

TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



Niên luận Cơ sở ngành Kỹ thuật phần mềm

CÀI ĐẶT TỰ ĐIỂN ANH – VIỆT
SỬ DỤNG THUẬT TOÁN TRÊN BẢNG BẮM

Giảng viên hướng dẫn

Trương Thị Thanh Tuyền
MSCB: 1068

Sinh viên thực hiện

Đào Minh Trung Thuận
MSSV: B1704855
Lớp: DI1796A2 - Kỹ thuật phần mềm A2

Học kỳ 2, năm học 2019-2020

Đề tài:

Chương trình Tự điển Anh – Việt

Giáo viên hướng dẫn:

Ths. Trương Thị Thanh Tuyền

ĐÁNH GIÁ KẾT QUẢ THỰC HIỆN NIÊN LUẬN CƠ SỞ NGÀNH KTPM

(Học kỳ: 02, Năm học 2019-2020)

TÊN ĐỀ TÀI: CHƯƠNG TRÌNH TỰ ĐIỂN ANH – VIỆT

GIÁO VIÊN HƯỚNG DẪN:

STT	HỌ VÀ TÊN	MSCB
1	Trương Thị Thanh Tuyền	1068

SINH VIÊN THỰC HIỆN:

HỌ VÀ TÊN	MSSV	THƯỞNG (Tối đa 1,0 điểm)	ĐIỂM (thang điểm 10)
Đào Minh Trung Thuận	B1704855		

I. HÌNH THỨC (0,5 điểm)

Bìa (tối đa 0,25 điểm)

- Đầy đủ các thông tin
- Đúng định dạng

Bố cục (tối đa 0.25 điểm)

- Trang đánh giá kết quả thực hiện niên luận 1
- Mục lục: cấu trúc chương, mục và tiểu mục
- Phụ lục (nếu có)
- Tài liệu tham khảo

II. NỘI DUNG (3,5 điểm)

Giới thiệu (tối đa 0,5 điểm)

- Mô tả bài toán (0,25 điểm)
- Mục tiêu cần đạt, hướng giải quyết (0,25 điểm)

Lý thuyết (tối đa 0,5 điểm)

- Các khái niệm sử dụng trong chương trình (0,25 điểm)

<ul style="list-style-type: none"> Kết quả vận dụng lý thuyết trong đề tài (0,25 điểm) 	
Ứng dụng (tối đa 2,0 điểm)	
<ul style="list-style-type: none"> Phân tích yêu cầu, xây dựng các cấu trúc dữ liệu (0.5 điểm) Sơ đồ chức năng, lưu đồ giải thuật giải quyết vấn đề (1.0 điểm) Giới thiệu sử dụng chương trình (0,5 điểm) 	
Kết luận (tối đa 0,5 điểm)	
<ul style="list-style-type: none"> Nhận xét kết quả đạt được Hạn chế Hướng phát triển 	
III. CHƯƠNG TRÌNH DEMO (5,0 điểm)	
Giao diện thân thiện với người dùng (1,0 điểm)	
Hướng dẫn sử dụng (0,5 điểm)	
Kết quả thực hiện đúng với kết quả của phần ứng dụng (tối đa 3,5 điểm) <ul style="list-style-type: none"> Kết quả đúng (2,0 điểm) Cách thức thực hiện hợp lý (1,0 điểm) Chức năng bổ sung, sáng tạo (0,5 điểm) 	

Cần Thơ, ngày tháng năm 2020

GIÁO VIÊN CHẤM

Trương Thị Thanh Tuyền

Đề tài:

Chương trình Tự điển Anh – Việt

Giáo viên hướng dẫn:

Ths. Trương Thị Thanh Tuyền

LỜI CẢM ƠN

Lời đầu tiên, cho em xin dành lời cảm ơn chân thành nhất gửi đến giáo viên hướng dẫn, Ths. Trương Thị Thanh Tuyền đã gợi ý và đề xuất cho em đề tài này, nhờ đó em đã tích lũy được nhiều kiến thức và kinh nghiệm hơn trong quá trình thực hiện đề tài. Và trên hết, con cũng xin gửi lời cảm ơn đến cha mẹ đã luôn luôn bên cạnh, động viên, nhắc nhở, khuyến khích và là hậu phương vững chắc cho con trên con đường học tập. Em xin cảm ơn các anh chị khoá trên trong Câu lạc bộ Tin học Khoa Công nghệ thông tin và Truyền thông đã cho em lời khuyên trong quá trình thực hiện khi em gặp khó khăn.

Cần Thơ, ngày 08 tháng 05 năm 2020

Người viết

Đào Minh Trung Thuận

MỤC LỤC

1	TỔNG QUAN	8
1.1	MÔ TẢ BÀI TOÁN	8
1.2	MỤC TIÊU CẦN ĐẠT ĐƯỢC	8
1.3	HƯỚNG GIẢI QUYẾT VÀ KẾ HOẠCH THỰC HIỆN	8
1.3.1	Hướng giải quyết.....	8
1.3.2	Kế hoạch thực hiện.....	9
2	CƠ SỞ LÝ THUYẾT	10
2.1	CÁC KHÁI NIỆM.....	10
2.1.1	Kiểu dữ liệu trừu tượng Danh sách (List)	10
2.1.2	Kiểu dữ liệu trừu tượng Tự điển (Dictionary)	11
2.1.3	Cấu trúc Bảng băm (HashTable).....	12
2.1.4	Bảng băm mở.....	12
2.1.5	Các phương pháp xác định hàm băm.....	12
2.2	KẾT QUẢ VẬN DỤNG LÝ THUYẾT	13
3	GIẢI QUYẾT VẤN ĐỀ.....	13
3.1	Phân tích yêu cầu bài toán, xây dựng cấu trúc dữ liệu.....	13
3.1.1	Phân tích yêu cầu bài toán.....	13
3.1.2	Xây dựng Cấu trúc dữ liệu.....	14
3.2	SƠ ĐỒ CHỨC NĂNG, LƯU ĐỒ GIẢI THUẬT	16
3.2.1	Sơ đồ chức năng	16
3.3	CÁCH THỨC XÂY DỰNG CHƯƠNG TRÌNH.....	18
3.3.1	Cấu trúc dự án.....	18
3.3.2	Xây dựng chương trình chính.....	19
4	KẾT LUẬN VÀ ĐÁNH GIÁ.....	26
4.1	KẾT QUẢ ĐẠT ĐƯỢC	26
4.2	HẠN CHẾ, NGUYÊN NHÂN	26
4.3	HƯỚNG PHÁT TRIỂN ĐỀ TÀI	26
5	PHẦN PHỤ LỤC.....	27
5.1	Hướng dẫn cài đặt chương trình.....	27
5.2	Hướng dẫn sử dụng	27
6	TÀI LIỆU THAM KHẢO	34

MỤC LỤC HÌNH

Hình 5.1: Giao diện chính của chương trình.....	28
Hình 5.2: Chức năng tra cứu từ	28
Hình 5.3: Kết quả tra cứu từ nếu tìm thấy từ.....	29
Hình 5.4: Thông báo không tìm thấy từ.....	29
Hình 5.5: Nhập từ mới cần thêm	30
Hình 5.6: Kết quả thêm từ mới	30
Hình 5.7: Thông báo thêm từ không thành công.....	31
Hình 5.8: Nhập từ cần xoá.....	31
Hình 5.9: Thông báo xoá từ thành công.....	32
Hình 5.10: Thông báo xoá từ không thành công	32
Hình 5.11: Chức năng xem lịch sử tra cứu.....	33
Hình 5.12: Lịch sử trống.....	33
Hình 5.13: Lưu lịch sử trước khi thoát	34

MỤC LỤC SƠ ĐỒ

Sơ đồ 2.1: Sơ đồ Cài đặt danh sách bằng mảng.....	10
Sơ đồ 2.2: Sơ đồ Danh sách liên kết đơn.....	11
Sơ đồ 2.3: Sơ đồ Cài đặt Danh sách liên kết đôi	11
Sơ đồ 2.4: Sơ đồ cài đặt Tự điển bằng Bảng băm mở	12
Sơ đồ 3.1: Sơ đồ DFD mức ngữ cảnh.....	16
Sơ đồ 3.2: Sơ đồ DFD mức 0	17
Sơ đồ 3.3: Sơ đồ DFD mức 1	18
Sơ đồ 3.4: Sơ đồ cấu trúc dự án.....	19
Sơ đồ 3.5: Sơ đồ hoạt động chương trình chính	20
Sơ đồ 3.6: Sơ đồ hoạt động chức năng tra cứu từ.....	21
Sơ đồ 3.7: Sơ đồ hoạt động chức năng thêm từ mới	22
Sơ đồ 3.8: Sơ đồ hoạt động chức năng sửa từ.....	23

Sơ đồ 3.9: Sơ đồ hoạt động chức năng xoá từ..... 24

Sơ đồ 3.10: Sơ đồ hoạt động chức năng xem lịch sử tra cứu 25

MỤC LỤC BẢNG

Bảng 1.1: Bảng kế hoạch thực hiện9

Bảng 5.25: Cú pháp lệnh đi đến đường dẫn..... 27

Bảng 5.26: Ví dụ đi đến đường dẫn chương trình..... 27

Bảng 5.27: Lệnh chạy chương trình..... 27

1 TỔNG QUAN

1.1 MÔ TẢ BÀI TOÁN

Xây dựng và cài đặt chương trình tự điển Anh – Việt với yêu cầu kỹ thuật sử dụng thuật toán trên bảng băm để thực hiện các thao tác cơ bản:

- Tra cứu từ trong tự điển.
- Thêm từ mới và tự điển.
- Sửa từ đã có trong tự điển
- Xoá từ khỏi tự điển.
- Xem lại từ đã tra cứu (lịch sử tra cứu từ).

1.2 MỤC TIÊU CẦN ĐẠT ĐƯỢC

Với nội dung yêu cầu của đề tài, mục tiêu cần đạt được bao gồm:

- Nắm vững được các khái niệm về kiểu dữ liệu, kiểu dữ liệu trừu tượng.
- Nắm vững và cài đặt được các kiểu dữ liệu trừu tượng, cụ thể gồm có: Bảng băm, Danh sách liên kết.
- Vận dụng kiến thức để giải quyết bài toán xây dựng chương trình tự điển Anh – Việt với các chức năng theo yêu cầu.

Ngoài những mục tiêu yêu cầu về kỹ năng chuyên môn, còn có các mục tiêu khác cần phải đạt được:

- Biết cách hiểu và giải quyết vấn đề yêu cầu của đề tài đã đặt ra.
- Biết cách lên kế hoạch và đề sắp xếp thời gian cho quá trình thực hiện đề tài.
- Rèn luyện được các kỹ năng viết báo cáo, tài liệu, kỹ năng thuyết trình.

1.3 HƯỚNG GIẢI QUYẾT VÀ KẾ HOẠCH THỰC HIỆN

1.3.1 Hướng giải quyết

Để có thể thực hiện đề tài một cách tốt nhất, cần tìm hiểu lại kỹ các kiến thức về Cấu trúc dữ liệu, các kiểu dữ liệu và kiểu dữ liệu trừu tượng, cách cài đặt cũng như thuật toán đã được học ở môn Cấu trúc dữ liệu và Giải thuật.

Vì đề tài yêu cầu về kỹ thuật cài đặt cũng như vận dụng thuật toán, nên hướng giải quyết cho đề tài sẽ là sử dụng ngôn ngữ lập trình C/C++ làm ngôn ngữ để phát triển. Xây dựng cài đặt các kiểu dữ liệu trừu tượng Danh sách liên kết, Bảng băm và từ đó áp dụng để xây dựng chương trình.

1.3.2 Kế hoạch thực hiện

STT	Thời gian	Nội dung
1	Tuần 2	Tập trung, nghe giảng viên hướng dẫn, nhận đề tài.
2	Tuần 3	Đăng ký đề tài qua Email.
3	Tuần 4	Tìm hiểu tài liệu, lên phương án, hướng giải quyết đề tài.
4	Tuần 5	Viết tài liệu đặt tả yêu cầu bài toán và hướng giải quyết.
5	Tuần 6	Chỉnh sửa tài liệu đặt tả, và nộp cho giảng viên hướng dẫn.
6	Tuần 7	Viết cơ sở lý thuyết, khái niệm và vận dụng.
7	Tuần 8	Viết tài liệu phân tích yêu cầu, tài liệu xây dựng kiểu dữ liệu.
8	Tuần 9	Viết tài liệu sơ đồ chức năng, vẽ lưu đồ giải thuật.
9	Tuần 10	Nộp tài liệu thiết kế giải thuật và dữ liệu sử dụng.
10	Tuần 11	Viết tài liệu cách thức xây dựng chương trình.
11	Tuần 12	Cài đặt kiểu dữ liệu trừu tượng Danh sách liên kết, Bảng băm.
12	Tuần 13	Cài đặt chương trình, xây dựng giao diện.
13	Tuần 14	Kiểm thử, điều chỉnh tài liệu, viết tài liệu hướng sử dụng và cài đặt chương trình cho người sử dụng.
14	Tuần 15	Hoàn thiện chương trình, kiểm thử lần cuối và hoàn thiện tài liệu báo cáo.
15	Tuần 17	Báo cáo lần 1.
16	Tuần 18	Chỉnh sửa lại chương trình.
17	Tuần 19	Chỉnh sửa lại tài liệu báo cáo.
18	Tuần 21	Báo cáo lần 2.

Bảng 1.1: Bảng kế hoạch thực hiện

2 CƠ SỞ LÝ THUYẾT

2.1 CÁC KHÁI NIỆM

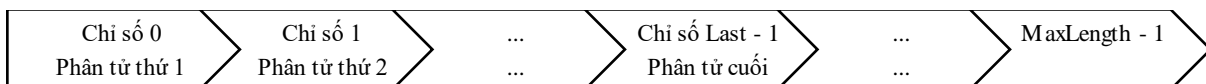
2.1.1 Kiểu dữ liệu trừu tượng Danh sách (List)

2.1.1.1 Khái niệm danh sách

Mô hình toán học của danh sách là một tập hợp hữu hạn các phần tử có cùng một kiểu, mà tổng quát ta gọi là kiểu phần tử (ElementType). Ta biểu diễn danh sách như là một chuỗi các phần tử của nó: $a_1, a_2, \dots, a_{n-1}, a_n$ (với $n \geq 0$). Nếu $n = 0$ ta nói danh sách rỗng (empty list). Nếu $n > 0$ ta gọi a_1 là phần tử đầu tiên và a_n là phần tử cuối cùng của danh sách. Số phần tử của danh sách ta gọi là độ dài của danh sách.

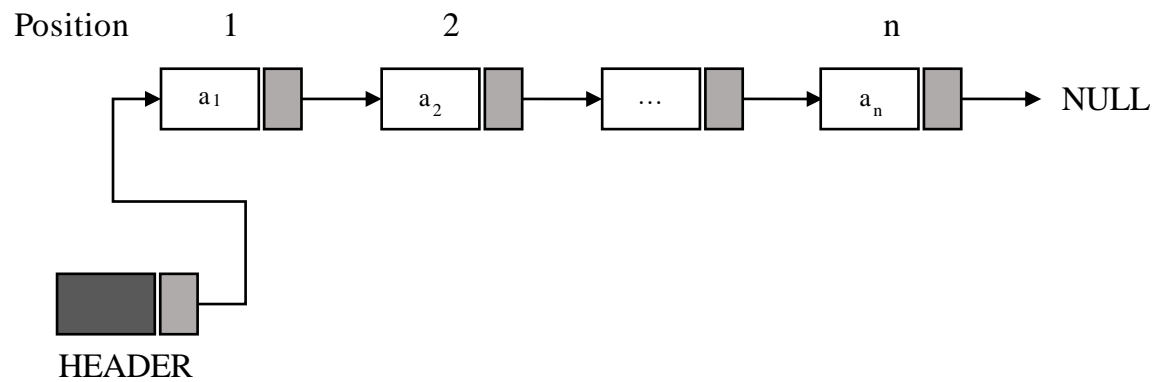
2.1.1.2 Cài đặt Danh sách

- **Cài đặt danh sách bằng mảng (danh sách đặc):** Dùng một mảng để lưu giữ liên tiếp các phần tử của danh sách từ vị trí đầu tiên của mảng. Với cách cài đặt này, ta phải ước lượng số phần tử của danh sách để khai báo số phần tử của mảng cho thích hợp. Dễ thấy rằng số phần tử của mảng phải được khai báo không ít hơn số phần tử của danh sách. Nói chung là mảng còn thừa một số chỗ trống. Mặt khác ta phải lưu giữ độ dài hiện tại của danh sách, độ dài này cho biết danh sách có bao nhiêu phần tử và cho biết phần nào của mảng còn trống. Ta định nghĩa vị trí của một phần tử trong danh sách là chỉ số của mảng tại vị trí lưu trữ phần tử đó + 1 (vì phần tử đầu tiên trong mảng là chỉ số 0).



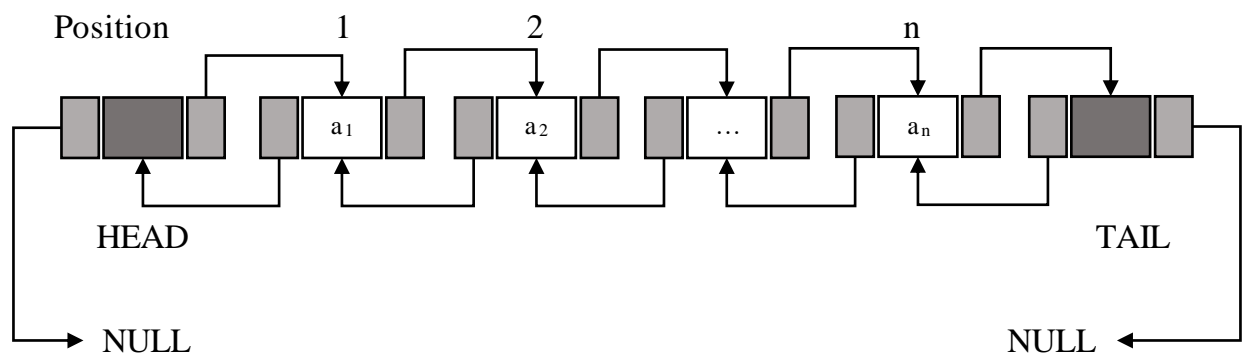
Sơ đồ 2.1: Sơ đồ Cài đặt danh sách bằng mảng

- **Cài đặt danh sách bằng con trỏ (danh sách liên kết, LinkedList):** Dùng con trỏ để liên kết các ô chứa các phần tử. Trong cách cài đặt này các phần tử của danh sách được lưu trữ trong các ô, mỗi ô có thể chỉ đến ô chứa phần tử kế tiếp trong danh sách. Để quản lý danh sách ta chỉ cần một biến giữ địa chỉ ô chứa phần tử đầu tiên của danh sách, tức là một con trỏ trỏ đến phần tử đầu tiên trong danh sách. Biến này gọi là chỉ điểm đầu danh sách (Header). Để đơn giản hóa vấn đề, trong chi tiết cài đặt, Header là một biến cùng kiểu với các ô chứa các phần tử của danh sách và nó có thể được cấp phát ô nhớ y như một ô chứa phần tử của danh sách. Tuy nhiên Header là một ô đặc biệt nên nó không chứa phần tử nào của danh sách, trường dữ liệu của ô này là rỗng, chỉ có trường con trỏ Next trỏ tới ô chứa phần tử đầu tiên thật sự của danh sách. Nếu danh sách rỗng thì Header \rightarrow next trỏ tới NULL. Việc cấp phát ô nhớ cho Header như là một ô chứa dữ liệu bình thường nhằm tăng tính đơn giản của các giải thuật thêm, xóa các phần tử trong danh sách. Ta có Danh sách liên kết đơn



Sơ đồ 2.2: Sơ đồ Danh sách liên kết đơn

Từ đó ta mở rộng cấu trúc Danh sách bằng các phần tử gồm 2 con trỏ Next và Previous để thuận tiện cho việc di chuyển tới và lùi giữa các phần tử trong danh sách. Lúc này, ta sẽ có thêm một ô đặt biệt nữa là Tail đặt ở cuối danh sách. Ta có danh sách Liên kết đôi



Sơ đồ 2.3: Sơ đồ Cài đặt Danh sách liên kết đôi

2.1.2 Kiểu dữ liệu trừu tượng Tự điển (Dictionary)

Từ điển là một kiểu dữ liệu trừu tượng tập hợp đặc biệt, trong đó chúng ta chỉ quan tâm đến các phép toán thêm phần tử (InsertSet), xóa phần tử (DeleteSet), lấy phần tử (Member) và tạo Từ điển trống (MakeNullSet). Sở dĩ chúng ta nghiên cứu từ điển là do trong nhiều ứng dụng không sử dụng đến các phép toán hợp, giao, hiệu của hai tập hợp. Ngược lại ta cần một cấu trúc làm sao cho việc tìm kiếm, thêm và bớt phần tử có phần hiệu quả nhất gọi là từ điển. Chúng ta cũng chấp nhận MakeNullSet như là phép khởi tạo cấu trúc.

Cài đặt Từ điển: Thực chất việc cài đặt từ điển bằng mảng hoàn toàn giống với việc cài đặt danh sách đặc không có thứ tự. Có rất nhiều kỹ thuật cài đặt Từ điển: cài đặt Từ điển bằng mảng, cài đặt bằng Danh sách liên kết hoặc có thể áp dụng kỹ thuật xây dựng bảng băm để cài đặt...

2.1.3 Cấu trúc Bảng băm (HashTable)

2.1.3.1 Kỹ thuật băm (Hashing)

Kỹ thuật băm hay phương pháp băm là một phương pháp tính toán trực tiếp vị trí của mảng lưu trữ một phần tử của tập hợp dựa vào giá trị khoá của phần tử này, tức là tính toán “địa chỉ” trực tiếp từ khoá.

Giả sử, ta có một mảng gồm n phần tử được đánh số từ 0 đến $n - 1$ và một tập hợp A muốn lưu trữ vào trong mảng này. Như vậy với mỗi phần tử $x \in A$ ta phải dựa vào khoá này để suy ra một giá trị số nguyên thuộc khoảng $[0, n - 1]$ là vị trí cất giữ khoá này. Nói cách khác, ta chọn một hàm $h(x)$, khi tìm kiếm phần tử x ta chỉ cần tính $h(x)$:

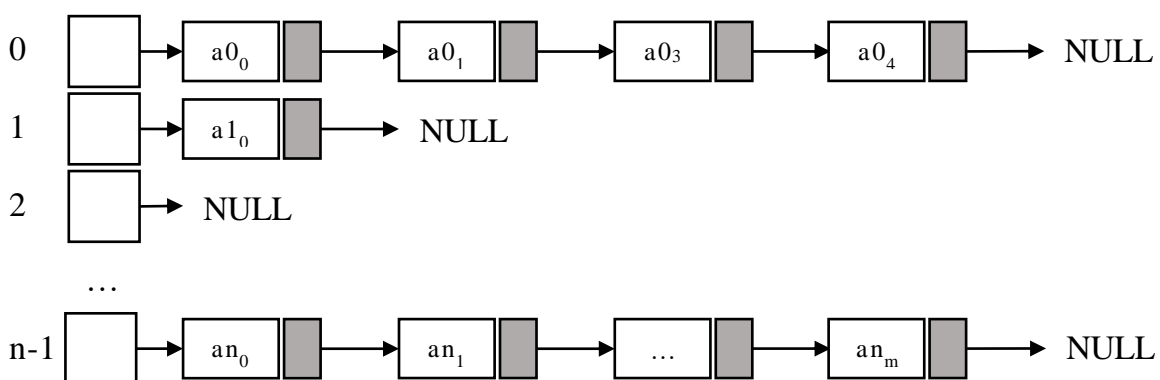
$$h: A \rightarrow 0..n-1$$

$$x \rightarrow h(x)$$

2.1.4 Bảng băm mở

Bảng băm mở là một ý tưởng đơn giản cho việc giải quyết đụng độ là tổ chức thêm một danh sách cho từng tập hợp các khoá có cùng giá trị băm. Tức là chia một tập hợp đang xét thành một số hữu hạn các lớp con bên trong. Nếu bảng băm có n phần tử được đánh số từ 0 đến $n - 1$ và h là hàm băm thì lớp thứ k là một danh sách gồm tất cả các phần tử x sao cho $h(x) = k$.

Tổ chức danh sách có thể theo bất kỳ cách nào, nhưng do số lượng các khoá trong mỗi lớp là không biết trước, nên mỗi lớp ta tổ chức một Danh sách liên kết cho các phần tử của có cùng kết quả băm khoá. Mỗi lớp như vậy gọi là một lô chứa (Bucket).



Sơ đồ 2.4: Sơ đồ cài đặt Tự điển bằng Bảng băm mở

2.1.5 Các phương pháp xác định hàm băm

Có rất nhiều cách để xác định hàm băm, ví dụ như:

- Phương pháp đồng dư (phương pháp chia)

- Phương pháp nhân
- Phương pháp tách

2.2 KẾT QUẢ VẬN DỤNG LÝ THUYẾT

Vận dụng các kiến thức trong Cấu trúc dữ liệu, ta mô hình hoá bài toán xây dựng chương trình Tự điển Anh – Việt dựa trong cấu trúc Kiểu dữ liệu trừu tượng Tự điển là một tập hợp gồm các phần tử từ (Word) gồm hai thành phần là trường từ tiếng Anh và trường nghĩa tiếng Việt. Áp dụng kỹ thuật cài đặt Bảng băm để xây dựng Tự điển. Sử dụng Danh sách liên kết làm lịch sử tra cứu.

3 GIẢI QUYẾT VẤN ĐỀ

3.1 PHÂN TÍCH YÊU CẦU BÀI TOÁN, XÂY DỰNG CẤU TRÚC DỮ LIỆU

3.1.1 Phân tích yêu cầu bài toán

Với yêu cầu từ bài toán đặt ra cụ thể là xây dựng Tự điển Anh – Việt bằng Bảng băm, ta có các phân tích như sau:

- Đối với kỹ thuật băm để xây dựng Bảng băm, để đơn giản ta sử dụng phương pháp chia, tức là vị trí được xác định bằng cách lấy giá trị của Khoá đem chia cho số Bucket của Tự điển và lấy phần dư của nó, ta sẽ có hàm băm với công thức: $h(x) = x \bmod n$ (với số Bucket $n = 100$).
- Đối với từ (Word) trong Tự điển, ta sẽ lấy trường từ tiếng Anh làm khoá đại diện, để thuận tiện cho việc băm, ta sẽ chuyển đổi từ tiếng Anh dưới dạng chuỗi (String) trở thành dạng số nguyên dương bằng cách lấy tổng mã Ascii của các ký tự của từ. Mỗi Bucket sẽ gồm một danh sách các từ có cùng giá trị băm, ta sẽ dùng Danh sách liên kết đơn để xây dựng danh sách đó.
- Dữ liệu Word sẽ được lưu vào một tệp dữ liệu (Data file) dưới dạng file text (phần mở rộng của file là .txt). Khi chương trình chạy sẽ tự động nạp dữ liệu bằng cách đọc file. Các chức năng như thêm, xoá từ sẽ thay đổi lên file.
- Ngoài ra, để lưu lại lịch sử tra cứu từ, ta sẽ sử dụng Danh sách liên kết đôi để làm lịch sử. Bằng cách đó, ta có thể xem lại từ đã tra cứu một cách nhanh hơn thay vì sử dụng Danh sách liên kết đơn.
- Chương trình sẽ lấy ngôn ngữ lập trình C/C++ làm ngôn ngữ phát triển, các mã lệnh (code) sẽ được phân chia và quản lý theo Kiểu dữ liệu thành các header include file trong C/C++.

Ngoài ra, còn phải có các yêu cầu như sau:

- Chương trình được gọi là hoàn thiện khi đáp ứng đầy đủ các năng tra cứu, thêm từ, sửa từ, xoá từ, xem lịch sử tra cứu.

- Chương trình được xây dựng hướng đến giao diện, người sử dụng thao tác dễ dàng nhất dù chương trình chạy trên nền Console.
- Chương trình chạy mượt, không quá chậm, không có lỗi.
- Dễ dàng bảo trì, nâng cấp sau này và thiết kế sao cho dễ gỡ lỗi nếu có lỗi xảy ra.

3.1.2 Xây dựng Cấu trúc dữ liệu

Ta sẽ xây dựng Cấu trúc dữ liệu, Kiểu dữ liệu mang tính gần tương đồng nhất với Lập trình hướng đối tượng. Một Kiểu dữ liệu giống với một Lớp, gồm có hàm tạo (Create), hàm huỷ (Destroy) và các hàm và thủ tục khác của nó. Đối với các kiểu dữ liệu trừu tượng chỉ cài đặt các hàm và phép tính cần thiết cho chương trình.

3.1.2.1 Kiểu Chuỗi ký tự (String)

Kiểu String là một dãy các ký tự Char, ta sẽ dùng con trỏ để xây dựng kiểu String vì đối với bài toán, ta không biết chính xác số ký tự của một chuỗi là bao nhiêu.

Các hàm trong String gồm có:

- Hàm khởi tạo chuỗi rỗng.
- Hàm huỷ chuỗi.
- Hàm kiểm tra hai chuỗi.
- Hàm thêm ký tự vào cuối chuỗi.
- Hàm nhận đầu vào là chuỗi từ bàn phím.

3.1.2.2 Kiểu Từ (Word)

Kiểu Word đại diện cho từ có cấu trúc gồm hai thành phần là trường từ tiếng Anh và trường nghĩa tiếng Việt dựa trên kiểu String vừa định nghĩa.

3.1.2.3 Kiểu Phần tử Tự điển (Element)

Để tương quan hơn trong thiết kế, ta xây dựng Kiểu Phần tử cho Tự điển sẽ là kiểu Word, sử dụng trường từ tiếng Anh để làm khoá đại diện cho phần tử.

Như vậy ta sẽ có một hàm của kiểu Element là Hàm lấy khoá.

3.1.2.4 Kiểu Danh sách liên kết (Linked List)

Cài đặt Danh sách dạng Danh sách liên kết đôi, trước hết ta cài đặt kiểu Nút liên kết (NodeLinkedList) cho Danh sách liên kết và các hàm của kiểu Node. Định nghĩa kiểu Node có cấu trúc gồm ba thành phần là:

- Phần dữ liệu.
- Phần con trỏ chỉ tới.
- Phần con trỏ chỉ lùi.

Các hàm của kiểu NodeLinkedList này gồm có hai hàm:

- Hàm khởi tạo.
- Hàm huỷ.

Sau đó, định nghĩa kiểu LinkedList với cấu trúc gồm hai thành phần là trường Head và trường Tail dựa trên kiểu NodeLinkedList.

Các hàm của kiểu LinkedList cũng gồm có hai hàm:

- Hàm khởi tạo.
- Hàm huỷ.

3.1.2.5 Kiểu Bảng băm (Hash Table)

Thiết kế thuật toán băm bằng phương pháp chia với công thức là: $h(x) = x \% 100$.

Sử dụng Bảng băm mở cho cài đặt Tự điển, cho nên ta thiết kết Node dưới dạng Node cho danh sách liên kết đơn, gồm một trường dữ liệu data và một trường con trỏ tới Node kế tiếp.

Các hàm của NodeHashTable gồm có:

- Hàm khởi tạo.
- Hàm huỷ.

Định nghĩa kiểu HashTable là một tập hợp các Danh sách liên kết gồm **100 phần tử** (số phần tử bằng với số chia trong hàm băm). Mỗi phần tử này được gọi là Bucket. Và ta cũng có các hàm cho kiểu HashTable:

- Hàm khởi tạo.
- Hàm huỷ.
- Hàm lấy Node.
- Hàm duyệt qua các Node.
- Hàm thêm Node mới.
- Hàm xoá Node.

3.1.2.6 Kiểu Lịch sử (History)

Sử dụng Danh sách liên kết đôi để cài đặt History cho dễ dàng việc di chuyển tới lùi thông qua liên kết của các Node trong LinkedList. Ngoài ra ta định nghĩa thêm một kiểu con trỏ để truy xuất History.

Các hàm của History gồm:

- Hàm khởi tạo.

- Hàm huỷ.
- Hàm thêm lịch sử.
- Hàm lấy dữ liệu lịch sử.
- Hàm di chuyển con trỏ cuộn tới.
- Hàm di chuyển con trỏ cuộn lùi.

3.1.2.7 Kiểu Tự điển (Dictionary)

Sau khi đã cài đặt hết các kiểu dữ liệu, ta chỉ cần định nghĩa kiểu Tự điển là kiểu HashTable. Và cũng xây dựng thêm các hàm cho Dictionary:

- Hàm khởi tạo.
- Hàm huỷ.
- Hàm viết dữ liệu ra file data.txt.
- Hàm đọc file data.txt và nạp dữ liệu vào từ điển.
- Hàm thêm từ mới.
- Hàm sửa từ đã có.
- Hàm xoá từ.
- Hàm tra cứu từ.

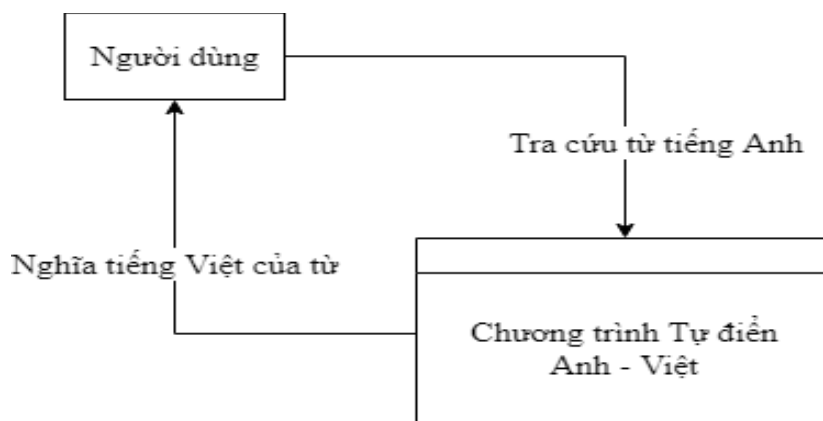
3.2 SƠ ĐỒ CHỨC NĂNG, LƯU ĐỒ GIẢI THUẬT

3.2.1 Sơ đồ chức năng

Để khái quát hơn về chương trình và chức năng của nó, ta sử dụng Sơ đồ luồng dữ liệu (Data Flow Diagram) để minh hoạ.

3.2.1.1 Mức ngữ cảnh (Context level)

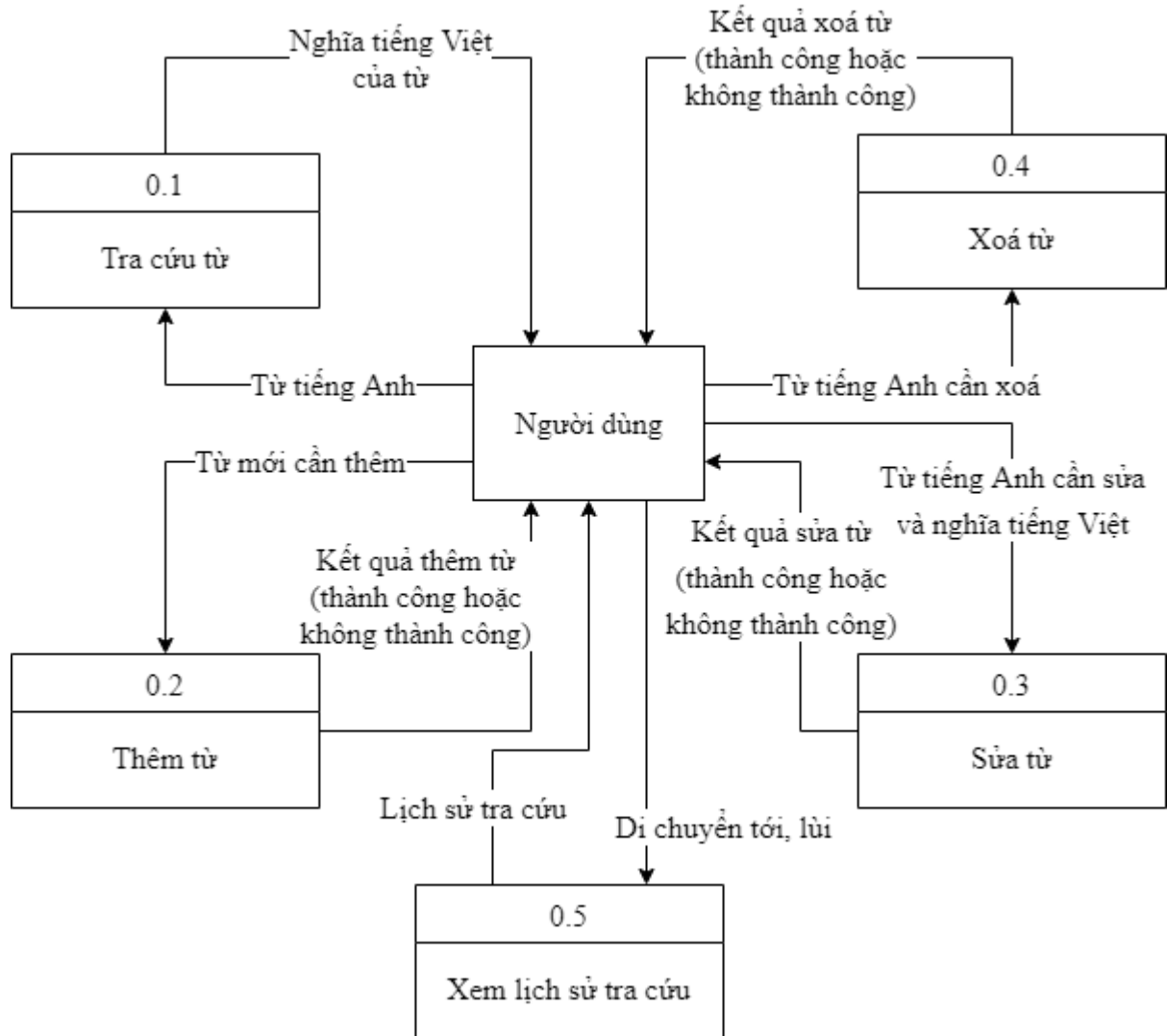
Ở đây, chức năng chính của chương trình sẽ là tra cứu từ tiếng Anh, chương trình sẽ trả kết quả về là nghĩa tiếng Việt của từ đó.



Sơ đồ 3.1: Sơ đồ DFD mức ngữ cảnh

3.2.1.2 Mức 0 (Level 0)

Đi sâu hơn, chương trình sẽ gồm các chức năng: Tra cứu từ, thêm từ, sửa từ, xoá từ và xem lịch sử tra cứu.

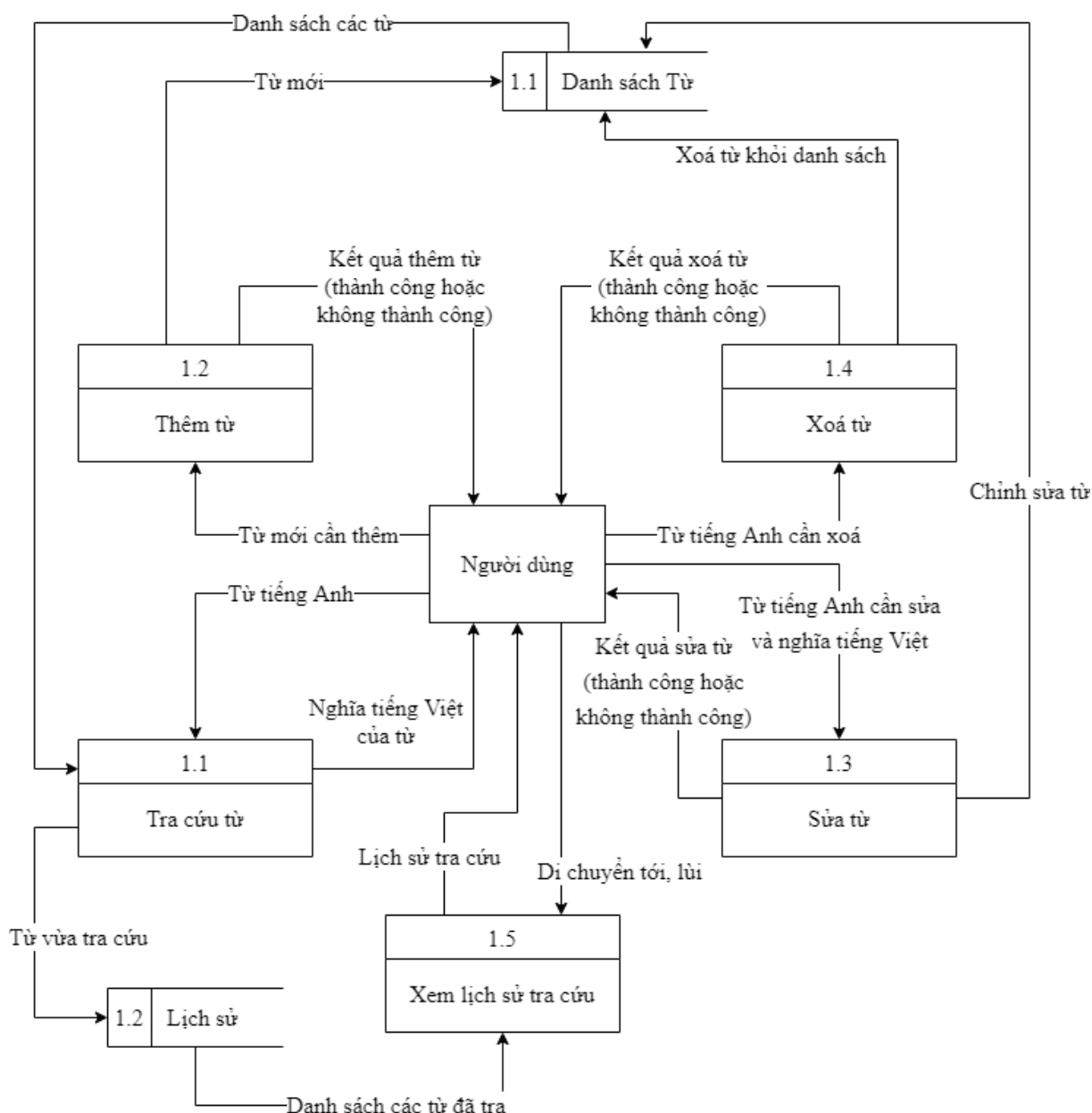


Sơ đồ 3.2: Sơ đồ DFD mức 0

3.2.1.3 Mức 1 (Level 1)

Đi sâu hơn nữa, các chức năng sẽ tương tác đến nơi lưu trữ dữ liệu từ. Chương trình sẽ có dữ liệu tạm thời được lưu trong quá trình chạy chương trình, và dữ liệu cứng sẽ được lưu trữ dưới dạng file text. Khi thêm từ mới, sửa từ đã có và xoá từ sẽ tác động lên loại dữ liệu tạm thời. Khi thoát chương trình, nếu dữ liệu tạm thời có sự thay đổi trong quá trình sử dụng, thì sẽ hỏi người dùng có muốn lưu lại dữ liệu thành dữ liệu cứng hay không. Nếu có, thì sẽ viết dữ liệu ra dữ liệu cứng (file text), ngược lại thì thoát chương trình.

Khi tra cứu từ, từ vừa tra cứu sẽ được lưu tự động vào Lịch sử dưới dạng dữ liệu tạm thời (được cài đặt bằng Danh sách liên kết).



Sơ đồ 3.3: Sơ đồ DFD mức 1

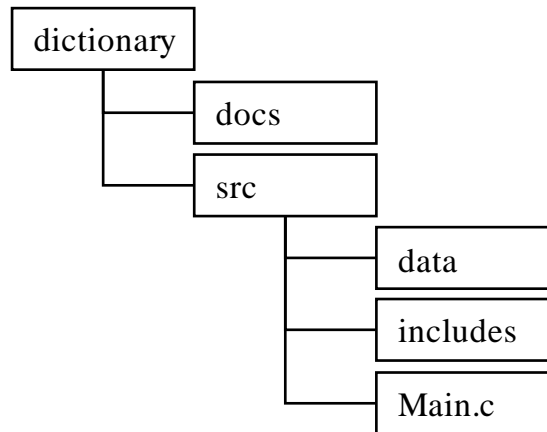
3.3 CÁCH THỨC XÂY DỰNG CHƯƠNG TRÌNH

3.3.1 Cấu trúc dự án

Dự án được đặt trong thư mục gốc dictionary gồm các thư mục con:

- Thư mục docs: chứa các tệp tài liệu cho dự án.
- Thư mục src: chứa các code dự án, được chia làm:
 - Thư mục data: chứa file data của chương trình (file dữ liệu nạp lúc chương trình khởi động).

- Thư mục includes: chứa các file header (các file này thường có phần mở rộng là .h) cài đặt các kiểu dữ liệu và các code khác của chương trình.
- File Main.c: file main chương trình.



Sơ đồ 3.4: Sơ đồ cấu trúc dự án

3.3.2 Xây dựng chương trình chính

Trước khi xây dựng chương trình, ta viết cả hàm để xử lý giao diện cho việc xây dựng giao diện nhanh hơn mà không cần viết lại hàm printf, bao gồm các hàm:

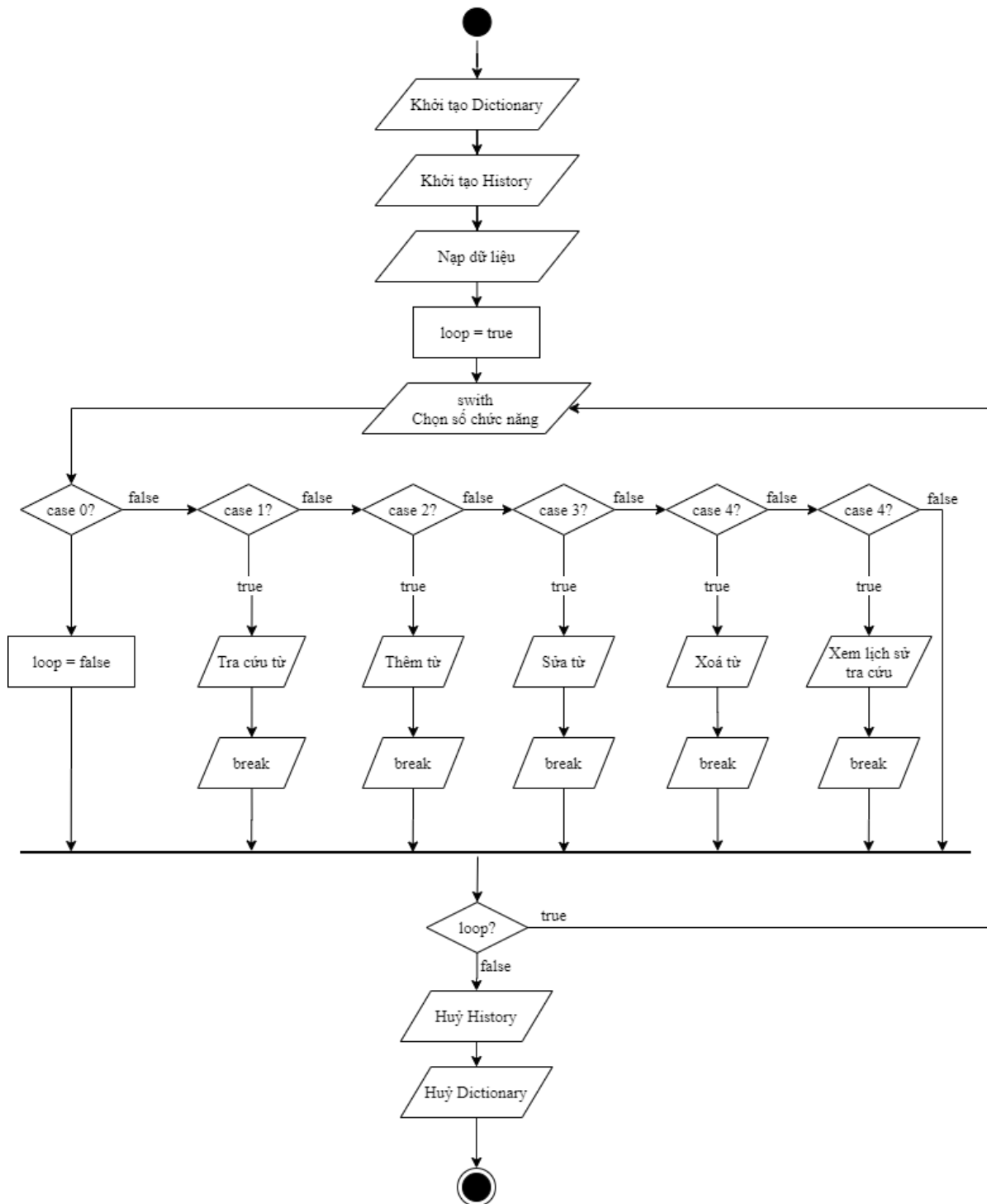
- Hàm in tên chương trình.
- Các hàm in chuỗi ký tự.
- Hàm in danh sách chức năng.
- Hàm in bảng chọn khi người thoát chương trình.
- Hàm in tên chức năng khi người dùng chọn.
- Hàm in kết quả tra cứu từ.

3.3.2.1 Cách thức hoạt động

Chương trình chạy với giao diện chức năng cho người dùng dễ thao tác với các chức năng: tra cứu từ, thêm từ, xóa từ và xem lịch sử tra cứu sẽ ứng với các phím số 1, 2, 3, 4, 5 và để thoát chương trình, người dùng chỉ cần ấn phím 0.

Để xây dựng được chương trình vậy, ta sử dụng switch...case... và đọc ký tự từ bàn phím để biết người dùng chọn chức năng gì. Nhưng trước đó, ta phải khởi tạo Tự điển và Lịch sử, sau đó nạp dữ liệu từ data.txt vào Tự điển.

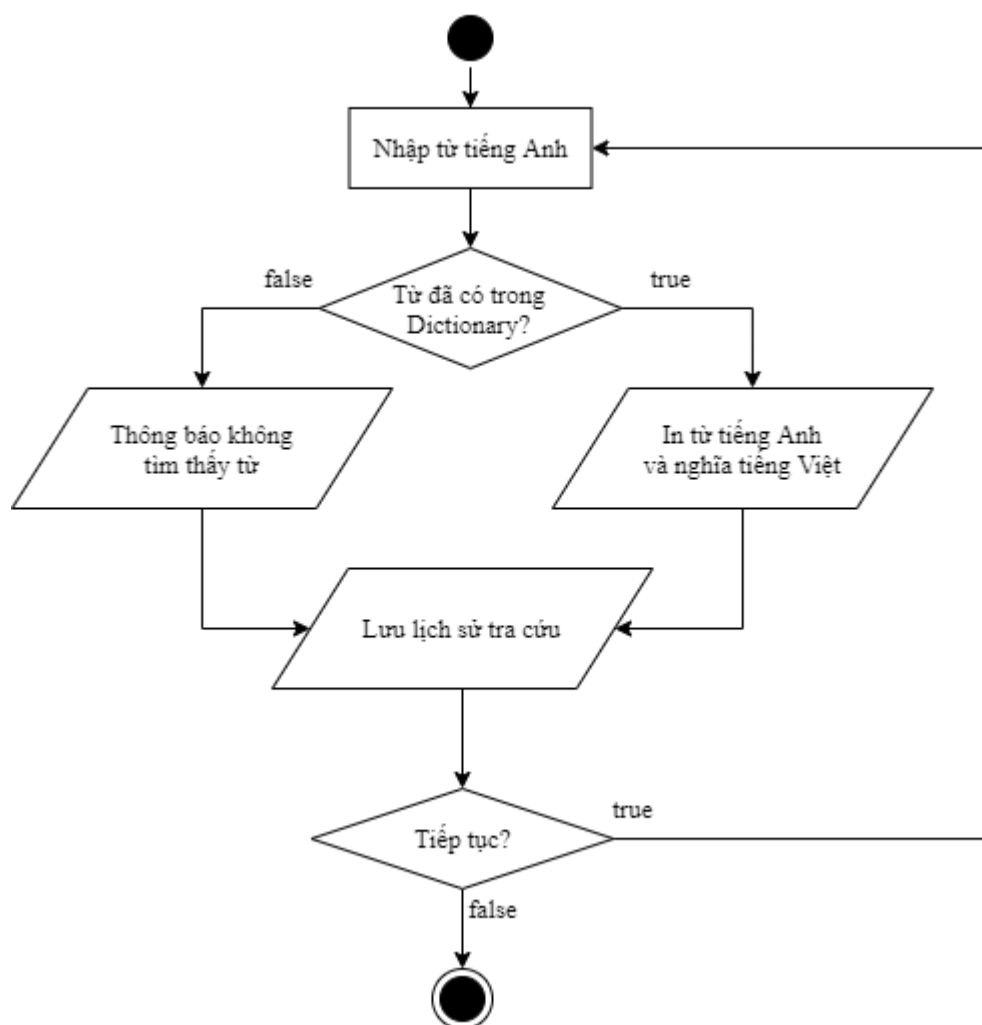
Ứng với mỗi chức năng ta chỉ cần gọi hàm đã cài đặt của kiểu Dictionary.



Sơ đồ 3.5: Sơ đồ hoạt động chương trình chính

3.3.2.2 Chức năng tra cứu từ

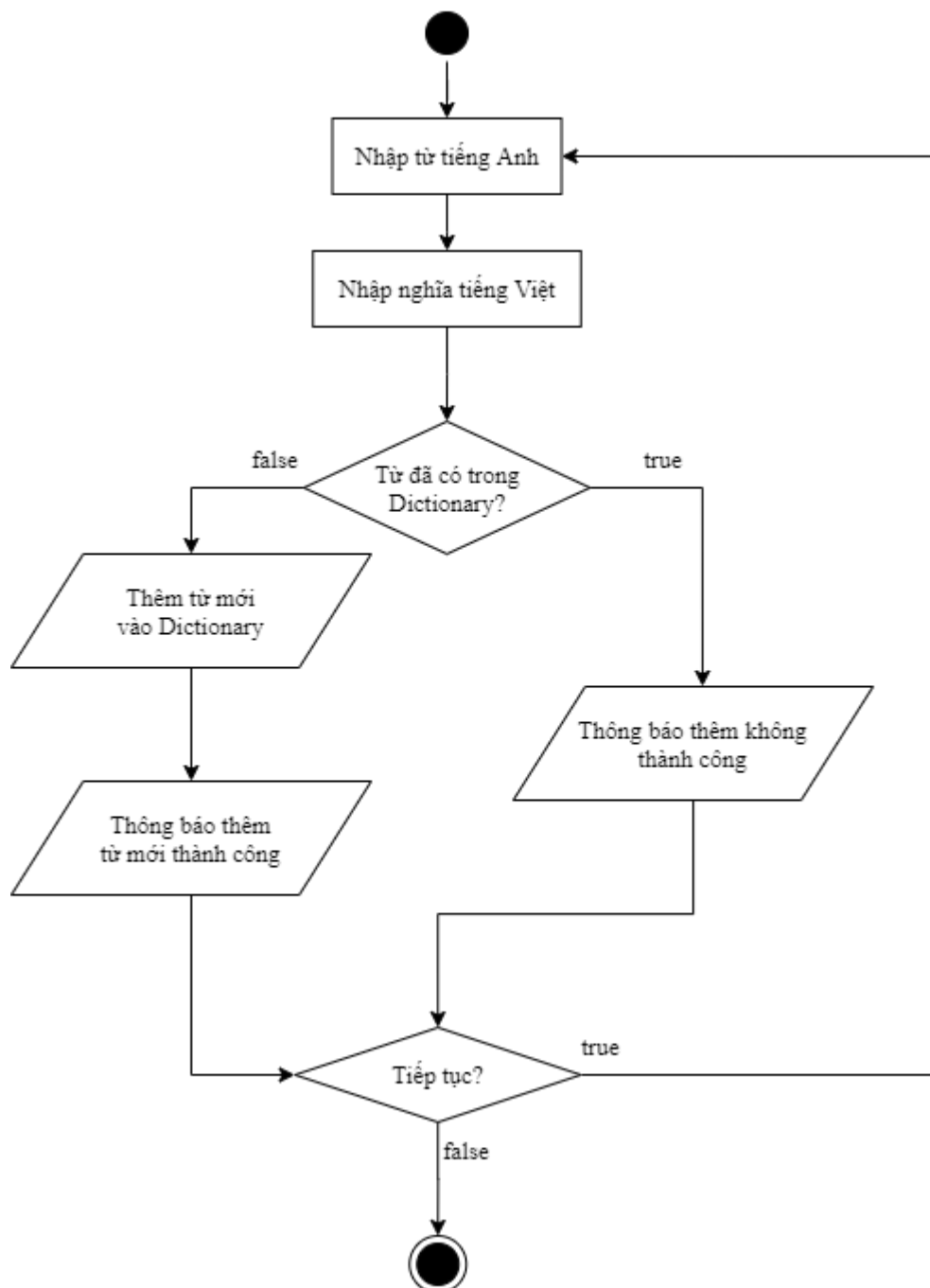
Chức năng tra cứu từ sẽ cho phép người dùng nhập từ tiếng Anh vào để tra cứu sau đó trả kết về nghĩa tiếng Việt nếu tìm thấy từ, còn nếu không thì trả về thông báo không tìm thấy từ.



Sơ đồ 3.6: Sơ đồ hoạt động chức năng tra cứu từ

3.3.2.3 Chức năng thêm từ mới

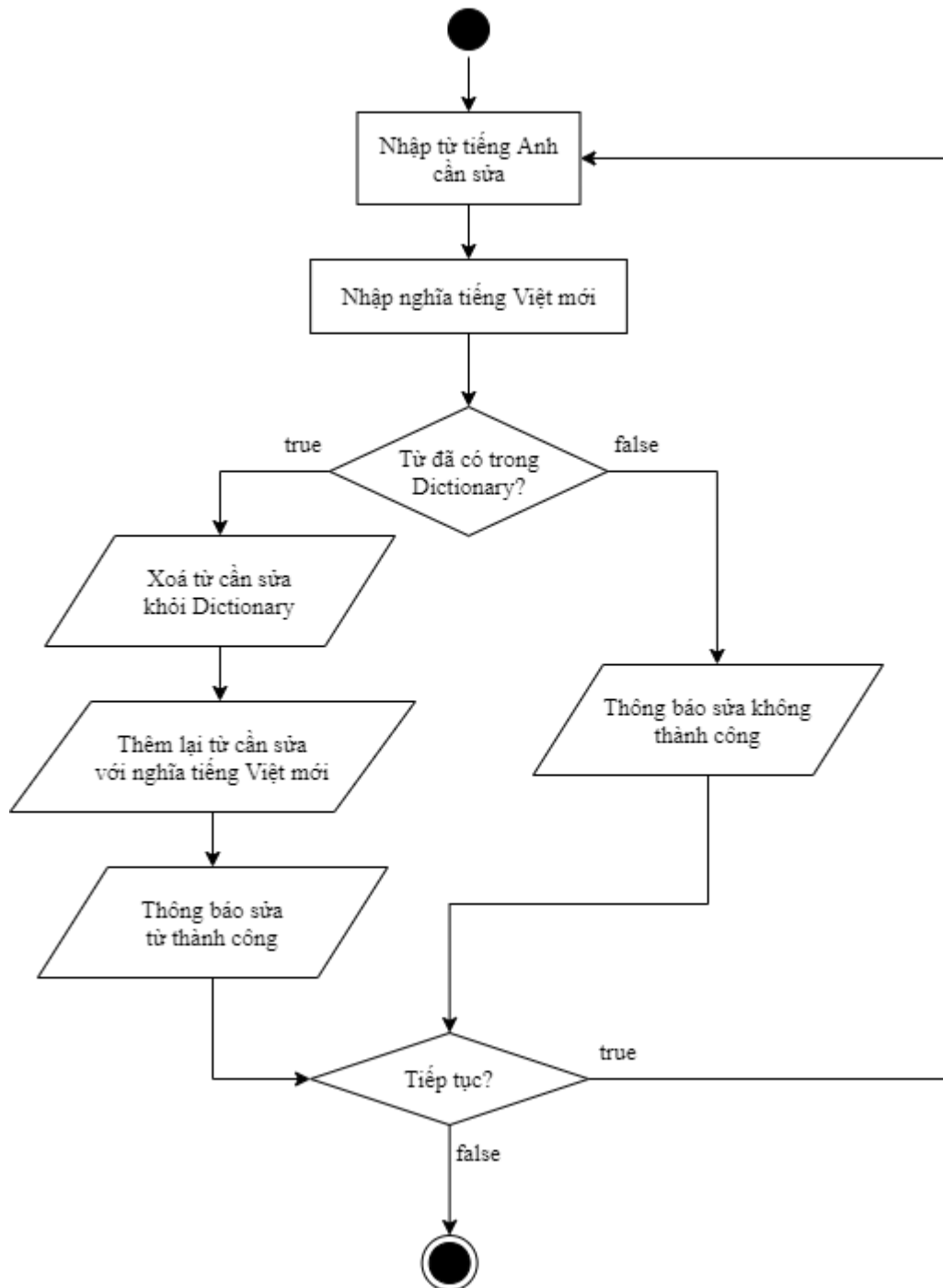
Chức năng thêm từ mới cho phép nhập từ tiếng Anh mới và nghĩa tiếng Việt của nó. Nếu thêm từ mới thành công, chương trình sẽ thông báo kết quả thành công. Nếu từ mà người dùng nhập vào đã tồn tại trong Từ điển thì chương trình sẽ thông báo từ đã tồn tại và thêm không thành công.



Sơ đồ 3.7: Sơ đồ hoạt động chức năng thêm từ mới

3.3.2.4 Chức năng sửa từ

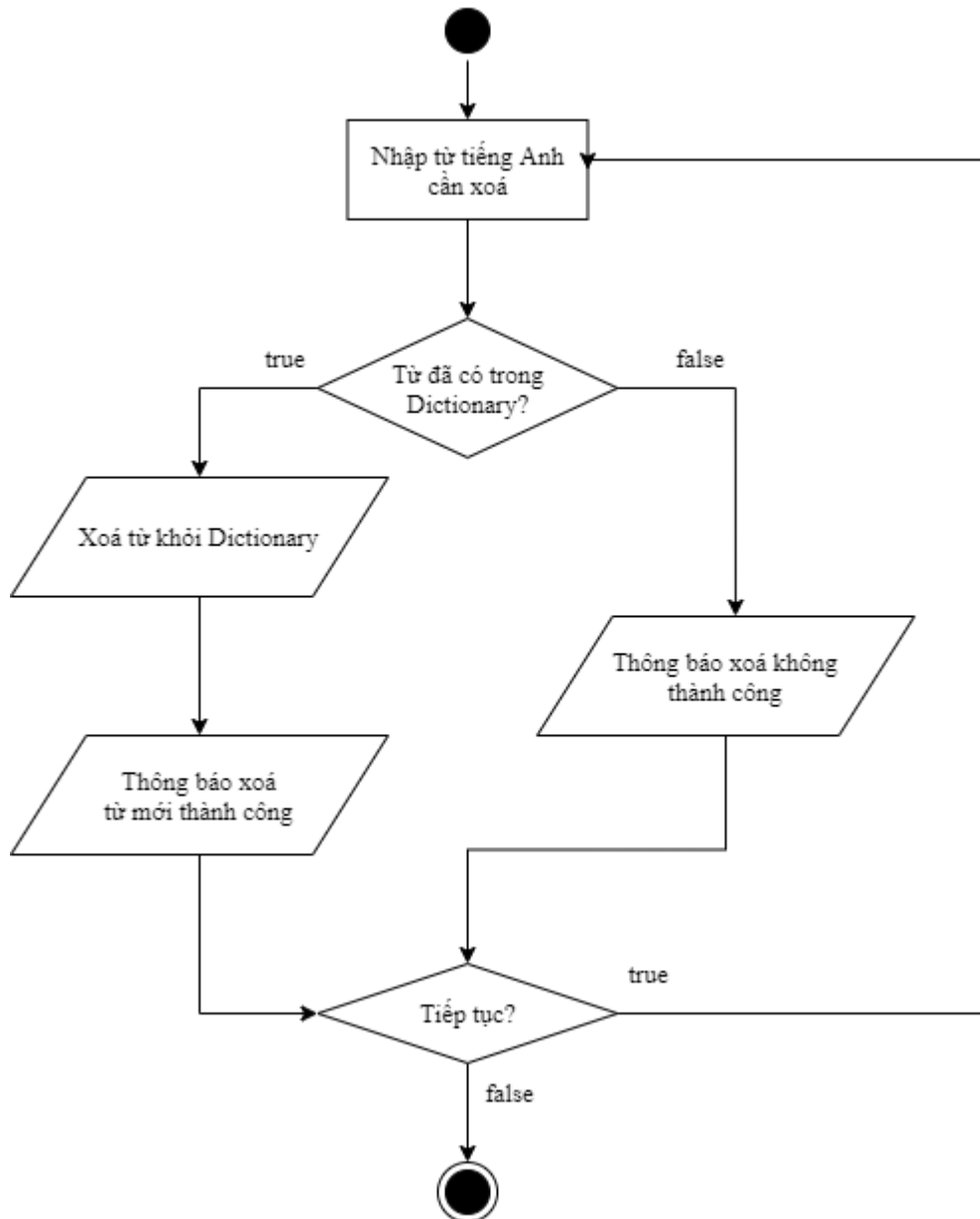
Chức năng sửa từ cho phép người dùng nhập từ tiếng Anh cần sửa và nghĩa tiếng Việt của từ đó để sửa từ. Nếu từ đó tồn tại trong Tự điển, xóa từ đã có đó trong từ điển và thêm lại từ đó với nghĩa tiếng Việt mới, sau đó thông báo sửa từ thành công sẽ được hiện lên. Nếu từ đó không có trong Tự điển, chương trình sẽ thông báo sửa từ không thành công.



Sơ đồ 3.8: Sơ đồ hoạt động chức năng sửa từ

3.3.2.5 Chức năng xoá từ

Chức năng xoá từ cho phép người dùng nhập từ tiếng Anh cần xoá để xoá từ đó ra khỏi Tự điển. Nếu từ đó tồn tại trong Tự điển, thông báo xoá từ thành công sẽ được hiện lên. Nếu từ đó không có trong Tự điển, chương trình sẽ thông báo xoá không thành công.

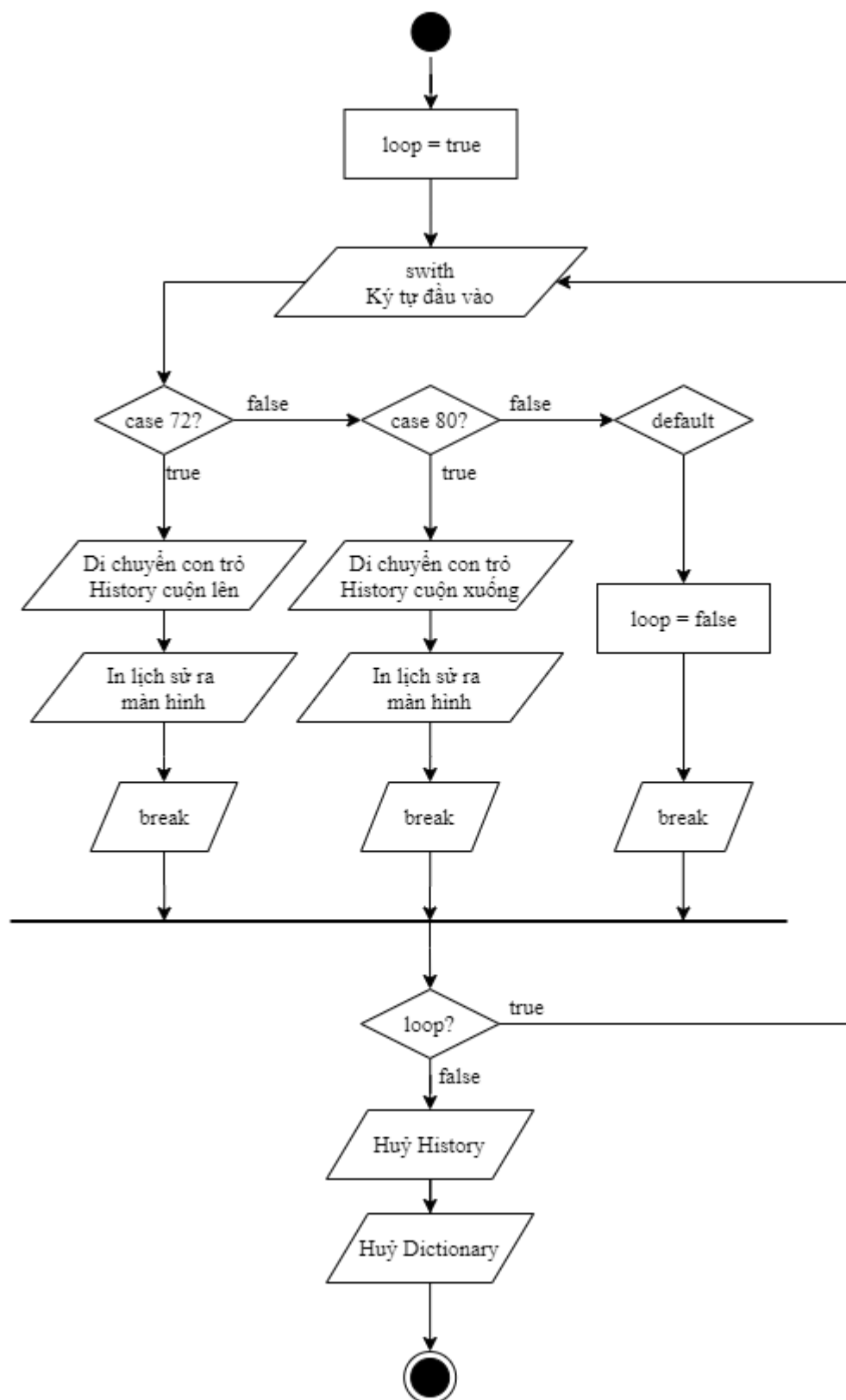


Sơ đồ 3.9: Sơ đồ hoạt động chức năng xoá từ

3.3.2.6 Xem lịch sử tra cứu

Chức năng xem lịch sử tra cứu cho phép người dùng có thể xem lại những từ mà mình đã tra cứu. Bằng cách sử dụng hai phím mũi tên lên và xuống mà cuộn con trỏ để có thể xem lịch sử. Lịch sử sẽ được ghi lại theo trình tự từ vừa tra cứu gần nhất đến xa nhất,

ghi lại từ tiếng Anh và cả nghĩa tiếng Việt của nó. Nếu từ tiếng Anh không có trong Tự điển, mục nghĩa tiếng Việt sẽ để trống.



Sơ đồ 3.10: Sơ đồ hoạt động chức năng xem lịch sử tra cứu

4 KẾT LUẬN VÀ ĐÁNH GIÁ

4.1 KẾT QUẢ ĐẠT ĐƯỢC

- Hiểu và nắm rõ về cấu trúc dữ liệu, các cài đặt và ứng dụng cấu trúc dữ liệu, kiểu dữ liệu trừu tượng.
- Thiết kế kiểu dữ liệu trừu tượng theo hướng gần với hướng đối tượng.
- Về phía chương trình thì được hoàn thiện đầy đủ các chức năng cơ bản của một Tự điển: tra cứu từ, thêm từ mới, sửa từ đã có, xoá từ và xem lịch sử tra cứu.
- Giao diện tương đối dễ sử dụng cho người dùng, hiển thị tiếng Việt.

4.2 HẠN CHẾ, NGUYÊN NHÂN

Về thiết kế thuật toán: Thuật toán tìm kiếm trên Tự điển ở Danh sách liên kết trên Bucket là cách tìm kiếm tuần tự tuyến tính (Linear Search), chưa được tối ưu, nếu danh sách có nhiều phần tử, khả năng thời gian chờ tìm kiếm sẽ lâu.

Về giao diện được xây dựng trên nền Console nên không đẹp và bắt mắt như trên giao diện Form hiện đại, nguyên nhân do chương trình được viết thuần bằng C/C++ với trình biên dịch GCC ra nền Console. Mặc dù chương trình có thể hiển thị tiếng Việt, nhưng người dùng khi nhập tiếng Việt thì chương trình không nhận được các ký tự UTF-8.

4.3 HƯỚNG PHÁT TRIỂN ĐỀ TÀI

- Cải thiện thuật toán tìm kiếm trên Tự điển (cụ thể là trên Danh sách). Có thể áp dụng các thuật toán tìm kiếm khác tối ưu hơn như: Tìm kiếm nhị phân (Binary Search) hay Tìm kiếm nội suy (Interpolation Search).
- Cải thiện giao diện người dùng, xây dựng giao diện người dùng đẹp mắt và dễ sử dụng hơn, người dùng có thể thao tác bằng chuột thay vì bằng bàn phím, sửa được lỗi khắc phục việc chương trình không nhận được ký tự tiếng Việt của người dùng nhập vào.
- Chuyển nền tảng và ngôn ngữ xây dựng chương trình sang một nền tảng khác như chương trình Tự điển Online với nền tảng Web app, hoặc cho điện thoại di động với App Mobile Android và IOS, hoặc vẫn trên máy tính nhưng với giao diện đồ hoạ với Winform viết bằng C# hay Java Application với Java...

5 PHẦN PHỤ LỤC

5.1 HƯỚNG DẪN CÀI ĐẶT CHƯƠNG TRÌNH

Chương trình có giao diện tiếng Việt và chạy trên nền Console nhưng do Command Prompt của Windows muốn hiển thị được tiếng Việt UTF-8 thì phải điều chỉnh thông số máy rất phức tạp. Nên có cách đơn giản hơn là thay vì dùng Command Prompt có sẵn ta sử dụng phần mềm Cmder.

- Trang chủ của Cmder: <https://cmder.net/>
- Hướng dẫn cài đặt và tài liệu về Cmder: <https://github.com/cmderdev/cmder>

Sau khi cài đặt Cmder. Kế tiếp ta tải file nén chương trình Tự điển qua đường dẫn sau: <https://github.com/daomtthuan/dictionary/blob/master/install/DictionaryInstall.zip?raw=true>

Sau khi tải về ta có file DictionaryInstall.zip, giải nén file tại thư mục bất kỳ theo ý muốn, ta được một thư mục có tên Dictionary.

Sau đó mở phần mềm Cmder và dẫn tới đường dẫn nơi đã giải nén bằng lệnh:

```
cd <tên đường dẫn>
```

Bảng 5.1: Cú pháp lệnh đi đến đường dẫn

Ví dụ: nơi giải nén của chương trình ở ổ đĩa D, giải nén được thư mục Dictionary, ta gõ:

```
D:
cd D:\Dictionary
```

Bảng 5.2: Ví dụ đi đến đường dẫn chương trình

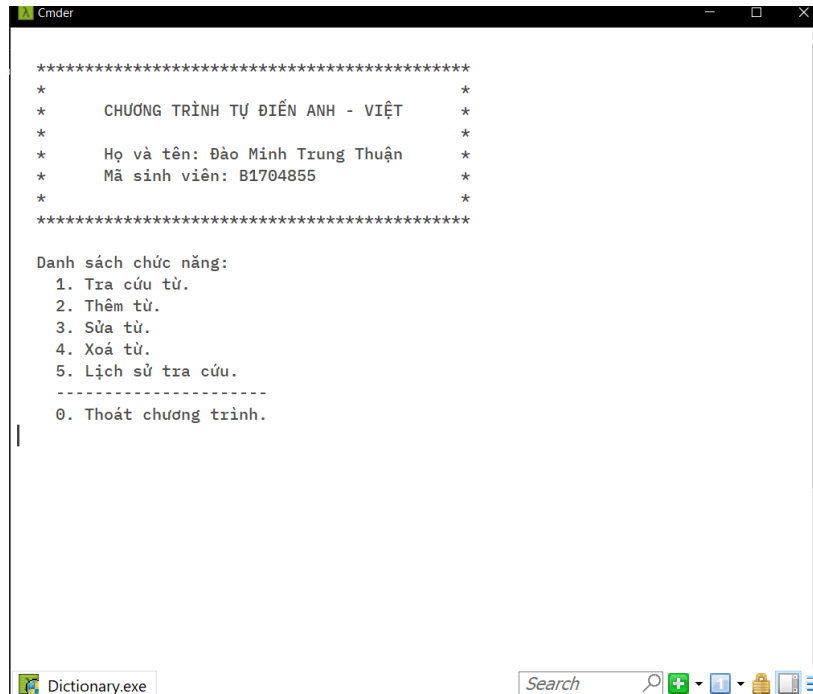
Sau đó ta gõ lệnh sau để chạy chương trình:

```
start Dictionary
```

Bảng 5.3: Lệnh chạy chương trình

5.2 HƯỚNG DẪN SỬ DỤNG

Khi chạy chương trình, màn hình sẽ hiện lên danh sách chức năng, bấm phím số 0, 1, 2, 3, 4 và 5 để lựa chọn chức năng.



Hình 5.1: Giao diện chính của chương trình

Khi vào chức năng tra cứu từ, nhập từ tiếng Anh muốn tra cứu và nhấn Enter.



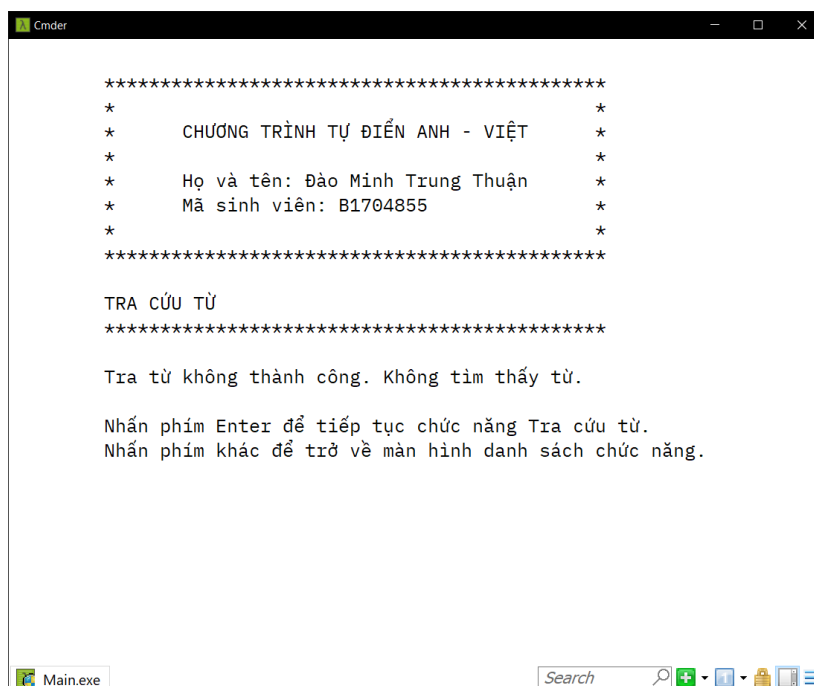
Hình 5.2: Chức năng tra cứu từ

Chương trình sẽ trả về kết quả nghĩa tiếng Việt của từ đó.



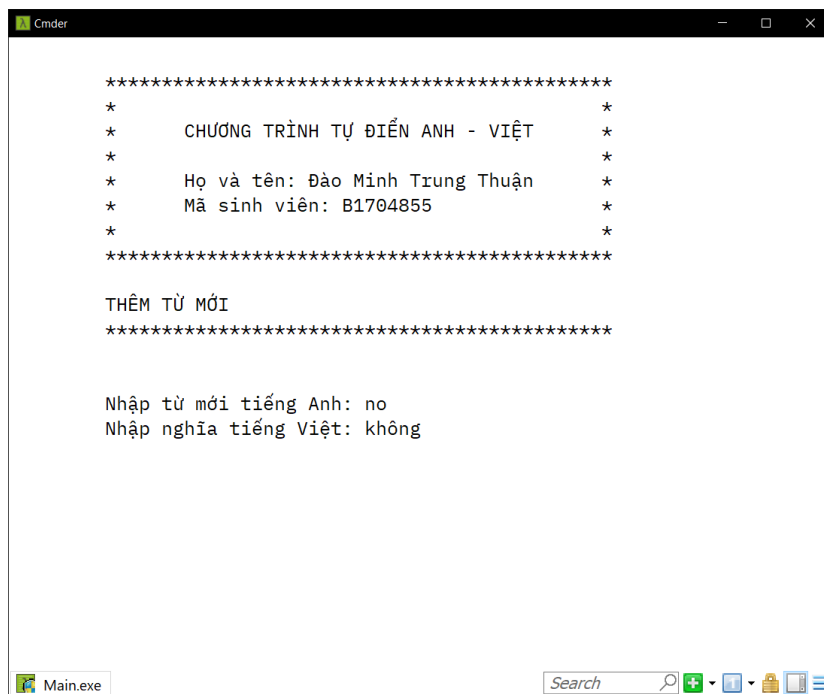
Hình 5.3: Kết quả tra cứu từ nếu tìm thấy từ

Nếu từ tra cứu không có trong Tự điển, chương trình sẽ thông báo không tìm thấy từ.



Hình 5.4: Thông báo không tìm thấy từ

Sang chức năng thêm từ mới, nhập từ tiếng Anh và nghĩa tiếng Việt của từ và nhấn Enter.



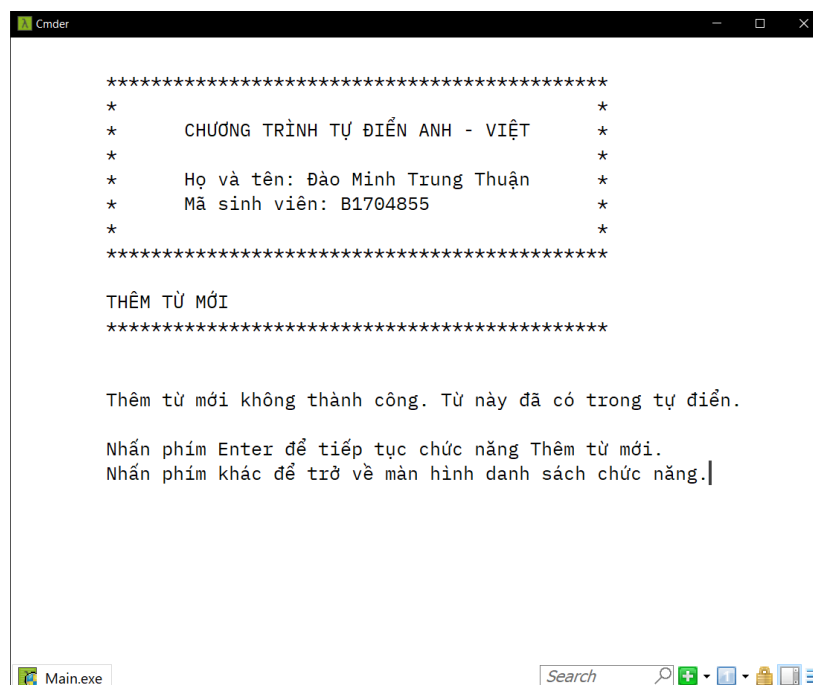
Hình 5.5: Nhập từ mới cần thêm

Nếu thêm từ mới thành, chương trình sẽ thông báo kết quả thành công.



Hình 5.6: Kết quả thêm từ mới

Nếu từ mà người dùng nhập vào đã tồn tại trong Từ điển thì chương trình sẽ thông báo từ đã tồn tại và thêm không thành công. Trong trường hợp này, buộc phải xóa từ đã có trong tự điển rồi mới thêm từ mới.



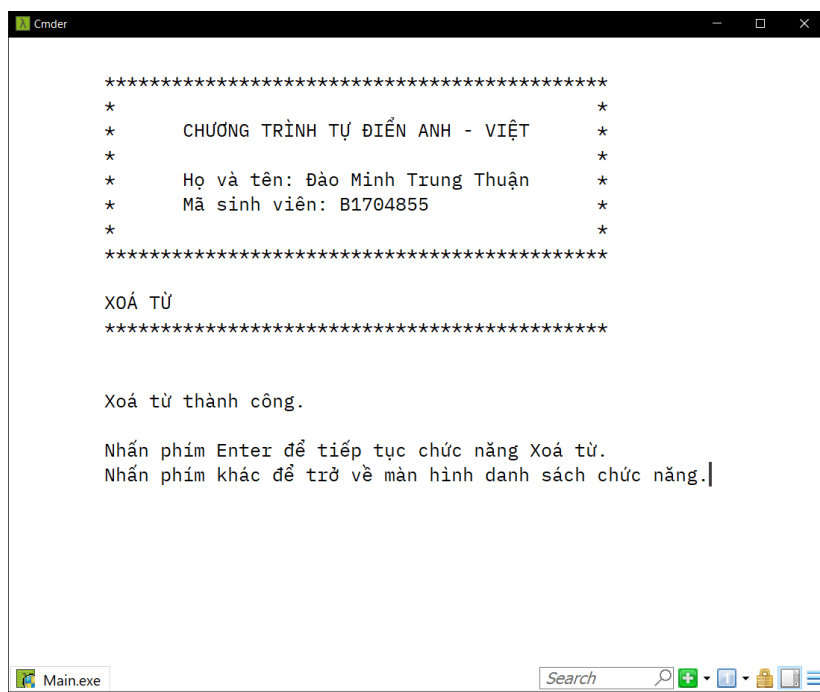
Hình 5.7: Thông báo thêm từ không thành công

Khi vào chức năng xóa từ, chỉ cần nhập từ tiếng Anh cần xóa và nhấn Enter.



Hình 5.8: Nhập từ cần xóa

Nếu từ đó tồn tại trong Tự điển, thông báo xóa từ thành công sẽ được hiện lên.



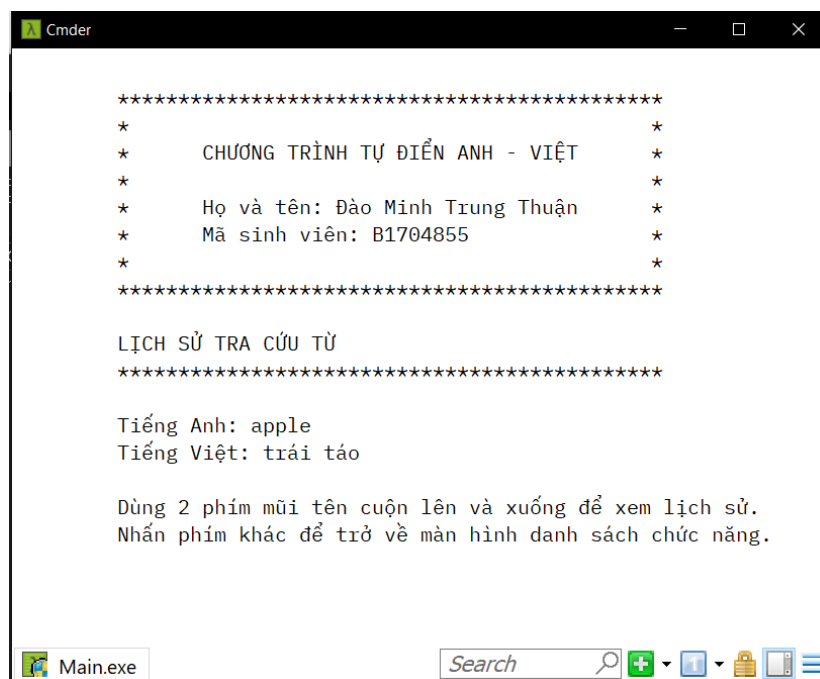
Hình 5.9: Thông báo xoá từ thành công

Nếu từ đó không có trong Tự điển, chương trình sẽ thông báo xoá không thành công.



Hình 5.10: Thông báo xoá từ không thành công

Khi vào chức năng xem lại lịch sử đã tra cứu từ, sử dụng hai phím mũi tên lên và xuống để điều khiển cuộn lịch sử.



Hình 5.11: Chức năng xem lịch sử tra cứu

Nếu chưa tra cứu từ nào, lịch sử sẽ trống.



Hình 5.12: Lịch sử trống

Nếu trong quá trình sử dụng có sự thay đổi dữ liệu (thêm từ, sửa từ, xóa từ) thì trước khi thoát chương trình, sẽ có thông báo hỏi có muốn lưu lại dữ liệu hay không.



Hình 5.13: Lưu lịch sử trước khi thoát

6 TÀI LIỆU THAM KHẢO

1. Trần Cao Đệ; Giáo trình Cấu trúc dữ liệu Data Structures; Nhà xuất bản Đại học Cần Thơ; 2012.
2. Nguyễn Văn Linh, Trần Cao Đệ, Trương Thị Thanh Tuyền, Lâm Hoài Bảo, Phan Huy Cường, Trần Ngân Bình; Giáo trình Cấu trúc dữ liệu; Nhà xuất bản Đại học Cần Thơ; 2003.
3. Bài giảng Lập trình căn bản, Nhà xuất bản Đại học Cần Thơ; Lưu hành nội bộ
4. Lê Trần Đạt; Tài liệu Khoá học C++; <https://cpp.daynhauhoc.com/>