

Môn học: Cấu trúc dữ liệu và giải thuật

Đề tài: Sử dụng bảng băm trong bài toán tra cứu từ điển

Giảng viên: Nguyễn Mạnh Hiền

Thành viên : Dương Đức Nam (59TH2)

Đỗ Thị Hải (59PM1)

Vũ Quốc Tuấn (59PM2)

Nguyễn T.Hồng Nhung (59PM2)

Mục lục

I. Mục tiêu	2
II. Nội dung.....	2
1. Phân tích bài toán, lựa chọn phương pháp tổ chức dữ liệu	2
2. Xử lý dữ liệu đầu vào	2
a. File mục từ.....	2
b. Dữ liệu người dùng nhập vào	3
3. Phân tích thời gian chạy	4
a. Thời gian load file.....	5
b. Thời gian truy xuất dữ liệu	5
c. Thời gian chèn dữ liệu	5
4. Nhận xét.....	5



BÁO CÁO BÀI TẬP LỚN

Môn học: Cấu trúc dữ liệu và giải thuật

Đề tài: Sử dụng bảng băm trong bài toán tra cứu từ điển

I) Mục tiêu

- Sử dụng cấu trúc dữ liệu trong xử lý dữ liệu lớn.
- Phân tích các bước thực hiện.
- Phân tích thời gian chạy khi truy xuất dữ liệu.

II) Nội dung

1. Phân tích bài toán, lựa chọn phương pháp tổ chức dữ liệu

- Bài toán từ điển có số lượng mục từ khá lớn nên ta cần phải có phương pháp tổ chức dữ liệu hợp lý để đạt tốc độ truy xuất nhanh nhất để không ảnh hưởng tới trải nghiệm của người dùng.
- Lựa chọn phương pháp tổ chức dữ liệu:
 - Khi sử dụng danh sách liên kết thì thời gian truy xuất dữ liệu là $O(n)$ vì ta phải duyệt từ đầu danh sách đến hết.
 - Khi sử dụng cây thì ta phải sử dụng cây AVL để đảm bảo thời gian truy xuất là không đổi $O(\log n)$ thì khi chèn dữ liệu vào cây ta phải xoay các nút để cây luôn ở trạng thái cân bằng.
 - Khi sử dụng bảng băm thì thời gian truy xuất dữ liệu $O(1)$.

⇒ Chọn bảng băm để tổ chức dữ liệu. Tuy nhiên, khi bảng đạt trạng thái “khá đầy” thì ta phải tổ chức lại bảng để luôn cho tốc độ truy xuất $O(1)$ với chi phí $O(n)$ nhưng ta không thường xuyên phải làm việc này.

2. Xử lý dữ liệu đầu vào

- Tất cả các file txt đều được xử lý bằng Python và đều đã được build thành file exe kèm thư viện để đảm bảo có thể hoạt động trên cả máy chưa cài Python.

a) File mục từ

- Để có thể phân tích được tốc độ của bảng băm ta cần phải có nguồn dữ liệu tương đối lớn. Vì vậy dữ liệu trong bài sử dụng được trích xuất từ dự án “Open Vietnamese Dictionary Project” được chia sẻ trên SourceForge bởi “peacemoon” (<https://sourceforge.net/projects/ovdp/files/Stardict/English>)
- Vì số lượng mục từ trong file text khá lớn (hơn 83.000 mục từ sau khi đã lọc từ hơn 106.000 mục từ được trích xuất từ “Open Vietnamese Dictionary Project”) vì vậy ta phải chia nhỏ file và dùng thread để giảm thời gian load dữ liệu vào chương trình.

* Chuẩn hóa dữ liệu:



ConvertFileRootToFinally.exe



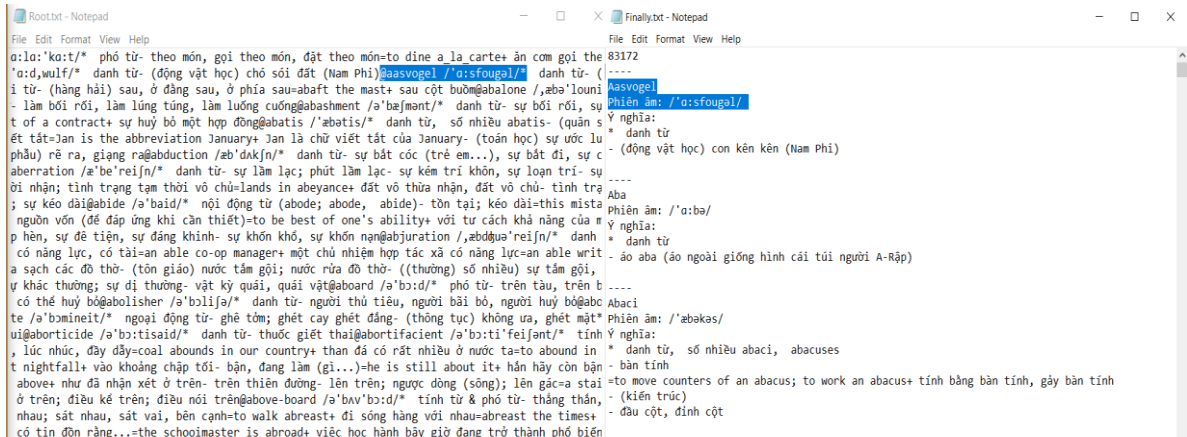
Finally.txt



Root.txt

- File Root.txt là file sau khi được trích xuất.
- Sau khi được chuẩn hóa dữ liệu bằng file ConvertFileRootToFinally.exe sẽ tạo ra file Finally.txt

- Hình ảnh trước và sau khi xử lý file:



Trước

Sau

* Chia nhỏ file:



FileSub



Root



SubFile.exe

- File SubFile.exe có chức năng chia file Finally.txt trong thư mục Root thành các file nhỏ và đưa vào thư mục FileSub.

- Kết quả sau khi chia file:

DictEV1.txt

DictEV2.txt

DictEV3.txt

DictEV4.txt

DictEV5.txt

DictEV6.txt

DictEV7.txt

DictEV8.txt

DictEV9.txt

DictEV10.txt

DictEV11.txt

DictEV12.txt

DictEV13.txt

DictEV14.txt

DictEV15.txt

DictEV16.txt

DictEV17.txt

DictEV18.txt

DictEV19.txt

DictEV20.txt

DictEV21.txt

DictEV22.txt

DictEV23.txt

DictEV24.txt

DictEV25.txt

DictEV26.txt

DictEV27.txt

DictEV28.txt

DictEV29.txt

DictEV30.txt

DictEV31.txt

DictEV32.txt

DictEV33.txt

DictEV34.txt

DictEV35.txt

DictEV36.txt

DictEV37.txt

DictEV38.txt

DictEV39.txt

DictEV40.txt

b) Dữ liệu người dùng nhập vào

- Khi người dùng nhập giá trị lựa chọn vào không phải là số dẫn đến biến lựa chọn có giá trị sai lệch dẫn đến chương trình hoạt động không như mong muốn.
- Trong trường hợp này biến lựa chọn đã nhận giá trị bằng 0 và chương trình đã bị dừng.

```
>> Menu
1. Tra từ điển
2. Thêm mục từ
0. Thoát
Lựa chọn: a

C:\Users\Double D\Desktop\BTL_Dictionary
Press any key to close this window . . .
```

⇒ Vì vậy, ta phải xử lý dữ liệu đầu vào của người dùng:

```
>> Menu
1. Tra từ điển
2. Thêm mục từ
0. Thoát
Lựa chọn: a

Error: Kí tự không hợp lệ !!!
-----Double D-----

Lựa chọn: 2a

Error: Dữ liệu chứa kí tự không hợp lệ ! ! !
-----Double D-----

Lựa chọn: 5

Error: Chỉ được chọn từ 0 -> 2 ! ! !
-----Double D-----
```

* Chuẩn hóa dữ liệu mục từ do người dùng nhập vào:

- Đối với từ tiếng Anh:

```
input:   dOUBLE   d ấ /
Result: Double D
Time of processing: 0s 5ms
```

- Dữ liệu người dùng đưa vào sẽ được xử lý để xóa đi những kí tự đặc biệt, chữ có dấu và chỉ để lại những kí tự trong bảng chữ cái tiếng Anh. Sau đó được chuẩn hóa về dạng Title (chữ cái đầu của mỗi từ viết hoa).

- Đối với từ tiếng Việt:

```
input: - KIỂM TRÁ
Result: - Kiểm tra
Time of processing: 0s 4ms
```

- Cũng giống như với từ tiếng Anh, tuy nhiên dữ liệu sẽ chỉ chuẩn hóa về dạng Capitalize (chữ cái đầu dòng viết hoa) và không xóa bất kỳ kí tự nào.

3. Phân tích thời gian chạy

a) Thời gian load file:

- Thời gian load file đã được giảm 96% khi chia nhỏ file và sử dụng thread.

```
Time load: 20m 18s 612ms
>> Menu
1. Tra từ điển
2. Thêm mục từ
0. Thoát
Lựa chọn: _
```

Trước

```
Time load: 46s 530ms
>> Menu
1. Tra từ điển
2. Thêm mục từ
0. Thoát
Lựa chọn: _
```

Sau

b) Thời gian truy xuất dữ liệu:

```
>> Tra từ (kí tự không hợp lệ sẽ bị xóa)
Nhập key: fresher

                Nghĩa của từ Fresher là:

Phiên âm: /'freʃə/
Ý nghĩa:
* danh từ
- học sinh đại học năm thứ nhất ((cũng) freshman)

Time search: 0s 12ms
```

c) Thời gian chèn dữ liệu:

```
>> Thêm từ (kí tự không hợp lệ sẽ bị xóa)
Nhập key: DeMo
Nhập nghĩa của từ Demo:
Line 1 (0 để hủy): Kiểm tra thời gian chèn dữ liệu
Line 2 (0 để dừng): 0

                Đã thêm từ Demo thành công ! ! !
                -----Double D-----

Time insert: 0s 9ms
```

4. Nhận xét

- Bảng băm có tốc độ rất nhanh, thời gian chèn – sửa – xóa chỉ mất khoảng thời gian rất ngắn, phù hợp với các bài toán cần tốc độ truy xuất nhanh. Tuy nhiên, bảng băm không phù hợp với các bài toán sắp xếp và luôn phải đảm bảo khối lượng dữ liệu trong bảng ở dưới hệ số tải λ từ đó gây lãng phí $1 - \lambda$ bộ nhớ.