

**BỘ GIAO THÔNG VẬN TẢI
TRƯỜNG ĐẠI HỌC HÀNG HẢI
BỘ MÔN: KHOA HỌC MÁY TÍNH
KHOA: CÔNG NGHỆ THÔNG TIN**

BÀI GIẢNG

NGÔN NGỮ HÌNH THỨC VÀ ÔTÔMAT

TÊN HỌC PHẦN : Ngôn ngữ hình thức và Ôtômat
MÃ HỌC PHẦN : 17204
TRÌNH ĐỘ ĐÀO TẠO : ĐẠI HỌC CHÍNH QUY
DÙNG CHO SV NGÀNH : CÔNG NGHỆ THÔNG TIN

HẢI PHÒNG - 2008

Bài giảng môn học: Ngôn ngữ hình thức và Otomat

ĐỀ CƯƠNG CHI TIẾT

Tên học phần: Ngôn ngữ hình thức và Ôtômat
Bộ môn phụ trách giảng dạy: Khoa học Máy tính
Mã học phần: 17204

Loại học phần: 1
Khoa phụ trách: CNTT
Tổng số TC: 2

TS tiết	Lý thuyết	Thực hành/Xemina	Tự học	Bài tập lớn	Đồ án môn học
45	45	0	0	0	0

Điều kiện tiên quyết:

Sinh viên phải học xong môn học toán rời rạc.

Mục tiêu của học phần:

- Cung cấp các kiến thức cơ bản về ngôn ngữ, văn phạm và otomat.
- Cơ bản về chương trình dịch.

Nội dung chủ yếu

Gồm các phần:

- Văn phạm và ngôn ngữ
- Ngôn ngữ chính quy và otomat đẩy xuống
- Ngôn ngữ phi ngữ cảnh và otomat đẩy xuống
- Cơ bản về chương trình dịch

Nội dung chi tiết của học phần:

TÊN CHƯƠNG MỤC	PHÂN PHỐI SỐ TIẾT				
	TS	LT	TH/Xemina	BT	KT
MỞ ĐẦU					
Chương I. Văn phạm và ngôn ngữ.	05	04	01		
1.1. Bảng chữ cái, từ và ngôn ngữ 1.2. Tích ghép, phép chia, phép soi gương 1.3. Các phép toán trên ngôn ngữ 1.4. Văn phạm 1.5. Các ví dụ về văn phạm					
Chương II. Ngôn ngữ chính quy và otomat hữu hạn	16	12	03		01
2.1. Nguồn và ngôn ngữ được sinh bởi nguồn 2.2. Các phép toán trên nguồn 2.3. Otomat hữu hạn không lỗi ra và ngôn ngữ được đoán nhận bởi otomat hữu hạn không lỗi ra 2.4. Sự tương đương của nguồn và Otomat hữu hạn không lỗi ra 2.5. Sự tương đương của nguồn và văn phạm chính quy 2.6. Sự tương đương của nguồn và biểu thức chính quy 2.7. Bài tập tổng hợp 2.8. Tính đóng của lớp ngôn ngữ chính quy 2.9. Điều kiện cần của ngôn ngữ chính quy 2.10. Điều kiện cần và đủ của lớp ngôn ngữ chính quy 2.11. Otomat hữu hạn có lỗi ra 2.12. Ngôn ngữ chính quy					

Bài giảng môn học: Ngôn ngữ hình thức và Otomat

TÊN CHƯƠNG MỤC	PHÂN PHỐI SỐ TIẾT				
	TS	LT	TH/Xemina	BT	KT
Chương III. Ngôn ngữ phi ngữ cảnh và otomat đẩy xuống.	09	06	02		01
3.1. Cây và phép thế cây 3.2. Dạng chuẩn Chomsky 3.3. Cây dẫn xuất và các tính chất 3.8. Khái niệm về Otomat đẩy xuống					
Chương IV: Cơ bản về chương trình dịch	15	12	02		01
4.1. Giới thiệu chương trình dịch 4.2. Chương trình dịch 4.2.1. Định nghĩa chương trình dịch 4.2.2. Cấu trúc của chương trình dịch 4.2.3. Cấu trúc tĩnh (cấu trúc logic) 4.2.4. Cấu trúc động 4.2.5. Vị trí của chương trình dịch trong hệ thống dịch 4.3. Sự cần thiết nghiên cứu chương trình dịch 4.4. Bộ phân tích cú pháp					

Nhiệm vụ của sinh viên :

Tham dự các buổi thuyết trình của giáo viên, tự học, tự làm bài tập do giáo viên giao, tham dự các bài kiểm tra định kỳ và cuối kỳ.

Tài liệu học tập :

- Nguyễn Văn Ba, *Ngôn ngữ hình thức*, Trường Đại học Bách khoa Hà nội, 1997
- Phan Đình Diệu, *Lý thuyết otomat và thuật toán*, Nhà xuất bản Đại học và Trung học Chuyên nghiệp, 1971
- Đỗ Đức Giáo, Đặng Huy Ruận, *Ngôn ngữ hình thức*, Nhà xuất bản KHKT, 1991
- Phạm Hồng Nguyên, *Giáo trình chương trình dịch*, ĐH KHTN HN
- Nguyễn Văn Ba, *Phân tích cú pháp*, ĐH Bách khoa HN

Hình thức và tiêu chuẩn đánh giá sinh viên:

- Hình thức thi cuối kỳ : Thi viết.
- Sinh viên phải đảm bảo các điều kiện theo Quy chế của Nhà trường và của Bộ

Thang điểm: Thang điểm chữ A, B, C, D, F

Điểm đánh giá học phần: $Z = 0,2X + 0,8Y$.

Bài giảng này là tài liệu **chính thức và thống nhất** của Bộ môn Khoa học máy tính, Khoa Công nghệ thông tin và được dùng để giảng dạy cho sinh viên.

Ngày phê duyệt: / /2010

Trưởng Bộ môn: Thạc sỹ Nguyễn Hữu Tuấn

MỤC LỤC

Nội dung	Trang
Mục lục	
Chương 1: Văn phạm và ngôn ngữ	1
1. 1. Bảng chữ cái, từ và ngôn ngữ	1
1. 2. Tích ghép, phép chia, phép soi gương	2
1. 3. Các phép toán trên ngôn ngữ	4
1. 4. Văn phạm	6
1. 5. Các ví dụ về văn phạm	8
Bài tập	8
Chương 2: Ngôn ngữ chính quy và Otomat hữu hạn	9
2. 1. Nguồn và ngôn ngữ được sinh bởi nguồn	9
2. 2. Các phép toán trên nguồn	10
2. 3. Otomat hữu hạn không lỗi ra và ngôn ngữ được đoán nhận bởi Otomat hữu hạn không lỗi ra	18
2. 4. Sự tương đương của nguồn và Otomat hữu hạn	26
2. 5. Sự tương đương của nguồn và văn phạm chính quy	30
2. 6. Sự tương đương của nguồn và biểu thức chính quy	30
2. 7. Bài tập tổng hợp	30
2. 8. Tính đóng của lớp ngôn ngữ chính quy	31
2. 9. Điều kiện cần của ngôn ngữ chính quy	31
2. 10. Điều kiện cần và đủ của lớp ngôn ngữ chính quy	31
2. 11. Otomat hữu hạn có lỗi ra	31
2. 12. ω - Ngôn ngữ chính quy	31
Bài tập	31
Chương 3: Ngôn ngữ phi ngữ cảnh và Otomat đẩy xuống	32
3. 1. Cây và phép thế cây	32
3. 2. Dạng chuẩn Chomsky	32
3. 3. Cây dẫn xuất và các tính chất	32
3. 4. Khái niệm về Otomat đẩy xuống	32
Bài tập	32
Chương 4: Cơ bản về chương trình dịch	33
4. 1. Giới thiệu về chương trình dịch	33
4. 2. Chương trình dịch	33
4. 2. 1. Định nghĩa chương trình dịch	33
4. 2. 2. Cấu trúc của chương trình dịch	37
4. 2. 3. Cấu trúc tĩnh của chương trình dịch	44
4. 2. 4. Cấu trúc động của chương trình dịch	50
4. 2. 5. Vị trí của chương trình dịch trong hệ thống dịch	58
4. 3. Sự cần thiết phải nghiên cứu chương trình dịch	58
4. 4. Bộ phân tích cú pháp	60
Bài tập	62
Một số đề thi mẫu	63
Tài liệu tham khảo	64

CHƯƠNG 1: VĂN PHẠM VÀ NGÔN NGỮ

Kiến thức cơ bản: Để tiếp thu tốt nội dung của chương này, sinh viên cần có một số kiến thức liên quan về chuỗi, ký hiệu, từ trong các ngôn ngữ tự nhiên như tiếng Việt, tiếng Anh; cấu trúc cú pháp của các chương trình máy tính viết bằng một số ngôn ngữ lập trình cơ bản như Pascal, C...

1.1. Bảng chữ cái, từ, ngôn ngữ

Các ngôn ngữ lập trình (như Pascal, C, ...) lẫn ngôn ngữ tự nhiên (như tiếng Việt, tiếng Anh, ...) đều có thể xem như là tập hợp các câu theo một cấu trúc quy định nào đó. Câu của ngôn ngữ, trong tiếng Việt như "An là sinh viên giỏi" hay trong Pascal là một đoạn chương trình bắt đầu bằng từ khóa program cho đến dấu chấm câu kết thúc chương trình, đều là một chuỗi liên tiếp các từ, như "An", "giỏi" hay "begin", "if", "x2", "215", tức các chuỗi hữu hạn các phần tử của một bộ chữ cái cơ sở nào đó. Ta có thể xem chúng như là các ký hiệu cơ bản của ngôn ngữ.

Từ nhận xét đó, ta dẫn tới một quan niệm hình thức về ngôn ngữ như sau (theo từ điển): Ngôn ngữ, một cách không chính xác là một hệ thống thích hợp cho việc biểu thị các ý nghĩ, các sự kiện hay các khái niệm, bao gồm một tập các ký hiệu và các quy tắc để vận dụng chúng.

Định nghĩa trên chỉ cung cấp một ý niệm trực quan về ngôn ngữ chứ không đủ là một định nghĩa chính xác để nghiên cứu về ngôn ngữ hình thức. Chúng ta bắt đầu xây dựng định nghĩa này bằng các khái niệm mà mọi ngôn ngữ đều đặt nền tảng trên đó.

1.1.1. Bảng chữ cái

Một dãy hữu hạn hoặc vô hạn các phần tử, kí hiệu Σ được gọi là một bảng chữ cái trong đó mỗi phần tử $a \in \Sigma$ được gọi là một kí hiệu (một chữ cái).

Ví dụ: $\Sigma = \{0,1\}$: Bảng chữ cái nhị phân

$\Sigma = \{0,1,\dots,9\}$: Bảng chữ cái thập phân

1.1.2. Từ

➤ Một dãy kí hiệu $a_1, a_2, \dots, a_n$ ($a_i, i=1 \div n$) được gọi là một từ $\in \Sigma$.

Ví dụ : $\Sigma = \{0,1\}$

$a_1 = 0, a_2 = 1, a_3 = 00, a_4 = \wedge, \dots$

➤ Từ rỗng được kí hiệu là $\wedge$, mọi bảng chữ cái đều sinh ra từ rỗng.

➤ Tổng số các kí hiệu tạo nên từ được gọi là độ dài của từ.

Ví dụ : $\Sigma = \{a,b,c\}$

$a_1 = aabb, a_2 = ac, |a_1| = 4, |a_2| = 2$

Bài giảng môn học: Ngôn ngữ hình thức và Otomat

- Độ dài của từ rỗng = 0.
- Nếu a là một từ thuộc bảng chữ cái Σ , Δ là một bảng chữ cái chứa bảng chữ cái Σ thì $a \in \Delta$. Ví dụ : $\Sigma = \{0,1\}$, $\Delta = \{0,1,2,\dots,9\}$
 $a_1 = 0110 \in \Sigma$, $\Sigma \in \Delta \Rightarrow a \in \Delta$
- Tập hợp tất cả các từ $\in \Sigma$ kể cả từ rỗng kí hiệu là Σ^* .
- Tập hợp tất cả các từ $\in \Sigma$ trừ rỗng kí hiệu là Σ^+ . Vậy $\Sigma^+ = \Sigma^* \setminus \{\epsilon\}$

1.1.3. Ngôn ngữ

Ngôn ngữ là một tập hợp các câu có một cấu trúc qui định nào đó. Một câu của ngôn ngữ là một dãy (hay xâu) các từ có sẵn được liệt kê trong một bảng chữ nào đó, như là các ký hiệu cơ bản của ngôn ngữ. Giả sử có bảng chữ cái Σ .

- Tập $A \in \Sigma^*$ được gọi là một ngôn ngữ sinh bởi Σ .
Ví dụ : $\Sigma = \{a,b,c\}$
 $A = \{x \in \Sigma^* / x \text{ bắt đầu bởi kí hiệu } a\}$ được gọi là một ngôn ngữ $\in \Sigma^*$.
- Ngôn ngữ trống kí hiệu ϕ .
- Mọi bảng chữ cái Σ đều sinh ra ϕ .
- Ngôn ngữ trống và ngôn ngữ chứa từ rỗng là khác nhau.

1.2. Tích ghép, phép chia, phép soi gương

1.2.1. Tích ghép hai xâu

- Giả sử có bảng chữ cái Σ
Các từ $\alpha = a_1a_2a_3\dots a_n$, $\beta = a_1a_2\dots a_m$ ($a_i \in \Sigma$)
- Tích ghép của α và β được hình thành bằng cách ghép từ β vào sau từ α .

$$\chi = \alpha.\beta = a_1a_2a_3\dots a_na_1a_2\dots a_m$$

Ví dụ : $\Sigma = \{0,1\}$

$$\alpha = 011, \beta = 1101$$

$$\chi = 0111101$$

- Tính chất của tích ghép
- + Tích ghép không có tính chất giao hoán

$$\alpha\beta \neq \beta\alpha$$

- + Tích ghép có tính chất kết hợp

$$(\alpha\beta)\chi = \alpha(\beta\chi)$$

Bài giảng môn học: Ngôn ngữ hình thức và Otomat

+ Tích ghép của một xâu với một xâu rỗng bằng chính nó

$$\alpha.\wedge = \wedge.\alpha = \alpha$$

Với tích ghép: $\chi = \alpha.\beta$

α : Tiếp đầu ngữ

β : Tiếp vị ngữ

1.2.2. Chia xâu

Giả sử có $z = x.y$ ($x, y, z \in \Sigma^*$)

+ Phép chia trái

Nếu ta thực hiện ngắt bỏ xâu x ra khỏi z có nghĩa là ta đã thực hiện phép chia trái xâu z cho x và kết quả là xâu y .

Kí hiệu $x \setminus^z = y$

+ Phép chia phải

Nếu ta thực hiện ngắt xâu y ra khỏi xâu z có nghĩa là ta đã thực hiện phép chia phải xâu z cho xâu y và kết quả là xâu x .

Kí hiệu $z/y = x$

Ví dụ : $\Sigma = \{0,1\}, x = 10, y = 11$

$$Z = x.y = 1011$$

$$x \setminus^Z = 11$$

$$Z/y = 10$$

1.2.3. Phép soi gương

+ Giả sử có bảng chữ cái Σ

+ Từ $\alpha = a_1a_2\dots a_{m-1}a_m \in \Sigma^*$

+ Từ $\tilde{\alpha} = a_ma_{m-1}\dots a_2a_1$ được gọi là từ soi gương của từ α hay còn gọi là từ ngược của từ α

$$|\tilde{\alpha}| = |\alpha|$$

1.3. Các phép toán trên ngôn ngữ

1.3.1. Phép hợp hai ngôn ngữ

Giả sử có bảng chữ cái Σ , A, B, C là các ngôn ngữ được sinh ra bởi Σ

$$C = A \cup B = \{x \in \Sigma^* / x \in A \text{ hoặc } x \in B\}$$

Ví dụ $\Sigma = \{a,b,c\}$

$$A = \{a, b, ab, ac, cb\}, B = \{aa, bb, cc\}$$

$$C = A \cup B = \{a, b, ab, ac, cb, aa, bb, cc\}$$

+ Tính chất

➤ Phép hợp có tính chất giao hoán $A \cup B = B \cup A$

➤ Có tính chất kết hợp

$$(A \cup B) \cup C = A \cup (B \cup C)$$

1.3.2. Phép giao hai ngôn ngữ

+ Giả sử có bảng chữ cái Σ , A, B, C là các ngôn ngữ sinh ra bởi Σ

$$C = A \cap B = \{x \in \Sigma^* \mid x \in A \text{ và } x \in B\}$$

Ví dụ : $\Sigma = \{a, b\}$

$$A = \{a, ab, ac, cb\}, B = \{b, ab, cb, aa, bc\}$$

$$C = A \cap B = \{ab, cb\}$$

+ Tính chất

➤ Tính kết hợp

$$C \cap (A \cap B) = B \cap (A \cap C)$$

➤ Tính giao hoán

$$A \cap B = B \cap A$$

➤ $A \cap \phi = \phi \cap A = \phi$

1.3.3. Phép tích ghép hai ngôn ngữ

+ Giả sử có bảng chữ cái Σ , A, B, C là các ngôn ngữ được sinh bởi từ Σ

$$C = A.B = \{z \in \Sigma^* \mid z = x.y \mid x \in A, y \in B\}$$

Ví dụ : $\Sigma = \{a, b, c\}$

$$A = \{a, b, ab, ac\}, B = \{c, cb\}$$

$$C = A.B = \{ac, bc, abc, acc, acb, bcb, abcb, accb\}$$

+ Tính chất

- Không có tính chất giao hoán

$$A.B \neq B.A$$

- Có tính chất kết hợp

$$A.(B.C) = (A.B).C$$

- Có tính chất phân bố (đối với phép hợp)

$$A.(B \cup C) = (A.B) \cup (A.C)$$

1.3.4. Phép hiệu

Giả sử có bảng chữ cái Σ , A , B , C là các ngôn ngữ sinh bởi Σ

$$C = A \setminus B = \{x \in \Sigma^* \mid x \in A, x \notin B\}$$

được gọi là ngôn ngữ hiệu của hai ngôn ngữ A và B .

Ví dụ : $\Sigma = \{a, b, c\}$

$$A = \{a, b, ac, bc, aa\}, B = \{b, bc, ab, bb\}$$

$$C = B \setminus A = \{b, ab, bb\}$$

1.3.5. Phép lấy phần bù

Giả sử có bảng chữ cái Σ , A , B là các ngôn ngữ được sinh ra bởi Σ

$$A = C_B = \{x \in \Sigma^* \mid x \notin B\}$$

được gọi là ngôn ngữ phần bù hay ngôn ngữ bù của ngôn ngữ B .

Ví dụ : Cho $B = \{a, bc\}$.

Khi đó ngôn ngữ bù của ngôn ngữ B sẽ là :

$$A = \{x \in \Sigma^* \mid x \neq a, x \neq bc\}.$$

1.3.6. Phép chia hai ngôn ngữ

➤ Giả sử có bảng chữ cái Σ , A , B là các ngôn ngữ được sinh ra bởi Σ .

➤ Tập các từ $C = \{z \in \Sigma^* \mid x \in A, y \in B, x=y.z\}$ được gọi là phép chia trái của ngôn ngữ A cho ngôn ngữ B và kí hiệu $B \setminus^A$.

➤ Tập các từ $C = \{z \in \Sigma^* \mid x \in A, y \in B, x = z.y\}$ được gọi là phép chia phải của ngôn ngữ A cho ngôn ngữ B và kí hiệu là A/B .

Ví dụ : $\Sigma = \{a, b, c\}$

$$A = \{a, bc, abc, aba\}, B = \{\wedge, b, ab, cbc\}$$

$$B \setminus^A = \{a, bc, abc, aba, c\}$$

$$A/B = \{a, bc, abc, aba\}$$

1.3.7. Phép soi gương ngôn ngữ

➤ Giả sử có bảng chữ cái Σ , A , B là các ngôn ngữ được sinh bởi Σ .

➤ $B = \tilde{A} = \{\tilde{x} \in \Sigma^* / \tilde{x} \text{ là từ soi gương của } x \in A\}$.

Ví dụ $\Sigma = \{0, 1\}$
 $A = \{0, 1, 10, 110\}$
 $B = \tilde{A} = \{0, 1, 01, 011\}$

1.3.8. Phép lặp

- Giả sử có bảng chữ cái Σ , A là ngôn ngữ được sinh bởi Σ .
- $A^* = \{\wedge\} \cup A \cup A \cup \dots \cup A^n$ gọi là ngôn ngữ lặp của ngôn ngữ A .
- Tính chất :
 - $(A^*)^* = A^*$
 - $\{\wedge\}^* = \{\wedge\}$
 - $(\emptyset)^* = \{\wedge\} \cup \emptyset \cup \emptyset.\emptyset \cup \dots = \{\wedge\}$
 - $(\emptyset)^+ = \emptyset \cup \emptyset.\emptyset \cup \dots = \emptyset$

1.4. Văn phạm

Với mục đích sản sinh (hay đoán nhận) ngôn ngữ, văn phạm được dùng như một cách thức hiệu quả để biểu diễn ngôn ngữ.

1.4.1. Định nghĩa văn phạm cấu trúc (Grammar)

Theo từ điển, văn phạm, một cách không chính xác, là một tập các quy tắc về cấu tạo từ và các quy tắc về cách thức liên kết từ lại thành câu.

Để hiểu rõ hơn khái niệm này, ta xét ví dụ cây minh họa cấu trúc cú pháp của một câu đơn trong ngôn ngữ tiếng Việt "An là sinh viên giỏi" ở thí dụ 1.5 của chương 1. Xuất phát từ nút gốc theo dần đến nút lá, ta nhận thấy các từ ở những nút lá của cây như "An", "sinh viên", "giỏi", ... là những từ tạo thành câu được sản sinh. Ta gọi đó là các ký hiệu kết thúc bởi vì chúng không còn phát sinh thêm nút nào trên cây và câu được hoàn thành. Trái lại, các nút trong của cây như "câu đơn", "chủ ngữ", "danh từ", ... sẽ không có mặt trong dạng câu sản sinh, chúng chỉ giữ vai trò trung gian

trong việc sinh chuỗi, dùng diễn tả cấu trúc câu. Ta gọi đó là các ký hiệu chưa kết thúc. Quá trình sản sinh câu như trên thực chất là sự diễn tả thông qua cấu trúc cây cho một quá trình phát sinh chuỗi. Các chuỗi được phát sinh bắt đầu từ một ký hiệu chưa kết thúc đặc biệt, sau mỗi bước thay thế một ký hiệu chưa kết thúc nào đó trong chuỗi thành một chuỗi lẫn lộn gồm các ký hiệu kết thúc và chưa, cho đến khi không còn một ký hiệu chưa kết thúc nào nữa thì hoàn thành. Quá trình này chính là phương thức phát sinh chuỗi của một văn phạm, được định nghĩa hình thức như sau:

Bài giảng môn học: Ngôn ngữ hình thức và Otomat

Định nghĩa : Văn phạm cấu trúc G là một hệ thống gồm bốn thành phần xác định như sau **G** (Σ, V, S, P), trong đó:

- Σ : tập hợp các ký hiệu kết thúc (terminal)
- V : tập hợp các biến (variables) hay các ký hiệu chưa kết thúc (non terminal)
- P : tập hữu hạn các quy tắc ngữ pháp được gọi là các luật sinh (production.
- S : ký hiệu chưa kết thúc dùng làm ký hiệu bắt đầu (start)

Người ta thường dùng các chữ cái Latinh viết hoa (A, B, C, ...) để chỉ các ký hiệu trong tập biến V ; các chữ cái Latinh đầu bằng viết thường (a, b, c, ...) dùng chỉ các ký hiệu kết thúc thuộc tập Σ .

Nhận xét : Bằng quy ước này chúng ta có thể suy ra các biến, các ký hiệu kết thúc và ký hiệu bắt đầu của văn phạm một cách xác định và duy nhất bằng cách xem xét các luật sinh. Vì vậy, để biểu diễn văn phạm, một cách đơn giản người ta chỉ cần liệt kê tập luật sinh của chúng.

1.4.2. Sự phân cấp Chomsky trên văn phạm

Bằng cách áp đặt một số quy tắc hạn chế trên các luật sinh, Noam Chomsky đề nghị một hệ thống phân loại các văn phạm dựa vào tính chất của các luật sinh. Hệ thống này cho phép xây dựng các bộ nhận dạng hiệu quả và tương thích với từng lớp văn phạm. Ta có 4 lớp văn phạm như sau :

1) Văn phạm loại 0: Một văn phạm không cần thỏa ràng buộc nào trên tập các luật sinh được gọi là văn phạm loại 0 hay còn được gọi là **văn phạm không hạn chế** (Unrestricted Grammar)

2) Văn phạm loại 1: Nếu văn phạm G là văn phạm các phép thế dạng $\alpha \rightarrow \beta$ và thỏa $|\alpha| \leq |\beta|$ thì G là văn phạm loại 1 hoặc còn được gọi là **văn phạm cảm ngữ cảnh CSG** (Context-Sensitive Grammar)

Ngôn ngữ của lớp văn phạm này được gọi là ngôn ngữ cảm ngữ cảnh (CSL)

3) Văn phạm loại 2: Nếu văn phạm G có các luật sinh dạng $A \rightarrow \alpha$ với A là một biến đơn và α là một chuỗi các ký hiệu thuộc $(V \cup T)^*$ thì G là văn phạm loại 2 hoặc còn được gọi là **văn phạm phi ngữ cảnh CFG** (Context-Free Grammar)

Ngôn ngữ của lớp văn phạm này được gọi là ngôn ngữ phi ngữ cảnh (CFL)

4) Văn phạm loại 3: Nếu văn phạm G có mọi luật sinh dạng **tuyến tính phải** (rightlinear): $A \rightarrow wB$ hoặc $A \rightarrow w$ với A, B là các biến đơn và w là chuỗi ký hiệu kết thúc (có thể rỗng); hoặc có dạng **tuyến tính trái** (left-linear): $A \rightarrow Bw$ hoặc $A \rightarrow w$ thì G là văn phạm loại 3 hay còn được

Bài giảng môn học: Ngôn ngữ hình thức và Otomat

gọi là **văn phạm chính quy RG** (Regular Grammar)

Ngôn ngữ của lớp văn phạm này được gọi là ngôn ngữ chính quy (RL)

Ký hiệu : L_0, L_1, L_2, L_3 là các lớp ngôn ngữ sinh ra bởi các văn phạm loại 0, 1, 2, 3 tương ứng.

Ta có : $L_3 \subset L_2 \subset L_1 \subset L_0$ và các bao hàm thức này là nghiêm ngặt.

1.6. Các ví dụ về văn phạm

Cho bảng chữ cái $\Sigma = \{a_1, a_2, \dots, a_n\}$. Xây dựng văn phạm sinh các ngôn ngữ sau:

VD 1. $L = \Sigma^*$

VD 2. $L = \Sigma^+$

VD 3. $L =$ Các xâu có độ dài chẵn dương trên Σ

VD 4. $L =$ Các xâu có độ dài chẵn trên Σ

VD 5. $L =$ Các xâu có độ dài lẻ trên Σ

VD 6. Cho bảng chữ cái $\Sigma = \{0, 1, 2\}$. ây dựng văn phạm G sinh ngôn ngữ L như sau:

$L =$ gồm các từ bắt đầu bằng 011, chứa từ con 11211 và kết thúc bằng 1210

BÀI TẬP

Bài 1. Cho các ngôn ngữ sau: $L_1 = \{a, ab, abb, aabbb, abbaa, aaaabb, ababba\}$

$L_2 = \{\wedge, ab, bb, aba, bbb, aabb\}$

Tìm các ngôn ngữ hợp, giao, tích ghép, lặp của L_1 và L_2 . Tìm ngôn ngữ chia trái, chia phải của L_1 cho L_2 . Tìm ngôn ngữ soi gương của L_1 .

Bài 2. Cho bảng chữ cái $\Sigma = \{a, b, c\}$. Xây dựng văn phạm sinh ngôn ngữ

$L = \{a^n b^{2n+1} c^k \in \Sigma^* \mid n > 0, k \geq 0\}$

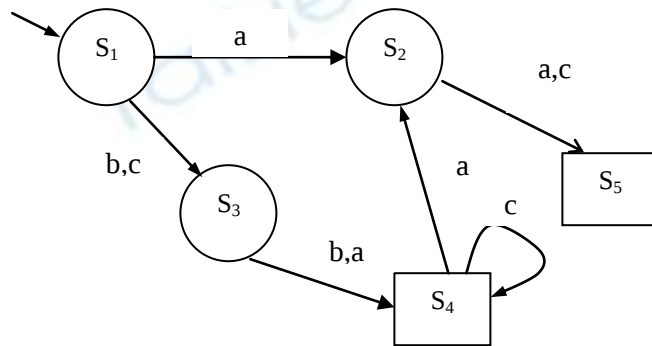
CHƯƠNG II: NGÔN NGỮ CHÍNH QUY VÀ OTOMAT HỮU HẠN

2.1. Nguồn và ngôn ngữ được sinh bởi nguồn

2.1.1. Định nghĩa: Nguồn là một đa đồ thị có hướng và có khuyên, có một đỉnh tách ra làm đỉnh vào, kí hiệu là ($\rightarrow O$), một tập con các đỉnh mà mỗi đỉnh này được gọi là một đỉnh ra hay đỉnh kết và đặt trong một ô chữ nhật ($\square$), đồng thời trên mỗi cung ghi một ký hiệu thuộc bảng chữ cái $\Sigma \cup \{\wedge\}$.

Cung trên đó ghi từ rỗng gọi là cung rỗng hay cung trống và thường người ta không ghi từ rỗng trên các cung này. Cung trên đó ghi ký hiệu thuộc Σ được gọi là cung cốt yếu. Đỉnh kết yếu là đỉnh có cung cốt yếu đi tới.

Ví dụ về nguồn :



Hình 1.1. Nguồn I_1

2.1.2. Nguồn đơn định

Nguồn I được gọi là nguồn đơn định nếu nó không chứa cung rỗng và hai cung tùy ý xuất phát từ một đỉnh đều được ghi bằng hai ký hiệu khác nhau. Nguồn I_1 là nguồn đơn định.

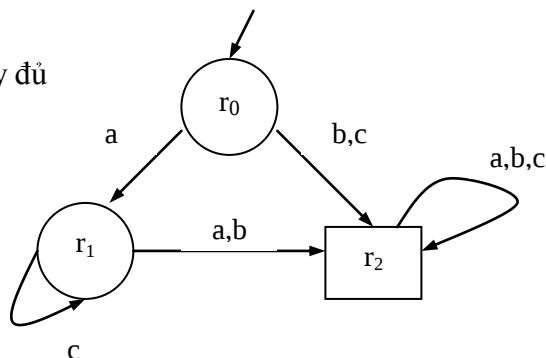
2.1.3. Nguồn đầy đủ

Nguồn đầy đủ là nguồn mà từ một đỉnh các cung đi ra phải chứa đầy đủ các ký hiệu $\in$ bảng chữ cái Σ .

2.1.4. Nguồn đơn định và đầy đủ

Là nguồn vừa đơn định vừa đầy đủ

Ví dụ :



Hình 1.2. Nguồn đơn định và đầy đủ I_2

2.1.5. Từ được sinh bởi nguồn

- Từ được sinh bởi nguồn là dãy các ký hiệu của bảng chữ cái Σ nằm trên các cung của nguồn đi từ đỉnh vào và đến một trong những đỉnh kết.

- Ngôn ngữ được sinh bởi nguồn I là tập hợp các từ được sinh bởi nguồn I , ký hiệu là $N(I) = \{x \in \Sigma^* \mid x \in N_I(v(I), s), s \in F(I)\}$.

Trong đó $v(I)$: đỉnh vào

$F(I)$: Tập đỉnh kết

- Theo nguồn trên ta xác định ngôn ngữ được sinh bởi nguồn I

$$N_I(s_1, s_3) = \{b, c\}$$

$$\begin{aligned} N_I(s_1, s_4) &= N_I(s_1, s_3) \cdot \{b, a\} \cdot c^* = \{b, c\} \cdot \{b, a\} \cdot \{c^s \mid s = 0, 1, 2, \dots\} \\ &= \{bbc^s, bac^s, cbc^s, cac^s \mid s = 0, 1, 2, \dots\} \end{aligned}$$

$$\begin{aligned} N_I(s_1, s_2) &= \{a\} \cup N_I(s_1, s_4) \cdot \{a\} = \{a\} \cup \{bbc^s a, bac^s a, cbc^s a, cac^s a\} \\ &= \{bbc^s a, bac^s a, cbc^s a, cac^s a \mid s = 0, 1, 2, \dots\} \end{aligned}$$

$$N_I(s_1, s_5) = N_I(s_1, s_2) \cdot \{a, c\} =$$

$$bbc^s aa, bac^s aa, cbc^s aa, cac^s aa, aa, bbc^s ac, bac^s ac, cbc^s ac, cac^s ac, ac \mid s = 0, 1, 2, \dots\}$$

$\Rightarrow$ Ngôn ngữ được sinh bởi nguồn I_1 là :

$$\begin{aligned} N(I_1) &= N_I(s_1, s_4) \cup N_I(s_1, s_5) = \{bbc^s, bac^s, cbc^s, cac^s \mid s = 0, 1, 2, \dots\} \cup \{bbc^t aa, bac^t aa, cbc^t aa, \\ &cbc^t aa, aa, bbc^t ac, bac^t ac, cbc^t ac, cac^t ac, ac \mid t = 0, 1, 2, \dots\} = \{aa, ac, bbc^s, bac^s, cbc^s, cac^s, bbc^t aa, \\ &bac^t aa, cbc^t aa, cbc^t aa, bbc^t ac, bac^t ac, cbc^t ac, cac^t ac \mid s, t = 0, 1, 2, \dots\} \end{aligned}$$

2.2. Các phép toán trên nguồn

2.2.1. Phép đơn định hóa

Giả sử nguồn I chưa đơn định và đầy đủ trên bảng chữ cái Σ , cần xây dựng nguồn K đơn định, đầy đủ tương đương với nó.

Thuật toán đơn định hoá nguồn như sau:

1) Đối với ký hiệu a tùy ý thuộc Σ và đỉnh s tùy ý của nguồn I xác định tập:

$$T_1(s, a) = \{u \in D(I) \mid a \in N(s, u)\}.$$

Đối với tập con M tùy ý của tập $D(I)$ và mọi ký hiệu $a \in \Sigma$ xác định tập:

$$H_1(M, a) = \bigcup_{s \notin M} T_1(s, a).$$

2) Xây dựng nguồn đơn định đầy đủ K

a. Đỉnh và cung:

Bài giảng môn học: Ngôn ngữ hình thức và Otomat

Đỉnh vào của K: $v(K) = \{v(I)\}$.

Đối với mọi đỉnh $M \in A(K)$ đã được xác định, ta xác định đỉnh $H_1(M,a)$ và kẻ một cung trên đó ghi ký hiệu a đi từ đỉnh M sang đỉnh $H_1(M,a)$.

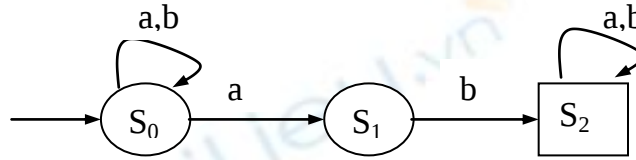
b. Tập đỉnh kết của K :

$$F(K) = \{M \in A(K) \mid M \cap F(I) \neq \emptyset\}$$

c. Với cách xây dựng như trên có nguồn K tương đương với nguồn I và K đơn định, đầy đủ.

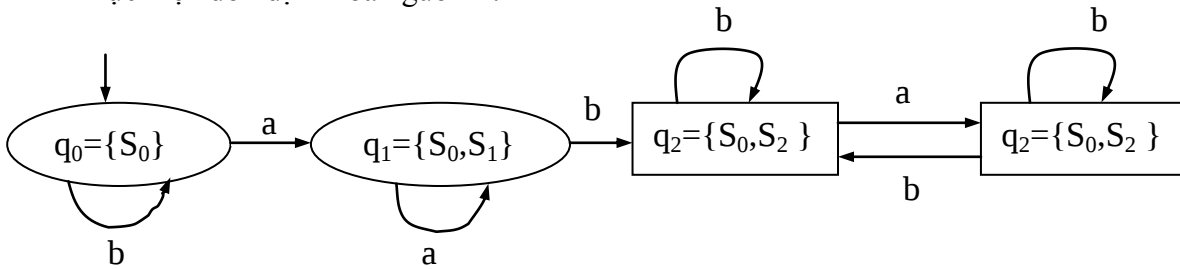
Chú ý: Nếu ta loại tập rỗng ra khỏi $A(K)$ thì nguồn K nhận được đơn định, nhưng không đầy đủ.

Ví dụ : Cho nguồn chưa đơn định và đầy đủ I :

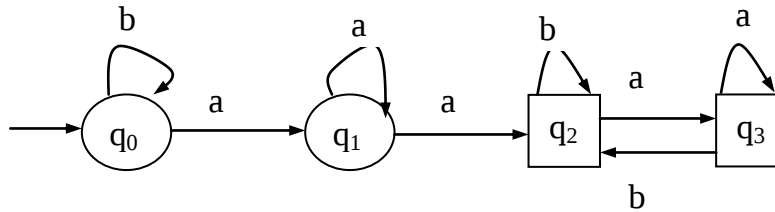


Hình 1.3. Nguồn I chưa đơn định đầy đủ

Thực hiện đơn định hóa nguồn I :



Nguồn đã đơn định đầy đủ, tương đương với I là :



Hình 1.4. Nguồn I' đơn định đầy đủ

2. 2. 2. Phép lấy phần bù

Giả sử I là nguồn tùy ý trên bảng chữ cái Σ . Để xây dựng nguồn sinh ngôn ngữ phần bù của ngôn ngữ do I sinh ra ta thực hiện một thuật toán gồm các bước sau:

1. Nếu I chưa đơn định và đầy đủ, ta xây dựng nguồn K đơn định, đầy đủ tương đương với I, tức là $N(I) = N(K)$.

2. Trên nguồn K ta đổi các đỉnh kết thành không kết và không kết thành kết. Khi đó nguồn nhận được sẽ sinh ra ngôn ngữ phản bù của ngôn ngữ do nguồn K sinh ra.

$$\text{Vậy } N(M) = C_{\Sigma} N(K) = C_{\Sigma} N(I).$$

2.2.3. Phép hợp nguồn

Giả sử I_1 và I_2 là các nguồn trên bảng chữ cái Σ . Cần xây dựng nguồn I sinh ngôn ngữ là hợp của hai ngôn ngữ do I_1 và I_2 sinh ra, tức là $N(I) = N(I_1) \cup N(I_2)$. Nguồn I được gọi là nguồn hợp của các nguồn I_1 và I_2 .

Thuật toán:

Để xây dựng nguồn I hợp của hai nguồn I_1 và I_2 , ta giữ nguyên cấu trúc của I_1 và I_2 , thêm vào một đỉnh mới (không phụ thuộc I_1 cũng như I_2) và thừa nhận đỉnh này là đỉnh vào của nguồn I, đồng thời từ đỉnh mới thêm vào kẻ các cung rỗng đi tới các đỉnh $v(I_1)$ và $v(I_2)$.

Tập đỉnh kết của nguồn I là $F(I) = F(I_1) \cup F(I_2)$.

Với cách xây dựng trên dễ dàng nhận thấy rằng: $N(I) = N(I_1) \cup N(I_2)$.

Chú ý: Ta có thể mở rộng phép hợp trên cho s ($s > 2$) nguồn tùy ý.

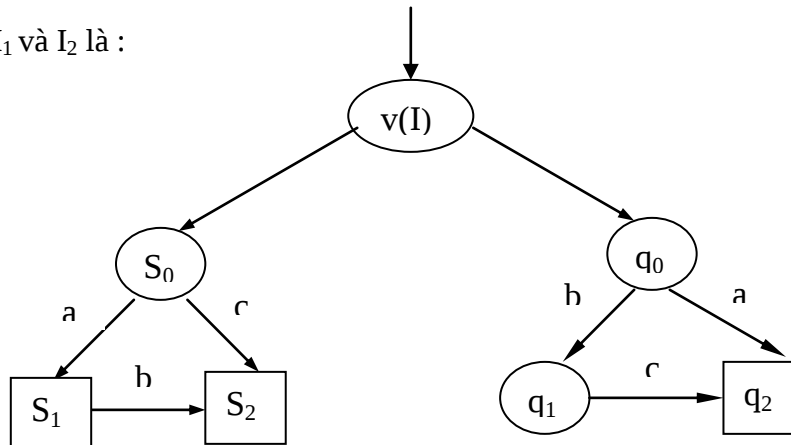
Định lý: Đối với s ($s > 1$) nguồn tùy ý $I_1, I_2, \dots, I_n$ luôn luôn xây dựng được nguồn I, để nó sinh ra ngôn ngữ là hợp của các ngôn ngữ do $I_1, I_2, \dots, I_n$ sinh ra, tức là $N(I) = \bigcup_{i=1}^s N(I_i)$.

Cho 2 nguồn I_1 và I_2 :



Hình 1.5. Nguồn I_1 và I_2

Hợp của I_1 và I_2 là :



Hình 1.6. Hợp của I_1 và I_2

2.2.4. Phép giao nguồn

Giả sử có các nguồn I_1, I_2 . Cần xây dựng nguồn I , sinh ngôn ngữ là giao của các ngôn ngữ do I_1, I_2 sinh ra, tức là $N(I) = N(I_1) \cap N(I_2)$. Nguồn I được gọi là nguồn giao của các nguồn I_1, I_2 .

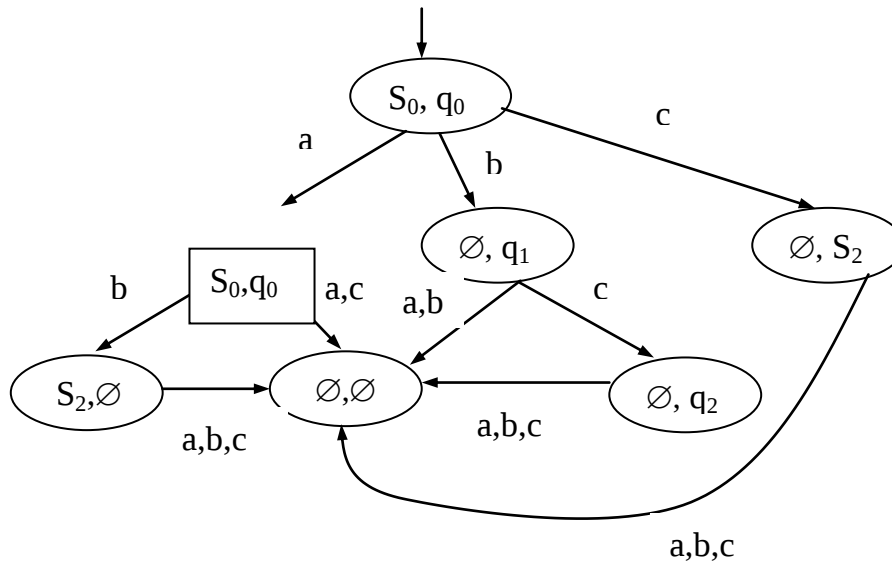
Thuật toán xây dựng nguồn giao I :

Để có nguồn giao I của các nguồn I_1, I_2 ta xác định đỉnh và cung của nó bằng phép quy nạp như sau:

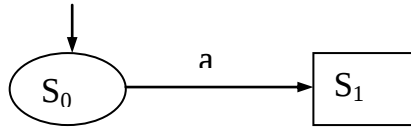
1. Đỉnh vào của nguồn I : $v(I) = \{v(I_1), v(I_2)\}$;
2. Giả sử $B = (P, Q)$, trong đó $P \subseteq D(I_1)$, $Q \subseteq D(I_2)$ là các đỉnh tùy ý đã được xác định và a là ký hiệu bất kỳ thuộc Σ . Khi đó xác định đỉnh $C = (H_{I_1}(P, a), H_{I_2}(Q, a))$, đồng thời kẻ một cung trên đó ghi ký hiệu a , đi từ đỉnh B sang C .
3. Đỉnh $D = (S, R)$ được thừa nhận là đỉnh kết của nguồn I ($D \in F(I)$) khi và chỉ khi $S \cap F(I_1) \neq \emptyset$ và $R \cap F(I_2) \neq \emptyset$.

Với cách xây dựng trên nguồn I sinh ngôn ngữ là giao của các ngôn ngữ do I_1, I_2 sinh ra.

Lấy ví dụ với 2 nguồn I_1 và I_2 (hình 1.5)



Nguồn giao của chúng là :



Hình 1.7. Nguồn giao của I_1 và I_2

2.2.5. Phép tích ghép hai nguồn

Đối với hai nguồn I_1, I_2 tùy ý, cần xây dựng nguồn I , sinh ngôn ngữ là tích ghép của các ngôn ngữ do I_1, I_2 sinh ra, tức là $N(I) = N(I_1).N(I_2)$.

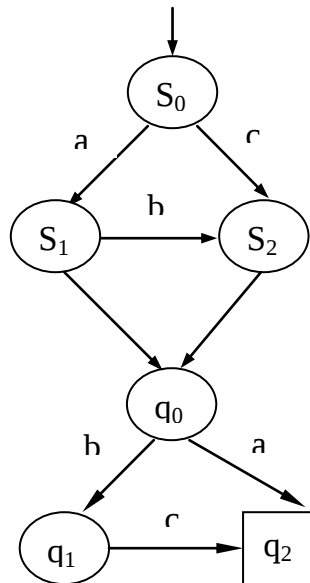
Nguồn I được gọi là nguồn tích ghép của nguồn I_1, I_2 .

Thuật toán xây dựng nguồn tích ghép:

Để xây dựng nguồn tích ghép (I) của các nguồn I_1, I_2 , ta giữ nguyên cấu trúc của I_1, I_2 thừa nhận đỉnh vào I_1 là đỉnh vào của nguồn I ($v(I) = v(I_1)$), các đỉnh kết của nguồn I_2 là đỉnh kết của nguồn I ($F(I) = F(I_2)$), đồng thời từ mỗi đỉnh kết của nguồn I_1 kẻ một cung rỗng đi tới đỉnh vào của nguồn I_2 . Đồng thời chuyển các đỉnh kết của I_1 thành đỉnh thường.

Với cách xây dựng như trên nguồn I , sinh ngôn ngữ là tích ghép của các ngôn ngữ do I_1, I_2 sinh ra.

Tích ghép của hai nguồn I_1 và I_2 (hình 1.5) :



**Hình 1.8. Tích ghép 2
nguồn I_1 và I_2**

2.2.6. Phép lặp

Đối với nguồn I_1 tùy ý, cần xây dựng nguồn I_1 sinh ngôn ngữ là lặp của ngôn ngữ do I_1 sinh ra. Nguồn I_1 được gọi là nguồn lặp của nguồn I .

Thuật toán xây dựng nguồn lặp I :

Để xây dựng nguồn I , sinh ngôn ngữ là lặp của ngôn ngữ do nguồn I_1 sinh ra, ta giữ nguyên

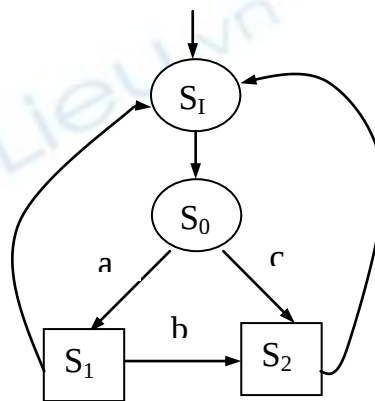
Bài giảng môn học: Ngôn ngữ hình thức và Otomat

cấu trúc của I_1 , thêm vào một đỉnh mới và thừa nhận đỉnh này là đỉnh vào đồng thời là đỉnh kết duy nhất của nguồn I . Từ đỉnh mới kẻ thêm một cung rỗng đi tới đỉnh $v(I_1)$, đồng thời từ mỗi đỉnh kết của nguồn I_1 kẻ một cung rỗng đi tới đỉnh mới thêm.

Nguồn sinh ngôn ngữ là lặp cắt của ngôn ngữ được sinh bởi I_1 chỉ khác với nguồn trên ở chỗ: đỉnh mới thêm chỉ được thừa nhận là đỉnh vào, còn tập đỉnh kết của I_1 được thừa nhận là đỉnh kết của nguồn mới.

Với cách xây dựng trên nguồn lặp (nguồn lặp cắt I) sinh ngôn ngữ là lặp (lặp cắt) của ngôn ngữ do I_1 sinh ra, tức là $N(I) = N(I_1)^* (N(I) = N(I_1)^*)$.

Nguồn lặp của nguồn I_1 :



Hình 1.9. Nguồn lặp của I_1

2.2.7. Phép soi gương nguồn

Đối với nguồn I_1 tùy ý, cần xây dựng nguồn I sinh ngôn ngữ soi gương của ngôn ngữ do I_1 sinh ra. Nguồn I được gọi là nguồn soi gương của nguồn I_1 .

Thuật toán xây dựng nguồn soi gương:

Để được nguồn I sinh ngôn ngữ soi gương của ngôn ngữ do I_1 sinh ra ta thực hiện các bước sau:

1) Thêm vào một đỉnh mới, thừa nhận nó là đỉnh vào của nguồn I , đồng thời từ đỉnh mới thêm ($v(I)$) kẻ các cung rỗng đi tới đỉnh kết của nguồn I_1 .

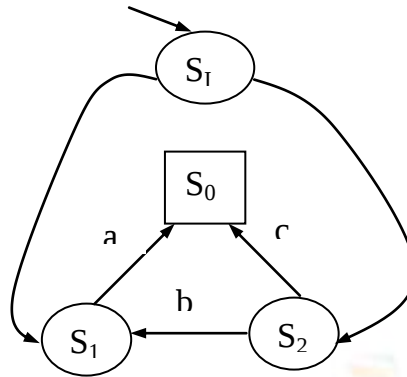
2) Thừa nhận đỉnh vào của I_1 ($v(I_1)$) là đỉnh kết duy nhất của nguồn I .

$$F(I) = \{v(I_1)\};$$

3) Đối với mỗi cung của nguồn I_1 , giữ nguyên ký hiệu ghi trên nó, nhưng đổi chiều ngược lại.

Với cách xây dựng trên ta có : $N(I) = N(I_1)$.

Nguồn soi gương của I_1 :



Hình 1.10. Nguồn soi gương của I_1

2.2.8. Phép chia trái

Giả sử I_1, I_2 là các nguồn trên bảng chữ cái Σ . Cần xây dựng nguồn K sinh ngôn ngữ là thương bên trái của các ngôn ngữ do I_1, I_2 sinh ra. Nguồn K được gọi là nguồn thương bên trái của các nguồn I_1, I_2 .

Thuật toán xây dựng nguồn thương bên trái:

- 1) Xây dựng nguồn giao I của các nguồn I_1, I_2 .
- 2) Thêm vào một đỉnh mới và thừa nhận đỉnh này là đỉnh vào của nguồn K . ký hiệu bằng $v(K)$;
- 3) Tập đỉnh kết: $F(K) = F(I_1)$;
- 4) Giả sử $Q = (S, T)$ là một đỉnh tùy ý của nguồn giao I . Khi đó đối với mỗi đỉnh $x \in S$ ($S \subseteq A(I_1)$) từ đỉnh vào của nguồn K ($v(K)$) có cung đi tới x khi và chỉ khi $T \cap F(I_2) \neq \emptyset$.

Với cách xây dựng này ta có: $N(K) =$

$N(I_1)$
/
 $N(I_2)$

2. 2. 9. Phép chia phải

Giả sử I_1, I_2 là các nguồn trên bảng chữ cái Σ . Cần xây dựng nguồn K sinh ngôn ngữ là thương bên phải của các ngôn ngữ do I_1, I_2 sinh ra. Nguồn K được gọi là nguồn thương bên phải của các nguồn I_1, I_2 .

Thuật toán xây dựng nguồn thương bên phải:

- 1) Dựa vào cấu trúc của I_1 đối với mỗi $s_1 \in D(I_1)$ xây dựng nguồn I_{s_1} có đỉnh vào là s_1 ($v(I_{s_1}) = s_1$) và $F(I_{s_1}) = F(I_1)$