

I. LÝ THUYẾT.....	4
1. Nội dung và mối quan hệ mật thiết của ba lĩnh vực nghiên cứu của Lý thuyết tính toán. (cuối trang 20 – đầu trang 22) .....	4
2. Sự ra đời và phát triển của Lý thuyết độ phức tạp tính toán. (cuối trang 22 – đầu trang 24) .....	4
3. Các điểm cơ bản trong cách tiếp cận bài toán, có ví dụ minh họa (trang 25 – gần hết trang 26) .....	5
4. Cách xác lập tương ứng bài toán tìm kiếm và bài toán quyết định, có ví dụ minh họa (trang 27 –trang 28).....	6
5. Ngôn ngữ biểu diễn bài toán quyết định, ví dụ dùng xâu liên kết để biểu diễn cho đồ thị có hướng nào đó (trang 29 – đầu trang 31) .....	7
6. Ngôn ngữ đặc trưng của bài toán quyết định và Mô hình tính toán đoán nhận ngôn ngữ (trang 31 – đầu trang 32). .....	8
7. Mô tả cấu tạo và nguyên tắc hoạt động của máy Turing (trang 36 – đầu trang 38) .....	9
8. Máy Turing chấp nhận đầu vào; Ngôn ngữ của máy Turing; Ngôn ngữ đoán nhận được; Ngôn ngữ khẳng định được; Thuật toán. Luận đề Church-Turing. ....	9
9. Bài toán chấp nhận (cuối trang 85- đầu trang 86) .....	11
10. Quy chuẩn về thời gian hoạt động của máy Turing (mục 2.1.1, không cần ví dụ) ..	11
11. Định nghĩa độ phức tạp thời gian của máy Turing tắt định dừng và máy Turing không tắt định dừng, có vẽ hình ảnh minh họa (cuối trang 121- đầu trang 122). ..	12
12. Nội dung phân tích thuật toán (mục 2.1.2, không cần ví dụ) .....	13
13. Nội dung các định lý về quan hệ thời gian giữa các loại máy (mục 2.1.3, không cần chứng minh) .....	14
14. Thời gian đa thức (cuối trang 138- đầu trang 140) .....	14
15. Định nghĩa lớp phức tạp Time, lớp P, vì sao lớp P đóng vai trò trung tâm trong lý thuyết độ phức tạp tính toán và là một lớp rất quan trọng (cuối trang 140- đầu trang 141). .....	15

16. Định nghĩa lớp phức tạp $NTime$ , lớp $NP$ . Trình bày một bài toán cụ thể thuộc lớp $NP$ là bài toán đường đi Hamilton trong đồ thị có hướng (cuối trang 154- đầu trang 156). .....	16
17. Định nghĩa về: độ phức tạp không gian của máy Turing tắt định 1 băng dừng, máy Turing tắt định nhiều băng dừng, máy Turing không tắt định dừng. Định lý về mối quan hệ giữa thời gian và không gian (định lý 3.2.2). .....	17
Bài tập .....	18
45	Má
y Turing 1 băng đoán nhận ngôn ngữ $0^i1^j$ .....	18
TH: $i = j; j > 0$ .....	18
TH: $i = j+1; j > 0$ .....	21
TH: $i = j + 2; j > 0$ .....	22
TH: $j = i+1; i > 0$ .....	25
TH: $j = i+2; i > 0$ .....	28
TH: $i > j; j > 0$ .....	29
TH: $i > j+1; j > 0$ .....	31
Máy Turing 2 băng đoán nhận .....	33
TH: $w\#w, w \in \{0, 1\}^+$ .....	33
Máy turing đoán nhận xâu vào $w\#w$ trên bộ chữ $\{0, 1\}$ được xây dựng như sau: .....	34
TH: $w\#w^R, w \in \{0, 1\}^+$ .....	35
Máy turing đoán nhận xâu vào $w\#w^R$ trên bộ chữ $\{0, 1\}$ được xây dựng như sau: .....	35
TH: $w\#0w^R, w \in \{0, 1\}^+$ .....	36
Máy turing đoán nhận xâu vào $w\#0w^R$ trên bộ chữ $\{0, 1\}$ được xây dựng như sau: .....	36
TH: $w\#0w, w \in \{0, 1\}^+$ .....	38

Máy turing đoán nhận xâu vào $w#0w$ trên bộ chữ $\{0, 1\}$ được xây dựng như sau: .....	38
TH: $w#00w, w \in \{0, 1\}^+$ .....	39
Máy turing đoán nhận xâu vào $w#00w$ trên bộ chữ $\{0, 1\}$ được xây dựng như sau: .....	39
TH: $w#00w^R, w \in \{0, 1\}^+$ .....	41
Máy turing đoán nhận xâu vào $w#00w^R$ trên bộ chữ $\{0, 1\}$ được xây dựng như sau: .....	41
Máy Turing 1 bảng tính hàm $f(n, m)$ .....	43
TH: $f(n, m) = n * m$ .....	43
TH: $f(n, m) = n * m - m + 1$ .....	43
TH: $f(n, m) = n * m + m + 1$ .....	46
TH: $f(n, m) = n * m - n + 1$ .....	48
TH: $f(n, m) = n * m + n + 1$ .....	50
TH: $f(n, m) = n * m - m - 1$ .....	52
TH: $f(n, m) = n * m + m - 1$ .....	54
TH: $f(n, m) = n * m - n - 1$ .....	56
TH: $f(n, m) = n * m + n - 1$ .....	58

## I. LÝ THUYẾT

### ***1. Nội dung và mối quan hệ mật thiết của ba lĩnh vực nghiên cứu của Lý thuyết tính toán. (cuối trang 20 – đầu trang 22)***

Ba lĩnh vực trọng tâm có tính truyền thống của **Lý thuyết tính toán** là **Lý thuyết ô tô mát**, **Lý thuyết về khả năng tính toán** và **Lý thuyết độ phức tạp tính toán**. **Lý thuyết ô tô mát** đề cập đến việc xây dựng các mô hình toán học về tính toán. Mục tiêu của **Lý thuyết về khả năng tính toán** là phân chia các bài toán thành các lớp các bài toán giải được và lớp các bài toán không giải được. **Lý thuyết độ phức tạp tính toán** phân chia các bài toán giải được thành các lớp khác nhau theo mức độ khó khăn khi giải chúng.

Về lý thuyết, quá trình giải bài toán bao gồm việc lập mô hình toán học cho bài toán khi cần thiết và dựa trên mô hình đó, xây dựng phương pháp giải (thuật toán giải). Tuy nhiên, để bài toán giải được một cách thực tế, việc lập mô hình toán học cho bài toán cũng như việc xây dựng thuật toán giải cho nó đều phải thích hợp với những trang thiết bị tính toán hiện có. Những trang thiết bị này có được là nhờ các thành tựu của công nghệ mang lại, mà cơ sở lý luận của nó dựa trên những kết quả nghiên cứu của **Lý thuyết ô tô mát**.

Đôi khi trong quá trình giải bài toán, dù đã rất cố gắng nhưng ta vẫn không thể tìm được một thuật toán giải. Khó khăn này có thể chính là do bản chất phức tạp của bài toán. Việc chứng tỏ không có thuật toán giải bài toán là phận sự của **Lý thuyết về khả năng tính toán**. Để làm được điều đó cần phải có một định nghĩa chính xác về thuật toán thay thế cho khái niệm theo định nghĩa trực giác ta thường dùng.

Trong trường hợp ta đã xây dựng được thuật toán giải, trên thực tế đôi khi để nhận được lời giải thỏa đáng lại rất gian nan, dù ta được cung cấp đầy đủ các trang thiết bị tiên tiến nhất. Như vậy, giữa “giải được về mặt lý thuyết” và “giải được một cách thực tế” có sự khác biệt đáng kể, trong đó “giải được một cách thực tế” là giải được trên máy tính điện tử trong phạm vi khả năng về thời gian và bộ nhớ mà máy có thể có được. **Lý thuyết độ phức tạp tính toán** giúp lý giải sự khác biệt này.

### ***2. Sự ra đời và phát triển của Lý thuyết độ phức tạp tính toán. (cuối trang 22 – đầu trang 24)***

Sau khi máy tính điện tử ra đời, việc thực hiện thuật toán để giải tiếp bài toán không hoàn toàn là một việc đơn giản. Khi thực hiện thuật toán trên máy tính, có nhiều bài toán giải được một cách dễ dàng và cũng có không ít bài toán khó giải, thậm chí không giải được, dù về mặt lý thuyết chúng hoàn toàn giải được bằng nhiều thuật toán khác nhau. VD, với bài toán sắp xếp các đối tượng theo một thứ tự đã định, ta có thể giải bài toán với hàng triệu đối tượng một cách nhanh chóng chỉ trên một máy tính nhỏ. Trong khi đó, với bài toán lập lịch biểu cho một trường đại học thì vấn đề lại trở nên khá phức tạp. Việc bố trí phòng học cho các lớp cần đảm

bảo không có hai lớp được bố trí vào cùng một phòng học tại cùng một thời điểm. Để có được một thời gian biểu cho một trường đại học, gồm khoảng 100 lớp với một số thầy cô và một lượng phòng học nào đó, có khi đòi hỏi một thời gian thực hiện tính toán đến hàng thế kỷ ngay trên cả siêu máy tính. Vậy là bài toán lập lịch biểu không thể giải được một cách thực tế.

Như vậy, trong hoạt động thực tiễn ta thường gặp những bài toán dễ và những bài toán khó. Ngoài ra, một bài toán cũng có thể giải được bằng nhiều thuật toán xấu tốt khác nhau, và với một thuật toán thì có thể trong trường hợp này nó cho ta kết quả nhanh còn trong trường hợp khác nó lại cho kết quả chậm. Hơn nữa, giữa *giải được về mặt thuyết* và *giải được một cách thực tế* đối với nhiều bài toán cũng có những khác biệt đáng kể.

Cuối thập niên 60 của thế kỷ XX, **Lý thuyết độ phức tạp tính toán** hình thành. Dựa trên những khám phá về sự phụ thuộc giữa “*kích cỡ của bài toán*” với thời gian thực hiện thuật toán cũng như với dung lượng bộ nhớ mà máy cần sử dụng trong quá trình tính toán, lý thuyết này tiến hành xem xét “độ phức tạp thời gian” và “độ phức tạp không gian” của thuật toán như những hàm phụ thuộc vào “kích cỡ” của bài toán. Nhờ đó ta có được một cách nhìn thống nhất về độ phức tạp của các thuật toán.

Tuy vẫn chưa có được một lý giải thỏa đáng cho những hiện tượng phổ biến đã nêu, nhưng **Lý thuyết độ phức tạp tính toán** đã có một bước tiến đáng kể với nhiều kết quả phong phú có ý nghĩa nhất định về lý thuyết cũng như ứng dụng. Nhờ tiến hành khảo sát các bài toán trong một khuôn khổ chung, **lý thuyết độ phức tạp tính toán** cho ta những hiểu biết khái quát về tính phức tạp của một lớp rộng rãi các bài toán, đồng thời cũng gợi mở những phương hướng giải quyết đối với các bài toán cụ thể mà ta gặp phải.

### **3. Các điểm cơ bản trong cách tiếp cận bài toán, có ví dụ minh họa (trang 25 – gần hết trang 26)**

Các bài toán đa phần là **dạng**: bài toán quyết định hoặc bài toán tìm kiếm (nói riêng là dạng bài toán tối ưu). Khi các bài toán chưa được **phát biểu** bằng ngôn ngữ toán học, phải dịch bài toán sang 1 ngữ cảnh toán học thích hợp

VD - Bài toán xếp ba lô: Cho 1 lô hàng hóa gồm các gói hàng, mỗi gói đều có khối lượng cùng giá trị cụ thể. Cho 1 chiếc ba lô. Hãy chọn từ lô này một số gói hàng nào đó và xếp đầy vào ba lô, nhưng không được quá, sao cho thu được 1 giá trị lớn nhất có thể

- Dữ kiện: cho 2 dãy số nguyên dương  $s_1, s_2, \dots, s_n, S$  và  $v_1, v_2, \dots, v_n$
- Yêu cầu: Tìm 1 tập con  $I \subseteq \{1, 2, \dots, n\}$  sao cho  $i \in I \Rightarrow s_i \leq S$  và  $i \in I \Rightarrow v_i \rightarrow \max$

Đầu tiên, cần lập được **mô hình** thích hợp cho bài toán. Sau đó, dựa trên mô hình toán học của bài toán để **đề xuất 1 phương pháp giải**, hay lý tưởng hơn là **xây**

**dựng 1 thủ tục** chặt chẽ xử lý 1 cách hiệu quả các thông tin liên quan đến mỗi dữ kiện bài toán, để từ đó thu được nghiệm tương ứng với dữ kiện. Cuối cùng, cần tiến hành **phân tích và đánh giá hiệu quả** của **cách giải** bài toán cũng như **khả năng hiện thực hóa** trên những trang thiết bị tính toán hiện có

Để có được những **kết luận** hay những **kết quả nghiên cứu mang tính khái quát**, các bài toán cần được xem xét trong **1 khuôn khổ chung**. Điều này giúp ta có 1 cách nhìn **thống nhất** đối với vấn đề phức tạp của các bài toán, đồng thời cho ta **hiểu biết khái quát và những hướng dẫn** cần thiết trong các **tình huống cụ thể**. Việc nghiên cứu này **được giới hạn trong phạm vi các bài toán quyết định** do chúng có một sự thể hiện khá tự nhiên dưới dạng ngôn ngữ hình thức rất thích hợp cho việc phân tích và xử lý thông tin bởi một mô hình tính toán chung.

#### ***4. Cách xác lập tương ứng bài toán tìm kiếm và bài toán quyết định, có ví dụ minh họa (trang 27 –trang 28)***

**Trước tiên cần phân biệt các loại bài toán mà ta xem xét.**

Một bài toán có thể được phát biểu thành hai phần tách biệt: Dữ kiện và Yêu cầu. Đối với Dữ kiện, ta cần xác định rõ tập dữ kiện của bài toán bao gồm những dữ kiện cụ thể nào. Yêu cầu thường có hai loại.

- Câu hỏi mà đối với mỗi dữ kiện bài toán chỉ cần trả lời “đúng” hoặc “sai”. Bài toán với câu hỏi như vậy là bài toán quyết định.
- yêu cầu Tìm kiếm nghiệm đối với dữ kiện bất kỳ cho trước. Bài toán với yêu cầu như vậy là bài toán tìm kiếm. Bài toán cực đại hóa và bài toán cực tiểu hóa là các bài toán tối ưu (thuộc lớp các bài toán tìm kiếm) với yêu cầu tìm kiếm nghiệm chấp nhận được với giá trị lớn nhất và nhỏ nhất tương ứng.

**Bài toán quyết định tương ứng với bài toán tìm kiếm** được xác định bằng cách thay Yêu cầu: “Tìm nghiệm đối với dữ kiện bất kỳ cho trước” bằng Câu hỏi: “Phải chăng tồn tại nghiệm đối với mỗi dữ kiện đã cho?” Trong trường hợp này tập dữ kiện không thay đổi.

VD, bài toán “Hãy tìm chu trình Hamilton trong mỗi đồ thị cho trước” **được tương ứng** với bài toán quyết định sau:

Dữ kiện. Cho một đồ thị  $G$ .

Câu hỏi: Phải chăng trong  $G$  có chu trình Hamilton?

**Bài toán quyết định tương ứng với bài toán tối ưu (Cực đại hóa ,cực tiểu hóa)** được xác định: Tập dữ kiện của nó được xây dựng bằng cách thêm vào mỗi dữ kiện của bài toán tối ưu một ranh giới  $B$  tùy ý, thuộc cùng miền xác định giá trị của nghiệm. Câu hỏi được phát biểu rằng, đối với mỗi dữ kiện như vậy, phải chăng có một nghiệm chấp nhận được với giá trị không nhỏ hơn  $B$ ?

VD, bài toán cực đại hóa MAX-KNAPSACK **được tương ứng** với bài toán quyết định sau:

Dữ kiện: Cho hai dãy số nguyên dương  $s_1, s_2, \dots, s_n, S$  và  $v_1, v_2, \dots, v_n, B$

Câu hỏi: Phải chăng có một tập con  $I \{1,2,...,n\}$  sao cho  
 $iI_S$  và  $iI_{ViB}$

Như vậy, mỗi BTTK hay mỗi bài tối ưu nói riêng đều có thể **được tương ứng** với BTQĐ thường là không khó hơn. Với bài toán tối ưu, do việc tính toán giá trị nghiệm khá dễ dàng nên BTQĐ tương ứng không thể khó hơn bài toán tối ưu ban đầu.

Cuối cùng, ta tìm cách đặc trưng mỗi bài toán quyết định bởi một ngôn ngữ hình thức nào đó bằng cách mã hóa từng dữ kiện bài toán bởi một từ trên bảng chữ thích hợp và xác định ngôn ngữ tương ứng với bài toán ấy. Việc mã hóa này có thể được thực hiện theo một quy trình chung với 1 vài ràng buộc cần thiết.

### **5. Ngôn ngữ biểu diễn bài toán quyết định, ví dụ dùng xâu liên kết để biểu diễn cho đồ thị có hướng nào đó (trang 29 – đầu trang 31)**

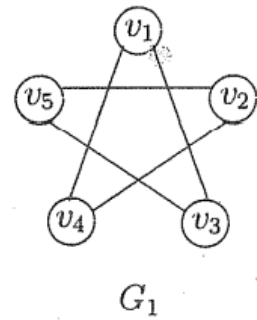
1 bài toán quyết định thường được **diễn tả** bằng nhiều ngôn ngữ hình thức khác nhau, theo những phép mã hóa khác nhau đối với các dữ kiện bài toán. **Để đảm bảo yêu cầu tối thiểu về tính súc tích đối với ngôn ngữ**, việc mã hóa cần thỏa mãn:

- **Từ mã** (xâu biểu diễn) mỗi dữ kiện bài toán phải ngắn gọn, và không được “độn thêm” những thông tin không cần thiết
- Các **số tham gia** trong dữ kiện bài toán cần được biểu diễn dưới dạng nhị phân hoặc theo 1 cơ số nào đó lớn hơn 1

Các phép mã hóa thích hợp có thể được xây dựng dựa trên một **sơ đồ mã hóa chuẩn**, các **dữ kiện** của bài toán được biểu diễn bởi các “xâu liên kết”, tức “xâu có cấu trúc”, trên bảng chữ chứa  $\Psi = \{0, 1, -, [, ], (, ), ,\}$  với ký tự cuối cùng là dấu phẩy “,”. **Xâu liên kết** được định nghĩa một cách đệ quy như sau:

- Biểu diễn **nhị phân** của một số nguyên  $k$  (có ký tự “—” ở phía trước nếu  $k$  là số âm) là một xâu liên kết biểu thị số nguyên  $k$ .
- Nếu  $x$  là một xâu liên kết biểu thị số nguyên  $k$ , thì  $[x]$  là một xâu liên kết biểu thị “**tên**” của đối tượng mang số hiệu  $k$ .
- Nếu  **$y_1, y_2, ..., y_m$**  là các xâu liên kết biểu thị các **đối tượng**  $Y_1, Y_2, ..., Y_m$ , thì  **$(y_1, y_2, ..., y_m)$**  là một xâu liên kết biểu thị **dãy**  $(Y_1, Y_2, ..., Y_m)$

Ta có thể biểu diễn các dữ kiện bài toán bởi các xâu liên kết trên một bảng chữ nào đó chứa các ký tự thuộc bảng chữ  $\Psi$  kể trên. Trong từng trường hợp cụ thể ta **có thể bổ sung những quy tắc thích hợp cần thiết**, nhưng vẫn phải đảm bảo tính chuẩn xác của việc biểu diễn.



VD:

trên bảng chữ gồm sáu ký tự  $\{0, 1, (, ), \#, \}$ , trong đó xâu trước ký tự  $\#$  biểu diễn tập đỉnh và xâu sau  $\#$  biểu diễn các cung của đồ thị. Thí dụ, đồ thị hình ngôi sao năm cánh  $G_1$  với tập đỉnh  $\{v_1, v_2, v_3, v_4, v_5\}$  (Hình 0.2) được biểu diễn bởi xâu liên kết

$$\langle G_1 \rangle = \bar{1}, \bar{2}, \bar{3}, \bar{4}, \bar{5} \# (\bar{1}, \bar{3})(\bar{1}, \bar{4})(\bar{2}, \bar{4})(\bar{2}, \bar{5})(\bar{3}, \bar{5}).$$

## 6. Ngôn ngữ đặc trưng của bài toán quyết định và Mô hình tính toán đoán nhận ngôn ngữ (trang 31 – đầu trang 32).

### \* Ngôn ngữ đặc trưng của bài toán quyết định

Giả sử là một bài toán quyết định với tập dữ kiện  $D$  và câu hỏi  $Q$  trên mỗi dữ kiện bài toán.

Khi đó,  $Q$  xác định một tính chất đối với mọi dữ kiện của bài toán, tức một hàm từ tập  $D$  đến tập các giá trị chân lý  $\{\text{ĐÚNG}, \text{SAI}\}$  như sau: Đối với mỗi dữ kiện  $d \in D$

- có  $Q(d) = \text{ĐÚNG}$  khi câu hỏi  $Q$  trên dữ kiện  $d$  được trả lời là “đúng”
- ngược lại,  $Q(d) = \text{SAI}$ .

Giả sử  $e$  là một phép mã hóa thích hợp nào đó đối với bài toán, theo đó, **mỗi dữ kiện bài toán được biểu diễn bởi một xâu liên kết trên bảng chữ**. Như vậy,  $e$  ánh xạ các dữ kiện bài toán thành các xâu thuộc  $\Sigma^*$ . Khi không cần lưu ý đến phép mã hóa  $e$ , ta ký hiệu  $\langle d \rangle = e(d)$  đối với mỗi  $d \in D$

Ta định nghĩa các ngôn ngữ sau:

$$L(D_\Pi) \stackrel{\text{def}}{=} \{ \langle d \rangle \mid d \in D_\Pi \},$$

$$L(\Pi) \stackrel{\text{def}}{=} \{ \langle d \rangle \mid d \in D_\Pi \ \& \ Q_\Pi(d) = \text{ĐÚNG} \}.$$

$L()$  diễn đạt nội dung của bài toán, được gọi là ngôn ngữ đặc trưng của, ký hiệu là **nghiêng**:

$$\Pi \stackrel{\text{def}}{=} L(\Pi) \subseteq L(D_\Pi) \subseteq \Sigma^*.$$

### \* Mô hình tính toán đoán nhận ngôn ngữ

Cho là một bài toán quyết định,  $L(D)$  là ngôn ngữ trên khu ng chữ biểu diễn tập dữ kiện của, **nghiêng** là ngôn ngữ tương ứng với bài toán. Giả sử tồn tại một



mô hình tính toán hình thức, mà khi xử lý trên mỗi xâu thuộc  $L(D)$ , hay thuộc  $*$  nói chung, có thể phân biệt được xâu nào thuộc **nghiêng** và xâu nào không. Tức là mô hình tính toán ấy có khả năng “nhận biết” hay “đoán nhận” ngôn ngữ **nghiêng**. Khi đó ta dễ dàng thu được lời giải của bài quyết định. Thay cho việc xây dựng thuật toán giải bài toán quyết định cho trước, ta có thể tìm kiếm một mô hình tính toán hình thức nào đó để đoán nhận ngôn ngữ tương ứng với bài ấy.

### **7. Mô tả cấu tạo và nguyên tắc hoạt động của máy Turing (trang 36 – đầu trang 38)**

a. Máy Turing là 1 mô hình **xử lý tin tự động**, mỗi dữ liệu đầu vào được biểu diễn dưới dạng 1 **từ** trên bảng chữ  $\Sigma$  nào đó. Quá trình hoạt động của máy diễn ra trong **thời gian rời rạc**  $t = 0, 1, 2, \dots$ . Máy bao gồm 1 bộ phận điều khiển, 1 băng vô hạn về phía phải, và 1 đầu đọc-ghi kết nối bộ phận điều khiển với băng

- Băng: được chia thành **các ô** bắt đầu từ trái qua phải, mỗi ô chứa 1 ký tự thuộc **bảng chữ  $\Gamma$**  nào đó. Bảng chữ này chứa  $\Sigma$ ,  $\emptyset$  (biểu diễn ô trống) và có thể chứa một vài ký tự cần thiết khác. Tại  $t = 0$ , dữ liệu đầu vào được ghi trên băng từ ô đầu tiên, còn lại là các ô trống
- Đầu đọc-ghi: tại mỗi thời điểm hoạt động của máy **soi xét** 1 ô trên băng, có nhiệm vụ đọc ký tự ở ô đó rồi thông báo cho bộ phận điều khiển để được chỉ dẫn thay ký tự đó bằng 1 ký tự khác, sau đó di chuyển sang ô bên cạnh. Tại  $t = 0$ , đầu đọc-ghi soi xét ô đầu tiên
- Bộ phận điều khiển: gồm 1 **tập hữu hạn các trạng thái  $Q$** , và tại mỗi thời điểm hoạt động nó chỉ ở 1 trạng thái. Tập  $Q$  chứa 3 trạng thái đặc biệt:  $q_0$  (trạng thái ban đầu),  $q_Y$  (trạng thái chấp nhận),  $q_N$  (trạng thái bác bỏ)

b. Nguyên tắc hoạt động:

- **Thời điểm đầu**, trạng thái của bộ phận điều khiển, việc nạp dữ liệu đầu vào và vị trí đầu đọc-ghi được xác định như đã nêu. Quá trình **xử lý tin** trên mỗi dữ liệu đầu vào được thực hiện theo các **lệnh** của máy. Tại **mỗi thời điểm** hoạt động, khi bộ phận điều khiển ở trạng thái  $q$  và đầu đọc-ghi đọc được ký tự  $x$  trên băng, nếu máy có lệnh với khúc đầu  $qx$ , thì máy thực hiện với nội dung nói trên. Ngược lại, máy dừng hoạt động. Việc thực hiện 1 lệnh của máy được gọi là **phép biến đổi** (step), hay **phép biến đổi cơ bản**. Quá trình được tiếp diễn cho đến khi máy có thể đi đến những **quyết định cuối cùng** đối với dữ liệu đầu vào. Dữ liệu đầu vào được **chấp nhận** hay **bác bỏ** tùy thuộc vào việc máy dừng ở trạng thái  $q_Y$  hay  $q_N$  khi quá trình xử lý kết thúc. Trong trường hợp khi **không đến được trạng thái kết thúc**, quá trình tính toán của máy trên đầu vào sẽ không bao giờ dừng

### **8. Máy Turing chấp nhận đầu vào; Ngôn ngữ của máy Turing; Ngôn ngữ đoán nhận được; Ngôn ngữ khẳng định được; Thuật toán. Luận đề Church-Turing.**

**\* Máy Turing chấp nhận đầu vào**

**Định nghĩa 1.1.5** Cho máy Turing  $M$  với bảng chữ vào  $\Sigma$ . Ta nói rằng máy Turing  $M$  **chấp nhận đầu vào** (accepts input)  $w$ , nếu  $q_0w \xrightarrow{M} uq_vv$ , nghĩa là khi tính toán trên từ vào  $w$ , máy  $M$  chuyển từ hình thái ban đầu đến hình thái chấp nhận.

Tập các từ vào mà máy  $M$  chấp nhận tạo thành ngôn ngữ chấp nhận được của  $M$ , được gọi là **ngôn ngữ của máy Turing** (language of a Turing machine)  $M$  và được ký hiệu bởi  $L_M$ .

**\* Ngôn ngữ của máy Turing**

**Định nghĩa 1.1.5** Cho máy Turing  $M$  với bảng chữ vào  $\Sigma$ . Ta nói rằng máy Turing  $M$  **chấp nhận đầu vào** (accepts input)  $w$ , nếu  $q_0w \xrightarrow{M} uq_vv$ , nghĩa là khi tính toán trên từ vào  $w$ , máy  $M$  chuyển từ hình thái ban đầu đến hình thái chấp nhận.

Tập các từ vào mà máy  $M$  chấp nhận tạo thành ngôn ngữ chấp nhận được của  $M$ , được gọi là **ngôn ngữ của máy Turing** (language of a Turing machine)  $M$  và được ký hiệu bởi  $L_M$ .

**\* Ngôn ngữ đoán nhận được**

**Định nghĩa 1.1.6** Ngôn ngữ  $L$  được gọi là **đ đoán nhận được theo Turing** (Turing-recognizable), hay đơn giản là **đ đoán nhận được** (recognizable),<sup>1</sup> nếu nó là ngôn ngữ chấp nhận được của một máy Turing nào đó, nghĩa là nếu tồn tại một máy Turing  $M$  sao cho  $L = L_M$ . Khi đó ta nói rằng “máy Turing  $M$  đoán nhận ngôn ngữ  $L$ ”, hay “ngôn ngữ  $L$  được đoán nhận bởi máy Turing  $M$ ”.

**\* Ngôn ngữ khẳng định được**

**Định nghĩa 1.1.7** Ngôn ngữ  $L$  được gọi là **khẳng định được** (decidable), hay cụ thể hơn là **khẳng định được theo Turing** (Turing-decidable)<sup>2</sup>, nếu nó được đoán nhận bởi máy quyết định  $M$  nào đó. Trong trường hợp ấy, ta nói rằng “máy Turing dừng  $M$  khẳng định ngôn ngữ  $L$ ”, hay “ngôn ngữ  $L$  được khẳng định bởi máy Turing dừng  $M$ ”; ngược lại, ngôn ngữ  $L$  được gọi là **không khẳng định được** (undecidable).

**\* Thuật toán**

Thuật toán là máy Turing dừng. :)

**\* Luận đề Church-Turing**

Thuật toán theo nghĩa trực giác đồng nhất với thuật toán máy Turing.

### 9. Bài toán chấp nhận (cuối trang 85- đầu trang 86)

Đầu tiên, ta xem xét bài toán kiểm tra xem liệu máy Turing có **chấp nhận một từ vào bất kỳ** cho trước. Bài toán này là **bài toán chấp nhận** đối với máy Turing, ký hiệu là ATM. Để chứng minh **tính không giải được** của bài toán, ta sẽ tìm cách chứng tỏ rằng ngôn ngữ tương ứng của nó là **không khẳng định được**, tức không tồn tại máy Turing dừng đoán nhận ngôn ngữ ấy. Ngôn ngữ tương ứng với bài toán này được ký hiệu là ATM và được xác định như sau:

$ATM = \{ \langle M, w \rangle \mid M \text{ là máy Turing và } M \text{ chấp nhận } w \}$

**Định lý 1.4.1** Ngôn ngữ ATM là không khẳng định được.

Lưu ý: **ATM là ngôn ngữ đoán nhận được** (theo Turing). Như vậy, lớp các ngôn ngữ đoán nhận được rộng hơn hẳn lớp các ngôn ngữ khẳng định được. Việc đòi hỏi máy Turing phải dừng trên mọi từ vào phần nào hạn chế các ngôn ngữ mà máy có thể đoán nhận. Máy Turing U sau đây đoán nhận ATM.

$U =$  “Trên đầu vào  $\langle M, w \rangle$ , trong đó  $M$  là một máy Turing và  $w$  là một từ:

1. **Mô phỏng** việc tính toán của  $M$  trên từ  $w$ .
2. Nếu vào bất cứ lúc nào máy  $M$  ở **trạng thái chấp nhận**, *chấp nhận*; nếu bao giờ  $M$  ở trạng thái bác bỏ, *bác bỏ*.”

### 10. Quy chuẩn về thời gian hoạt động của máy Turing (mục 2.1.1, không cần ví dụ)

Ta tiến hành xác định thời gian hoạt động của máy Turing như một hàm phụ thuộc vào độ dài của từ vào được dùng để biểu diễn dữ kiện bài toán. Việc khảo sát **thời gian trung bình** mà thuật toán cần sử dụng khi tính toán trong phạm vi nào đó, chẳng hạn như trong phạm vi các đầu vào cùng “kích cỡ”, là cần thiết. Theo quan điểm chung, việc xác định thời gian tính toán của thuật toán trong trường hợp xấu nhất là quan trọng. Nó cho ta thấy được lượng thời gian tối đa cần sử dụng khi tính toán trên những đầu vào với cùng “kích cỡ”. Trong trường hợp cụ thể, đối với máy Turing tất định  $M$  cho trước, lượng thời gian tối đa mà máy cần dùng khi tính toán trên mỗi từ vào độ dài  $n$ , ký hiệu bởi  $t_M(n)$ , được xác định như sau:

$$t_M(n) = \max\{t_M(w) \mid w \in \Sigma^n\},$$

trong đó  $\Sigma^n$  là tập các từ độ dài  $n$  trên bảng chữ  $\Sigma$ .

$t_M(n)$  thể hiện lượng thời gian máy  $M$  cần sử dụng khi tính toán trong TH xấu nhất. Với máy Turing tất định nhiều băng, quá trình tính toán trên mỗi từ vào được diễn ra theo cách chuyển đổi các hình thái của máy từ hình thái ban đầu đến hình thái kết thúc. Đối với máy Turing không tất định dừng  $N$ , quá trình tính toán trên mỗi từ vào  $w$  diễn ra theo một nhánh nào đó mà máy lựa chọn trong cây tính toán

$\uparrow N(w)$  của máy  $N$  trên từ vào  $w$ . Thời gian tính toán của  $N$  trên  $w$  được định nghĩa bởi độ dài của nhánh dài nhất trong cây  $\uparrow N(w)$ .

**11. Định nghĩa độ phức tạp thời gian của máy Turing tất định dừng và máy Turing không tất định dừng, có vẽ hình ảnh minh họa (cuối trang 121- đầu trang 122).**

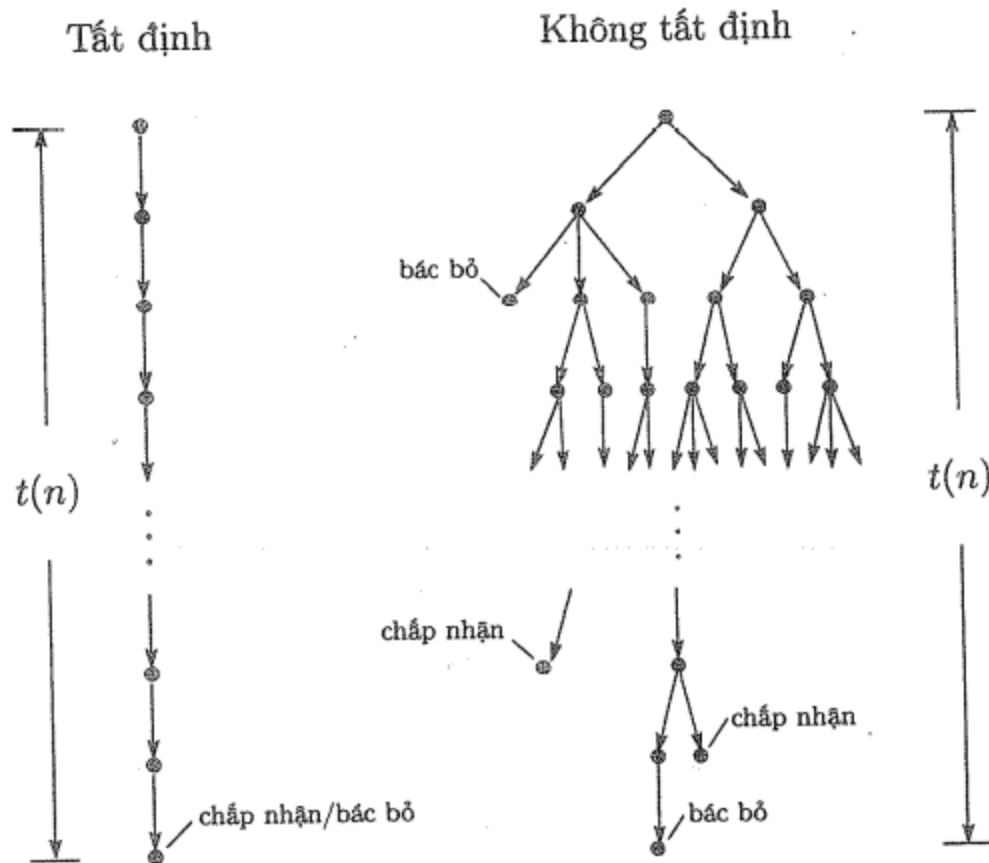
**Máy Turing tất định dừng**

**Định nghĩa 2.1.1** Cho  $M$  là một máy Turing tất định dừng (một băng hoặc nhiều băng). **Độ phức tạp thời gian** (time complexity) hay **thời gian hoạt động** (running time) của  $M$  là hàm  $t : \mathbb{N} \rightarrow \mathbb{N}$ , mà giá trị  $t(n)$  là số lần tối đa các phép biến đổi cơ bản được sử dụng trong quá trình tính toán của  $M$  trên bất cứ từ vào nào độ dài  $n$ . Nếu  $t(n)$  là độ phức tạp thời gian của  $M$  thì ta nói rằng  $M$  hoạt động trong thời gian  $t(n)$  hoặc  $M$  là **máy Turing thời gian  $t(n)$** . Ta thường dùng  $n$  để chỉ độ dài của từ vào.

**Máy Turing không tất định dừng**

**Định nghĩa 2.1.2** Cho một máy Turing không tất định dừng  $N$ . **Độ phức tạp thời gian của máy Turing không tất định  $N$**  là hàm  $t : \mathbb{N} \rightarrow \mathbb{N}$ , mà giá trị  $t(n)$  là số lần tối đa các phép biến đổi cơ bản được sử dụng trong quá trình tính toán của máy  $N$  trên mỗi từ vào độ dài  $n$  và theo từng nhánh trong cây tính toán của máy.

Độ phức tạp thời gian của máy Turing tất định và của máy Turing không tất định được diễn tả trong hình



Hình 2.1 Quy chuẩn về thời gian tất định và không tất định

### 12. Nội dung phân tích thuật toán (mục 2.1.2, không cần ví dụ)

Để giải trọn vẹn một bài toán cho trước, ta không chỉ xây dựng thuật toán cho bài toán mà cần phải tiến hành phân tích thuật toán nhằm xác định độ phức tạp thời gian của thuật toán và qua đó thấy được tính hữu hiệu của nó. Việc phân tích thuật toán giúp ta phát hiện “những trường hợp xấu nhất” đối với thuật toán (những trường hợp mà khi tính toán cần một lượng thời gian lớn nhất) và tiến hành xác định độ phức tạp thời gian của thuật toán.

Do lượng thời gian thuật toán cần đến trong trường hợp xấu nhất thường được diễn đạt bởi một biểu thức phức tạp, nên thông thường ta tìm cách ước lượng nó bởi biểu thức đơn giản. Mặt khác, theo yêu cầu ta cũng chỉ cần biết được lượng thời gian khi tính toán trên những đầu vào cỡ lớn. Để đáp ứng những yêu cầu này, khái niệm **O-lớn** thường được sử dụng và khá tiện lợi trong việc diễn tả độ phức tạp thời gian của thuật toán. Như vậy, khi không cần thiết phải xác định chính xác, độ phức tạp thời gian của thuật toán được thể hiện một cách tương đối bởi cận trên tiệm cận của nó.

Máy Turing có khả năng diễn tả một cách tỉ mỉ các quá trình tính toán và được thừa nhận là một mô hình toán học của thuật toán. Bởi vậy, việc phân tích thuật toán đối với máy Turing sẽ cho ta hiểu được một cách căn kẽ về bản chất của các quá trình tính toán, và qua đó không chỉ giúp ta tìm được **cận trên tiệm cận** hoặc **xác định chính xác** độ phức tạp thời gian của máy Turing mà còn gợi mở cho ta **cải tiến cách tính toán** hoặc **lựa chọn mô hình tính toán khác** hiệu quả hơn.

### **13. Nội dung các định lý về quan hệ thời gian giữa các loại máy (mục 2.1.3, không cần chứng minh)**

**Định lý 2.1.3** *Giả sử  $t : \mathbb{N} \rightarrow \mathbb{N}$  là một hàm mà  $t(n) \geq n$ .*

*Khi đó mỗi máy Turing tất định nhiều•bằng thời gian  $t(n)$  đều có máy Turing tất định một bằng thời gian  $O[t^2(n)]$  tương đương.*

Như vậy, nếu một ngôn ngữ được khẳng định trong thời gian  **$t(n)$**  bởi máy Turing **tất định nhiều bằng** thì được khẳng định trong thời gian  **$O[t^2(n)]$**  bởi máy Turing **tất định một bằng**. Sau đây ta sẽ trình bày **định lý tương tự** cho máy Turing tất định và máy Turing không tất định, và qua đó thấy được một sự **khác biệt** rất lớn theo **tiêu chuẩn độ phức tạp tính toán** giữa hai mô hình tính toán này.

Do những đặc tính riêng, **quá trình tính toán** của máy Turing không tất định  $N$  trên mỗi từ vào được thể hiện dưới dạng một **cây tính toán có hướng**  $TN()$ , sao cho mỗi **nhánh** xuất phát từ gốc cây là một **phương án** tính toán mà máy có thể lựa chọn. Nếu  $N$  là một máy Turing **dừng** thì mọi nhánh tính toán trong  $TN()$  trên mỗi từ vào đều **hữu hạn**. Trong trường hợp này, **độ phức tạp** thời gian của  $N$  chính là **độ dài** của nhánh dài nhất trong cây tính toán  $TN()$  trên mọi từ vào với cùng độ dài  $n$ .

Máy Turing không tất định hoàn toàn **không thích hợp** với bất cứ mô hình tính toán thực tế nào hiện có. Tuy vậy, đây lại là một mô hình tính toán hình thức được sử dụng một cách rất hữu ích trong việc **đặc trưng độ phức tạp** của một lớp khá rộng rãi các bài toán quan trọng.

**Định lý 2.1.4** *Giả sử  $t : \mathbb{N} \rightarrow \mathbb{N}$  là một hàm mà  $t(n) \geq n$ .*

*Khi đó mỗi máy Turing không tất định thời gian  $t(n)$  đều có máy Turing tất định một bằng thời gian  $2^{O[t(n)]}$  tương đương.*

### **14. Thời gian đa thức (cuối trang 138- đầu trang 140)**

Về mặt lý thuyết, sự khác nhau về thời gian tính toán ở mức độ đa thức là nhỏ, còn khác nhau ở mức độ hàm mũ là rất lớn. Một mặt, đó là do sự khác nhau rất lớn giữa tốc độ tăng trưởng của đa thức và của hàm mũ. Mặt khác, tốc độ tăng của đa thức cũng đủ lớn nhưng vẫn trong phạm vi quản lý được, trong khi đó hàm mũ tăng với một tốc độ to lớn đến mức không thể kiểm soát nổi. Một trong những tình huống cần đến những thuật toán thời gian hàm mũ là khi ta giải các bài toán bằng



cách tìm kiếm nghiệm một cách tường tận trong không gian nghiệm bài toán. Khi không gian tìm kiếm với số lượng nghiệm cỡ hàm mũ, thì việc tìm kiếm cần thực hiện trong thời gian hàm mũ. Đôi khi ta có thể tránh được việc duyệt toàn bộ nhờ sự hiểu biết sâu sắc về bài toán, tạo cơ sở xây dựng những thuật toán thời gian đa thức.

Các mô hình máy Turing tất định một băng và nhiều băng là tương đương đa thức với nhau. Ta có thể nói rằng, tất cả các mô hình tính toán tất định thích hợp với các máy tính thực tế đều tương đương đa thức. Nghĩa là, bất cứ một mô hình nào như vậy cũng mô phỏng được mô hình khác chỉ với một lượng thời gian tăng thêm cỡ đa thức.

Từ đây ta tập trung xem xét các khía cạnh của lý thuyết độ phức tạp tính toán mà chúng không bị ảnh hưởng bởi những thay đổi về thời gian ở mức độ đa thức. Những thay đổi như vậy được coi là không đáng kể và được bỏ qua. Cách tiếp cận này cho phép ta phát triển lý thuyết theo hướng không phụ thuộc vào sự tuyến chọn mô hình tính toán riêng biệt. Bỏ qua những khác nhau ở mức độ đa thức không có nghĩa là ta coi những khác nhau này là không quan trọng. Việc bỏ qua giúp ta có một cách nhìn thống nhất đối với vấn đề phức tạp, cho ta những hiểu biết tổng quát về độ phức tạp tính toán và một hướng đi đúng đắn khi cần giải quyết các bài toán khó mà ta gặp phải.

**15. Định nghĩa lớp phức tạp Time, lớp P, vì sao lớp P đóng vai trò trung tâm trong lý thuyết độ phức tạp tính toán và là một lớp rất quan trọng (cuối trang 140- đầu trang 141).**

**Lớp phức tạp Time**

**Định nghĩa 2.2.1** Cho một hàm  $t : \mathbb{N} \rightarrow \mathbb{R}_+$ . Ta định nghĩa lớp phức tạp (complexity class)  $\text{TIME}(t(n))$  là lớp tất cả các ngôn ngữ được khẳng định bởi máy Turing tất định thời gian  $O[t(n)]$ .

Rõ ràng

$$\text{TIME}(n) \subseteq \text{TIME}(n^2) \subseteq \text{TIME}(n^3) \subseteq \dots$$

Bây giờ ta định nghĩa một trong những lớp phức tạp quan trọng nhất của lý thuyết độ phức tạp tính toán.

**Lớp P**

### Định nghĩa 2.2.2

$$P \stackrel{\text{def}}{=} \bigcup_k \text{TIME}(n^k).$$

Nói cách khác,  $P$  là lớp tất cả các ngôn ngữ được khẳng định bởi máy Turing tất định thời gian đa thức.

Lớp  $P$  đóng vai trò trung tâm trong lý thuyết độ phức tạp tính toán và là một lớp rất quan trọng, bởi vì:

1.  $P$  không thay đổi đối với tất cả các mô hình tính toán tương đương đa thức với máy Turing tất định một băng, và
2.  $P$  gần như tương ứng với lớp các bài toán giải được một cách thực tế, tức giải được trên máy tính điện tử.

**16. Định nghĩa lớp phức tạp  $N\text{Time}$ , lớp  $NP$ . Trình bày một bài toán cụ thể thuộc lớp  $NP$  là bài toán đường đi Hamilton trong đồ thị có hướng (cuối trang 154- đầu trang 156).**

#### \* $N\text{Time}$

**Định nghĩa 2.3.1** Cho một hàm  $t : \mathbb{N} \rightarrow \mathbb{R}_+$ . Ta định nghĩa lớp phức tạp  $N\text{TIME}(t(n))$  là lớp tất cả các ngôn ngữ được khẳng định bởi máy Turing không tất định thời gian  $O[t(n)]$ .

#### \* $NP$

### Định nghĩa 2.3.2

$$NP \stackrel{\text{def}}{=} \bigcup_k N\text{TIME}(n^k).$$

Nói cách khác,  $NP$  là lớp tất cả các ngôn ngữ được khẳng định bởi máy Turing không tất định thời gian đa thức.

#### \*Bài toán cụ thể

#### **$DIHAMPATH$**

Dữ kiện: Cho một đồ thị có hướng  $G$  và hai đỉnh  $u, v$  thuộc  $G$ .

Câu hỏi: Phải chăng trong  $G$  tồn tại đường đi Hamilton từ  $u$  đến  $v$ ?

Ngôn ngữ tương ứng với bài toán này được xác định bởi

**$DIHAMPATH = \{ \langle G, u, v \rangle \mid G \text{ là một đồ thị có hướng chứa đường đi Hamilton từ } u \text{ đến } v \}$ .**

Ta chứng minh được:  **$DIHAMPATH \in NP$ .**

Giả sử  $G$  là một đồ thị có hướng với  $m$  đỉnh được gán “tên” bởi các số tự nhiên từ 1 đến  $m$ . Khi đó một dãy số  $(k_1, k_2, \dots, k_m)$ , bao gồm các số tự nhiên từ 1 đến  $m$ , tạo thành một đường đi Hamilton trong  $G$  có hướng từ đỉnh  $u$  đến đỉnh  $v$ , nếu thỏa mãn:

- $k_i$  khác nhau



- $u = k_1$  và  $v = k_m$
- Với mỗi  $1 \leq i \leq m-1$ ,  $G$  có cung  $(k_i, k_{i+1})$

Ta thu được máy Turing không tắt định  $N$  cho bài toán **DIHAMPATH**:

$N =$  “Trên từ vào  $\langle G, u, v \rangle$ ,  $G$  là đồ thị có hướng với  $m$  đỉnh chứa hai đỉnh  $u$  và  $v$ .

1. Đề xuất một cách không tắt định một dãy gồm  $m$  số tự nhiên khác nhau  $(k_1, k_2, \dots, k_m)$ , được chọn không tắt định trong tập  $\{1, 2, \dots, m\}$  sao cho  $k_1 = u$  và  $k_m = v$ .
2. Đối với mỗi  $i$  từ 1 đến  $m-1$ , lần lượt kiểm tra xem liệu  $G$  có chứa cung  $(k_i, k_{i+1})$  hay không. Nếu  $G$  không chứa dù bất cứ cung nào, bác bỏ. Ngược lại, chấp nhận.”

**17. Định nghĩa về: độ phức tạp không gian của máy Turing tắt định 1 băng dừng, máy Turing tắt định nhiều băng dừng, máy Turing không tắt định dừng. Định lý về mối quan hệ giữa thời gian và không gian (định lý 3.2.2).**

**Độ phức tạp không gian của máy Turing tắt định 1 băng dừng**

**Định nghĩa 3.1.1** Cho  $M$  là một máy Turing tắt định một băng dừng. Độ phức tạp không gian của máy Turing (space complexity of Turing machine)  $M$  là hàm  $s : \mathbb{N} \rightarrow \mathbb{N}$ , trong đó giá trị  $s(n)$  là số tối đa các ô trên băng mà máy  $M$  sử dụng tại mỗi thời điểm trong quá trình tính toán của máy trên bất cứ từ vào nào với cùng độ dài  $n$ .

Khi máy Turing  $M$  có độ phức tạp không gian  $s(n)$ , ta nói rằng  $M$  hoạt động trong không gian  $s(n)$  hay  $M$  là máy Turing không gian  $s(n)$ .

**Độ phức tạp không gian của máy Turing tắt định nhiều băng dừng**

**Định nghĩa 3.1.2** Cho  $M$  là một máy Turing nhiều băng dừng. Độ phức tạp không gian của máy Turing nhiều băng (space complexity of multitape Turing machine)  $M$  là hàm  $s : \mathbb{N} \rightarrow \mathbb{N}$ , trong đó giá trị  $s(n)$  là số tối đa các ô trên mỗi băng mà máy  $M$  sử dụng tại mỗi thời điểm trong quá trình tính toán của máy trên bất cứ từ vào nào độ dài  $n$ .

**Độ phức tạp không gian của máy Turing không tắt định**

**Định nghĩa 3.1.7** Cho  $N$  là máy Turing không tắt định dừng. Độ phức tạp không gian của máy Turing không tắt định  $N$  là hàm  $s: \mathbb{N} \rightarrow \mathbb{N}$ , trong đó  $s(n)$  là số tối đa các ô trên băng được sử dụng tại mỗi thời điểm trong quá trình tính toán theo mọi nhánh thuộc cây tính toán của máy  $N$  trên bất cứ từ vào nào độ dài  $n$ .

**Định lý 3.2.2** Giả sử  $M$  là một máy Turing dừng. Khi đó:

- (i) Nếu  $M$  có độ phức tạp thời gian  $f(n)$ ,  $f(n) \geq n$ , thì độ phức tạp không gian của  $M$  không vượt quá  $f(n) + 1$ ;
- (ii) Nếu  $M$  có độ phức tạp không gian  $f(n)$ ,  $f(n) \geq \log n$ , thì độ phức tạp thời gian của  $M$  không vượt quá  $2^{O[f(n)]}$ .

*Chứng minh* Ta chứng minh định lý cho trường hợp khi  $M$  là máy Turing tắt định một băng. Trong những trường hợp khác, việc chứng minh hoàn toàn tương tự.

### **Định lý về mối quan hệ giữa thời gian và không gian**

Giả sử  $M$  là một máy Turing dừng. Khi đó:

- Nếu  $M$  có độ phức tạp thời gian  $f(n)$ ,  $f(n) \geq n$ , thì độ phức tạp không gian của  $M$  không vượt quá  $f(n)+1$ ;
- Nếu  $M$  có độ phức tạp không gian  $f(n)$ ,  $f(n) > \log n$ , thì độ phức tạp thời gian của  $M$  không vượt quá  $2^{O[f(n)]}$ .

## **Bài tập**

**45**

**Máy**

**Turing 1 băng đoán nhận ngôn ngữ  $0^i1^j$**

**TH:  $i = j$ ;  $j > 0$**

Máy turing 1 băng đoán nhận ngôn ngữ  $0^i1^j$  với  $(i = j, j > 0)$  được xây như sau:

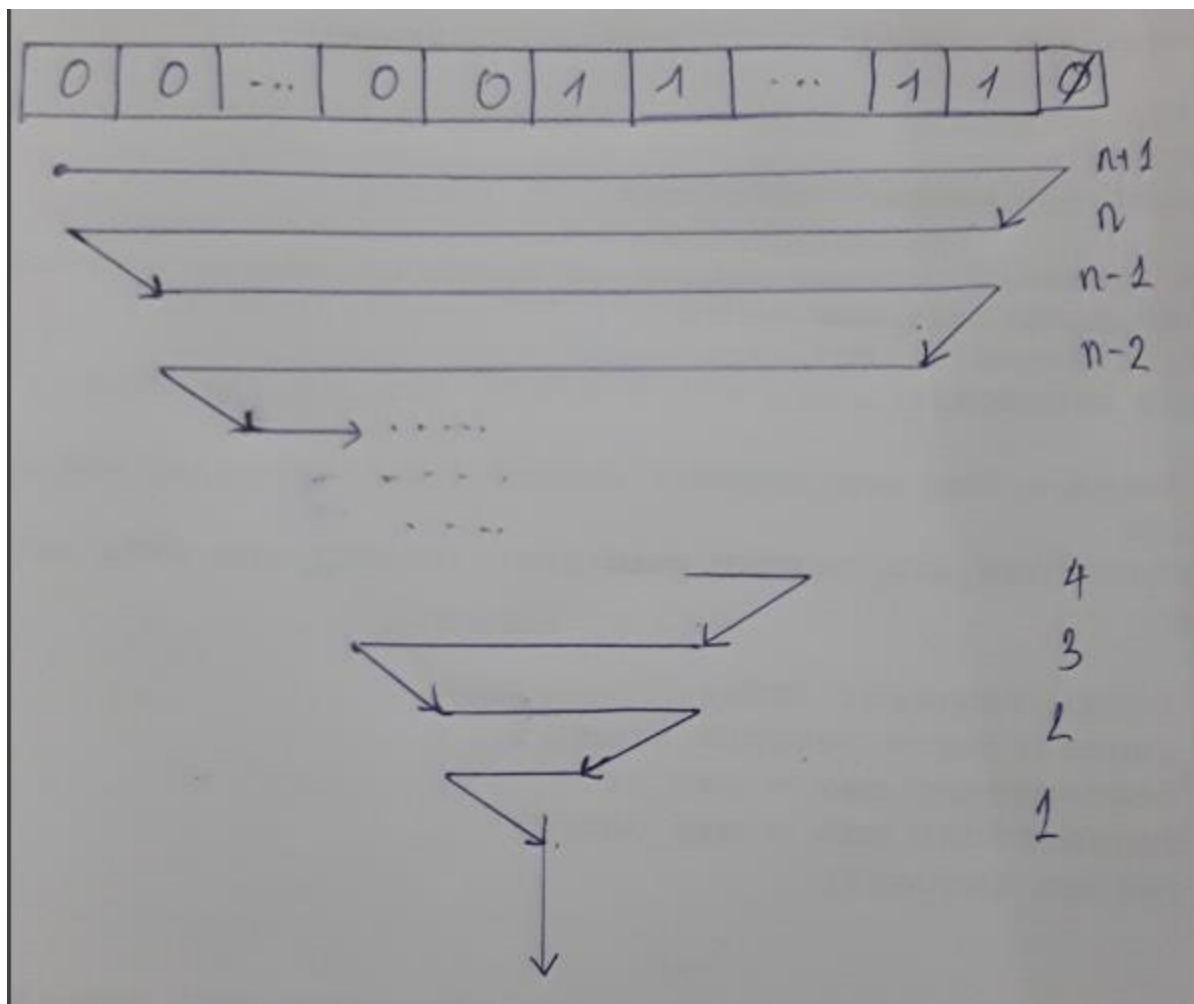
$M = \{Q, \{0,1\}, \{0,1, \emptyset\}, \square, q_n, q_0, q_y\}$

Trong đó  $Q = \{q_0, q_0', q_n, q_y, q_1, q_2, q_3, q_4\}$

1.  $\square(q_0, \emptyset) = (q_N, \emptyset, S)$
2.  $\square(q_0, 1) = (q_N, 1, S)$
3.  $\square(q_0, 0) = (q_1, \emptyset, R)$
4.  $\square(q_1, \emptyset) = (q_N, \emptyset, S)$
5.  $\square(q_1, 0) = (q_1, 0, R)$
6.  $\square(q_1, 1) = (q_2, 1, R)$
7.  $\square(q_2, 1) = (q_2, 1, R)$
8.  $\square(q_2, 0) = (q_N, 0, S)$
9.  $\square(q_2, \emptyset) = (q_3, \emptyset, L)$
10.  $\square(q_3, 1) = (q_4, \emptyset, L)$
11.  $\square(q_4, 1) = (q_4, 1, L)$
12.  $\square(q_4, 0) = (q_4, 0, L)$
13.  $\square(q_4, \emptyset) = (q_0', \emptyset, R)$
14.  $\square(q_0', 0) = (q_1, \emptyset, R)$
15.  $\square(q_0', \emptyset) = (q_y, \emptyset, L)$
16.  $\square(q_0', 1) = (q_N, 1, S)$

Độ phức tạp thời gian:

Trong quá trình tính toán của máy turing A trên mỗi từ vào việc loại bỏ những từ không có dạng  $0i1j$  được thực hiện đồng thời cùng với bước lặp đầu tiên của quá trình lặp. Do đó độ phức tạp thời gian  $T_A(n)$  của máy turing A được xác định bởi lượng thời gian cần thiết để thực hiện toàn bộ quá trình lặp trên từ vào  $0i1j$  độ dài  $n$ . Một trường hợp xấu nhất đối với A mà ta đã phát hiện. Trong trường hợp này mỗi bước lặp được thực hiện bằng cách di chuyển đầu đọc ghi trên băng từ ký tự 0 bên trái nhất đến ký tự 1 bên phải nhất và sau khi xóa những ký tự này lại di chuyển về ký tự 0 bên trái nhất của phần tử còn lại. Hành trình của đầu đọc ghi trên băng được diễn tả trên hình dưới đây cùng với lượng thời gian sử dụng từng đoạn:



Ta có độ phức tạp thời gian là :

$$T_A(n) = (n+1)(n+2) / 2$$

Cuối cùng ta có kết luận rằng ngôn ngữ L được khẳng định trong thời gian  $O[n^2]$  bởi máy Turing tất định một băng A

Độ phức tạp không gian:

Trong quá trình tính toán, số ô mà máy sử dụng không tăng thêm và cứ sau mỗi bước lặp lại có 2 ô được giải phóng. Số ô mà máy sử dụng là  $n+1$  tại thời điểm  $t=n$  ở lần lặp đầu tiên, đầu đọc lúc này ở tại vị trí ô trống (phần này thầy bảo thế thì phải, còn trong sách là  $n$  thôi). Vì vậy, độ phức tạp không gian chính xác của máy là  $s_A(n) = n+1$ . Ta có thể nói độ phức tạp không gian của máy là  $O[n]$

**TH:  $i = j+1; j > 0$**

Máy turing đoán nhận ngôn ngữ  $0i1j$  với  $(i = j+1, j > 0)$  được xây dựng bởi:

$$M = \{Q, \{0, 1\}, \{0, 1, \emptyset\}, \square, q_0, q_n, q_y, \}$$

Trong đó  $Q = \{q_0, q_0', q_y, q_n, q_1, q_2, q_3, q_4, q_5, q_6\}$

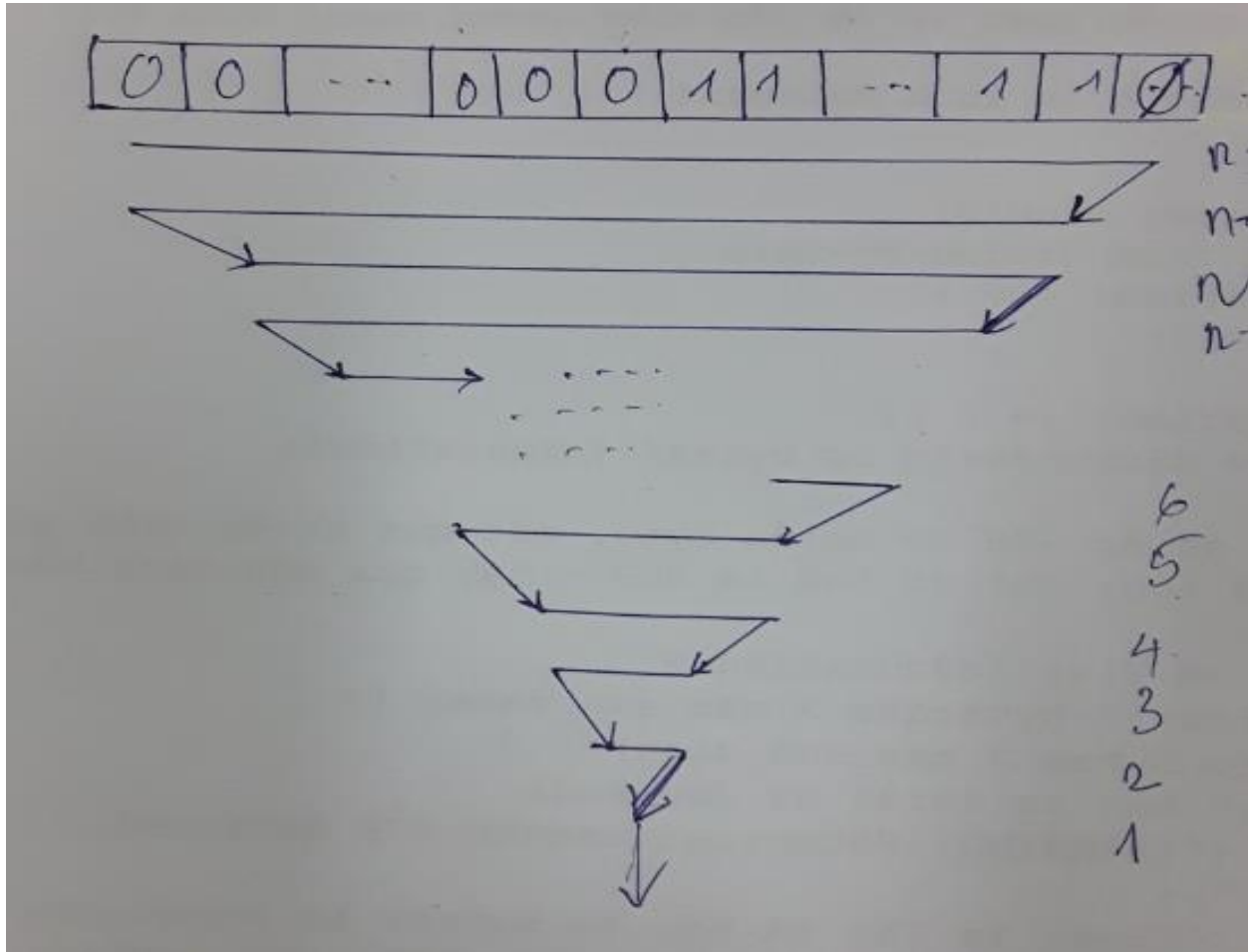
Hàm chuyển được xây dựng như sau:

1.  $\square(q_0, \emptyset) = (q_n, \emptyset, S)$
2.  $\square(q_0, 1) = (q_n, 1, S)$
3.  $\square(q_0, 0) = (q_1, \emptyset, R)$
4.  $\square(q_1, \emptyset) = (q_n, \emptyset, S)$
5.  $\square(q_1, 0) = (q_1, 0, R)$
6.  $\square(q_1, 1) = (q_2, 1, R)$
7.  $\square(q_2, 1) = (q_2, 1, R)$
8.  $\square(q_2, 0) = (q_n, 0, S)$
9.  $\square(q_2, \emptyset) = (q_3, \emptyset, L)$
10.  $\square(q_3, 1) = (q_4, \emptyset, L)$
11.  $\square(q_4, 1) = (q_4, 1, L)$
12.  $\square(q_4, 0) = (q_4, 0, L)$
13.  $\square(q_4, \emptyset) = (q_0', \emptyset, R)$
14.  $\square(q_0', 0) = (q_5, \emptyset, R)$
15.  $\square(q_0', \emptyset) = (q_n, \emptyset, L)$
16.  $\square(q_0', 1) = (q_n, 1, S)$
17.  $\square(q_5, 0) = (q_5, 0, R)$
18.  $\square(q_5, 1) = (q_2, 1, R)$
19.  $\square(q_5, \emptyset) = (q_6, \emptyset, L)$
20.  $\square(q_6, 0) = (q_n, 0, S)$
21.  $\square(q_6, \emptyset) = (q_y, \emptyset, S)$

Độ phức tạp thời gian:

Ta thấy rằng trong quá trình tính toán của máy turing A trên mỗi từ vào việc loại bỏ như từ không có dạng  $0i1j$  được thực hiện đồng thời với những bước lặp đầu tiên của quá trình lặp. Do đó độ phức tạp thời gian  $T_A(n)$  của máy turing A được xác định bởi lượng thời gian cần thiết để thực hiện toàn bộ quy trình lặp trên từ vào  $0i1j$  độ dài n. Một trường hợp xấu nhất đối với A mà ta đã phát hiện. Trong trường

hợp này mỗi bước lặp được thực hiện bằng cách di chuyển đầu đọc ghi trên băng từ ký tự 0 bên trái nhất đến ký tự 1 bên phải nhất và sau khi xóa những ký tự này lại di chuyển về ký tự 0 bên trái nhất của phần còn lại. Hành trình của đầu đọc ghi trên băng được diễn tả dưới hình dưới đây với lượng thời gian sử dụng ở từng đoạn:



Độ phức tạp không gian:

Trong quá trình tính toán số ô mà máy sử dụng không tăng thêm và cứ sau mỗi bước lặp có 2 ô được giải phóng số ô mà máy sử dụng là  $n+1$  tại thời điểm  $t = n$  ở lần lặp đầu tiên. Đầu đọc lúc này ở tại vị trí ô trống vì vậy độ phức tạp không gian chính xác của máy  $S_A(n) = n+1$ . Ta có thể nói độ phức tạp không gian là  $O[n]$

**TH:  $i = j + 2; j > 0$**

Máy turing đoán nhận ngôn ngữ  $0^i1^j$  với  $(i = j+2, j > 0)$  được xây dựng bởi:

$$M = \{Q, \{0, 1\}, \{0,1,\emptyset\}, \square, q_0, q_n, q_y, \}$$

$$\text{Trong đó } Q = \{q_0, q_0', q_y, q_n, q_1, q_2, q_3, q_4, q_5, q_6, q_7\}$$

Hàm chuyển được xây dựng như sau:

$$1. \square(q_0, \emptyset) = (q_n, \emptyset, S)$$

$$2. \square(q_0, 1) = (q_n, 1, S)$$

$$3. \square(q_0, 0) = (q_1, \emptyset, S)$$

$$4. \square(q_1, \emptyset) = (q_n, \emptyset, S)$$

$$5. \square(q_1, 0) = (q_1, 0, R)$$

$$6. \square(q_1, 1) = (q_2, 1, R)$$

$$7. \square(q_2, 1) = (q_2, 1, R)$$

$$8. \square(q_2, 0) = (q_n, 0, S)$$

$$9. \square(q_2, \emptyset) = (q_3, \emptyset, L)$$

$$10. \square(q_3, 1) = (q_4, \emptyset, L)$$

$$11. \square(q_4, 1) = (q_4, 1, L)$$

$$12. \square(q_4, 0) = (q_4, 0, L)$$

$$13. \square(q_4, \emptyset) = (q_0', \emptyset, R)$$

$$14. \square(q_0', 0) = (q_5, \emptyset, R)$$

$$15. \square(q_0', 1) = (q_n, 1, S)$$

$$16. \square(q_0', \emptyset) = (q_n, \emptyset, S)$$

$$17 \quad \square(q_5, 0) = (q_5, 0, R)$$

$$18 \quad \square(q_5, 1) = (q_2, 1, R)$$

$$19 \quad \square(q_5, \emptyset) = (q_6, \emptyset, L)$$

$$20 \quad \square(q_6, 0) = (q_7, \emptyset, L)$$

$$21 \quad \square(q_7, 0) = (q_n, 0, S)$$

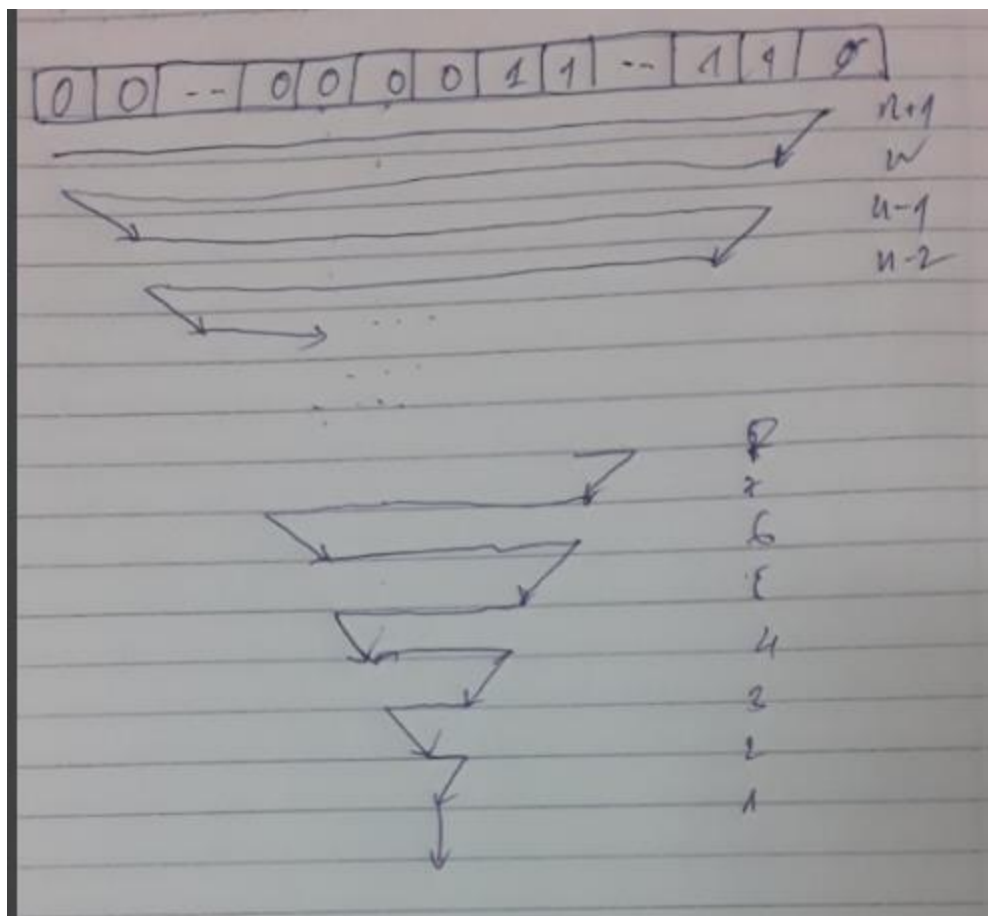
$$22 \quad \square(q_7, \emptyset) = (q_y, \emptyset, S)$$

Độ phức tạp thời gian:

Ta nhận thấy rằng trong quá trình tính toán máy turing A trên mỗi từ vào, việc loại bỏ các trường hợp không có dạng 0i1j được thực hiện đồng thời với bước lặp đầu tiên. Do đó độ phức tạp thời gian  $T_A(n)$  của máy turing A được xác định bởi lượng thời gian cần thiết để hoàn thiện toàn bộ quy trình lặp trên từ vào 0i1j độ dài n.

Một trường hợp xấu nhất đối với A ta đã phát hiện. Trong trường hợp này mỗi bước lặp được thực hiện bằng cách di chuyển đầu đọc ghi trên băng từ ký tự 0 bên trái nhất đến ký tự bên phải nhất và sau khi xóa những ký tự này lại di chuyển về ký tự 0 bên trái nhất phần còn lại. Hành trình của đầu đọc ghi trên băng được diễn tả trên hình dưới đây cùng với lượng thời gian tương ứng với từng đoạn.





Độ phức tạp không gian:

Trong quá trình tính toán ô mà máy tính sử dụng không thay đổi và mỗi lần lặp thì có 2 ô được giải phóng, số ô mà máy tính sử dụng là  $n+1$  tại thời điểm  $t = n$  ở lần lặp đầu tiên. Đầu đọc lúc này ở vị trí ô trống nên độ phức tạp không gian của máy là  $S_A = n+1$  ta có thể nói độ phức tạp không gian là  $O[n]$

**TH:  $j = i+1; i > 0$**

Máy turing đoán nhận ngôn ngữ  $0^i1^j$  với ( $j = i+1, i > 0$ ) được xây dựng bởi:

$M = \{Q, \{0, 1\}, \{0, 1, \emptyset\}, \square, q_0, q_n, q_y, \}$

Trong đó  $Q = \{q_0, q_0', q_y, q_n, q_1, q_2, q_3, q_4, q_5\}$

Hàm chuyển được xây dựng như sau:

$$1 \quad \square(q_0, \emptyset) = (q_N, \emptyset, S)$$

$$2 \quad \square(q_0, 1) = (q_N, 1, S)$$

$$3 \quad \square(q_0, 0) = (q_1, \emptyset, S)$$

$$4 \quad \square(q_1, \emptyset) = (q_N, \emptyset, S)$$

$$5 \quad \square(q_1, 0) = (q_1, 0, R)$$

$$6 \quad \square(q_1, 1) = (q_2, 1, R)$$

$$7 \quad \square(q_2, 1) = (q_2, 1, R)$$

$$8 \quad \square(q_2, 0) = (q_N, 0, S)$$

$$9 \quad \square(q_2, \emptyset) = (q_3, \emptyset, L)$$

$$10 \quad \square(q_3, 1) = (q_4, \emptyset, L)$$

$$11 \quad \square(q_4, 1) = (q_4, 1, L)$$

$$12 \quad \square(q_4, 0) = (q_4, 0, L)$$

$$13 \quad \square(q_4, \emptyset) = (q_0', \emptyset, R)$$

$$14 \quad \square(q_0', 0) = (q_1, \emptyset, R)$$

$$15 \quad \square(q_0', 1) = (q_5, \emptyset, R)$$

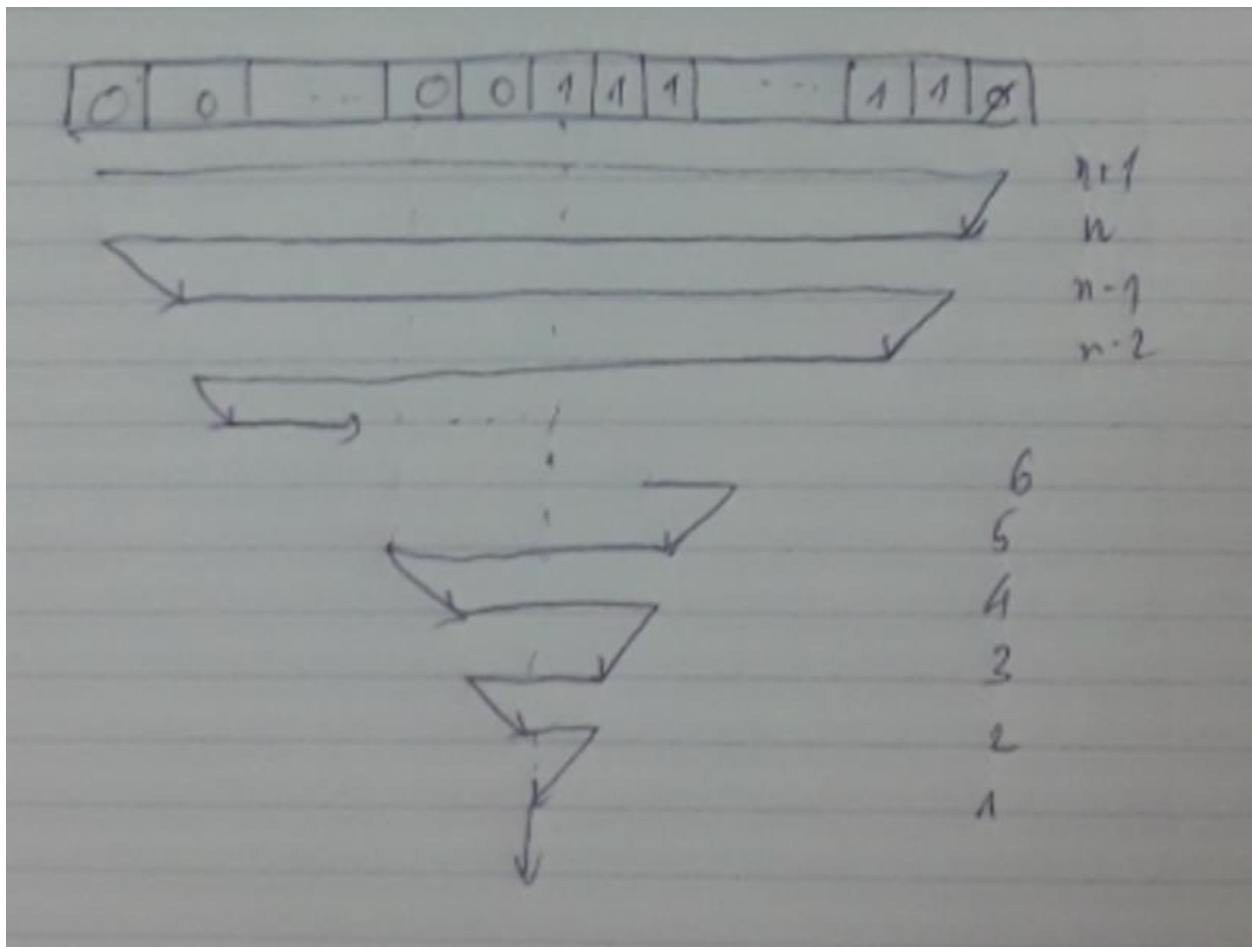
$$16 \quad \square(q_0', \emptyset) = (q_N, \emptyset, S)$$

$$17 \quad \square(q_5, 1) = (q_N, 1, S)$$

$$18 \quad \square(q_5, \emptyset) = (q_y, \emptyset, S)$$

Độ phức tạp thời gian:

Ta nhận thấy rằng trong quá trình tính toán của máy turing A trên mỗi từ vào việc xóa bỏ những trường hợp xấu vào không có dạng 011j được thực hiện đồng thời với những bước lặp đầu tiên. Do đó ta có thể tính độ phức tạp thời gian  $T_A(n)$  của máy turing A được xác định bằng tổng thời gian cần thiết thực hiện toàn bộ quá trình lặp trên từ vào 011j độ dài n. Trường hợp xấu nhất đối với A mà ta đã phát hiện. Trường hợp này mỗi bước lặp được thực hiện bằng cách di chuyển đầu đọc ghi trên băng từ ký tự 0 bên phải nhất đến ký tự 1 bên trái nhất sau khi xóa những ký tự này lại di chuyển về ký tự 0 bên trái nhất của phần còn lại. Hành trình của đầu đọc ghi trên băng được diễn tả trên hình dưới đây với lượng thời gian sử dụng ở từng đoạn:



Độ phức tạp không gian:

Trong quá trình tính toán ô mà máy tính sử dụng không thay đổi và mỗi lần lặp thì có 2 ô được giải phóng, số ô mà máy tính sử dụng là  $n+1$  tại thời điểm  $t = n$  ở lần lặp đầu tiên. Đầu đọc lúc này ở vị trí ô trống nên độ phức tạp không gian của máy là  $S_A = n+1$  ta có thể nói độ phức tạp không gian là  $O[n]$

**TH:  $j = i+2; i > 0$**

Máy turing đoán nhận ngôn ngữ  $0^i1^j$  với  $(j = i+2, i > 0)$  được xây dựng bởi:

$M = \{Q, \{0, 1\}, \{0, 1, \emptyset\}, \square, q_0, q_n, q_y, \}$

Trong đó  $Q = \{q_0, q_0', q_y, q_n, q_1, q_2, q_3, q_4, q_5, q_6\}$

Hàm chuyển được xây dựng như sau:

1  $\square(q_0, \emptyset) = (q_n, \emptyset, S)$

2  $\square(q_0, 1) = (q_n, 1, S)$

3  $\square(q_0, 0) = (q_1, \emptyset, S)$

4  $\square(q_1, \emptyset) = (q_n, \emptyset, S)$

5  $\square(q_1, 0) = (q_1, 0, R)$

6  $\square(q_1, 1) = (q_2, 1, R)$

7  $\square(q_2, 1) = (q_2, 1, R)$

8  $\square(q_2, 0) = (q_n, 0, S)$

9  $\square(q_2, \emptyset) = (q_3, \emptyset, L)$

10  $\square(q_3, 1) = (q_4, \emptyset, L)$

11  $\square(q_4, 1) = (q_4, 1, L)$

12  $\square(q_4, 0) = (q_4, 0, L)$

$$13 \square (q_4, \emptyset) = (q_0', \emptyset, R)$$

$$14 \square (q_0', 0) = (q_1, \emptyset, R)$$

$$15 \square (q_0', \emptyset) = (q_N, \emptyset, S)$$

$$16 \square (q_0', 1) = (q_5, \emptyset, R)$$

$$17 \square (q_5, 1) = (q_6, \emptyset, R)$$

$$18 \square (q_6, \emptyset) = (q_y, \emptyset, S)$$

$$19 \square (q_6, 1) = (q_N, 1, S)$$

Độ phức tạp thời gian:

Ta nhận thấy rằng trong việc tính toán của máy turing A trên mỗi từ vào việc loại bỏ các trường hợp không có dạng  $0i1j$  được thực hiện đồng thời trong những bước lặp đầu tiên. Do đó độ phức tạp thời gian  $T_A(n)$  được xác định bằng lượng thời gian cần thiết thực hiện toàn bộ quy trình lặp trên băng trên từ vào  $0i1j$  độ dài  $n$ . Trường hợp xấu nhất ta đã phát hiện. Trong trường hợp này mỗi bước lặp được thực hiện bằng cách di chuyển đầu đọc ghi từ ký tự 0 bên phải nhất sang ký tự 1 bên phải nhất sau đó xóa các ký tự này ta lại di chuyển về ký tự 0 bên trái nhất của phần còn lại. Hành trình di chuyển của đầu đọc ghi trên băng được diễn tả trên hình dưới đây với lượng thời gian sử dụng ở đoạn:

**<T không biết vẽ>**

Độ phức tạp không gian:

Trong quá trình tính toán ô mà máy tính sử dụng không thay đổi và mỗi lần lặp thì có 2 ô được giải phóng, số ô mà máy tính sử dụng là  $n+1$  tại thời điểm  $t = n$  ở lần lặp đầu tiên. Đầu đọc lúc này ở vị trí ô trống nên độ phức tạp không gian của máy là  $S_A = n+1$  ta có thể nói độ phức tạp không gian là  $O[n]$

**TH:  $i > j; j > 0$**

Máy turing đoán nhận ngôn ngữ  $0i1j$  với  $(i > j, j > 0)$  được xây dựng bởi:

$$M = \{Q, \{0, 1\}, \{0,1,\emptyset\}, \square, q_0, q_n, q_y, \}$$

$$\text{Trong đó } Q = \{q_0, q_0', q_y, q_n, q_1, q_2, q_3, q_4, q_5, \}$$

Hàm chuyển được xây dựng như sau:

$$\square(q_0, \emptyset) = (q_n, \emptyset, S)$$

$$\square(q_0, 1) = (q_n, 1, S)$$

$$\square(q_0, 0) = (q_1, \emptyset, S)$$

$$\square(q_1, \emptyset) = (q_n, \emptyset, S)$$

$$\square(q_1, 0) = (q_1, 0, R)$$

$$\square(q_1, 1) = (q_2, 1, R)$$

$$\square(q_2, 1) = (q_2, 1, R)$$

$$\square(q_2, 0) = (q_n, 0, S)$$

$$\square(q_2, \emptyset) = (q_3, \emptyset, L)$$

$$\square(q_3, 1) = (q_4, \emptyset, L)$$

$$\square(q_4, 1) = (q_4, 1, L)$$

$$\square(q_4, 0) = (q_4, 0, L)$$

$$\square(q_4, \emptyset) = (q_0', \emptyset, R)$$

$$\square(q_0', 0) = (q_5, \emptyset, R)$$

$$\square(q_0', \emptyset) = (q_n, \emptyset, S)$$

$$\square(q_0', 1) = (q_n, 1, S)$$

$$\square(q_5, 0) = (q_5, 0, R)$$

$$\square(q_5, 1) = (q_2, 1, R)$$

$$\square(q_5, \emptyset) = (q_y, \emptyset, L)$$

Độ phức tạp thời gian:

Ta nhận thấy rằng trong việc tính toán của máy turing A trên mỗi từ vào việc loại bỏ các trường hợp không có dạng 0i1j được thực hiện đồng thời trong những bước lặp đầu tiên. Do đó độ phức tạp thời gian  $T_A(n)$  được xác định bằng lượng thời gian cần thiết thực hiện toàn bộ quy trình lặp trên băng trên từ vào 0i1j độ dài n. Trường hợp xấu nhất ta đã phát hiện. Trong trường hợp này mỗi bước lặp được thực hiện bằng cách di chuyển đầu đọc ghi từ ký tự 0 bên phải nhất sang ký tự 1 bên phải nhất sau đó xóa các ký tự này ta lại di chuyển về ký tự 0 bên trái nhất của phần còn lại. Hành trình di chuyển của đầu đọc ghi trên băng được diễn tả trên hình dưới đây với lượng thời gian sử dụng ở đoạn:

**<T không biết vẽ>**

Độ phức tạp không gian:

Trong quá trình tính toán ô mà máy tính sử dụng không thay đổi và mỗi lần lặp thì có 2 ô được giải phóng, số ô mà máy tính sử dụng là  $n+1$  tại thời điểm  $t = n$  ở lần lặp đầu tiên. Đầu đọc lúc này ở vị trí ô trống nên độ phức tạp không gian của máy là  $S_A = n+1$  ta có thể nói độ phức tạp không gian là  $O[n]$

**TH:  $i > j+1; j > 0$**

Máy turing đoán nhận ngôn ngữ 0i1j với  $(i > j+1, j > 0)$  được xây dựng bởi:

$$M = \{Q, \{0, 1\}, \{0, 1, \emptyset\}, \square, q_0, q_n, q_y\}$$

$$\text{Trong đó } Q = \{q_0, q_0', q_y, q_n, q_1, q_2, q_3, q_4, q_5, q_6\}$$

Hàm chuyển được xây dựng như sau:

$$1 \sqcap (q_0, \emptyset) = (q_N, \emptyset, S)$$

$$2 \sqcap (q_0, 1) = (q_N, 1, S)$$

$$3 \sqcap (q_0, 0) = (q_1, \emptyset, S)$$

$$4 \sqcap (q_1, \emptyset) = (q_N, \emptyset, S)$$

$$5 \sqcap (q_1, 0) = (q_1, 0, R)$$

$$6 \sqcap (q_1, 1) = (q_2, 1, R)$$

$$7 \sqcap (q_2, 1) = (q_2, 1, R)$$

$$8 \sqcap (q_2, 0) = (q_N, 0, S)$$

$$9 \sqcap (q_2, \emptyset) = (q_3, \emptyset, L)$$

$$10 \sqcap (q_3, 1) = (q_4, \emptyset, L)$$

$$11 \sqcap (q_4, 1) = (q_4, 1, L)$$

$$12 \sqcap (q_4, 0) = (q_4, 0, L)$$

$$13 \sqcap (q_4, \emptyset) = (q_0', \emptyset, R)$$

$$14 \sqcap (q_0', 0) = (q_5, \emptyset, R)$$

$$15 \sqcap (q_0', \emptyset) = (q_N, \emptyset, S)$$

$$16 \sqcap (q_0', 1) = (q_N, 1, S)$$

$$17 \sqcap (q_5, 0) = (q_5, 0, R)$$

$$18 \sqcap (q_5, 1) = (q_2, 1, R)$$

$$19 \sqcap (q_5, \emptyset) = (q_6, \emptyset, L)$$



$$20 \quad \square(q_6, 0) = (q_Y, \emptyset, L)$$

$$21 \quad \square(q_6, \emptyset) = (q_N, \emptyset, S)$$

Độ phức tạp thời gian:

Ta nhận thấy rằng trong việc tính toán của máy turing A trên mỗi từ vào việc loại bỏ các trường hợp không có dạng  $0i1j$  được thực hiện đồng thời trong những bước lặp đầu tiên. Do đó độ phức tạp thời gian  $T_A(n)$  được xác định bằng lượng thời gian cần thiết thực hiện toàn bộ quy trình lặp trên băng trên từ vào  $0i1j$  độ dài  $n$ . Trường hợp xấu nhất ta đã phát hiện. Trong trường hợp này mỗi bước lặp được thực hiện bằng cách di chuyển đầu đọc ghi từ ký tự 0 bên phải nhất sang ký tự 1 bên phải nhất sau đó xóa các ký tự này ta lại di chuyển về ký tự 0 bên trái nhất của phần còn lại. Hành trình di chuyển của đầu đọc ghi trên băng được diễn tả trên hình dưới đây với lượng thời gian sử dụng ở đoạn:

**<T không biết vẽ>**

Độ phức tạp không gian:

Trong quá trình tính toán ô mà máy tính sử dụng không thay đổi và mỗi lần lặp thì có 2 ô được giải phóng, số ô mà máy tính sử dụng là  $n+1$  tại thời điểm  $t = n$  ở lần lặp đầu tiên. Đầu đọc lúc này ở vị trí ô trống nên độ phức tạp không gian của máy là  $S_A = n+1$  ta có thể nói độ phức tạp không gian là  $O[n]$

**<T không biết vẽ>**

Độ phức tạp không gian:

Trong quá trình tính toán ô mà máy tính sử dụng không thay đổi và mỗi lần lặp thì có 2 ô được giải phóng, số ô mà máy tính sử dụng là  $n+1$  tại thời điểm  $t = n$  ở lần lặp đầu tiên. Đầu đọc lúc này ở vị trí ô trống nên độ phức tạp không gian của máy là  $S_A = n+1$  ta có thể nói độ phức tạp không gian là  $O[n]$

**Máy Turing 2 băng đoán nhận**

**TH:  $w\#w, w \in \{0, 1\}^+$**

Máy turing đoán nhận xâu vào  $w\#w$  trên bộ chữ  $\{0, 1\}$  được xây dựng như sau:

$$M = \{Q, \{0,1,\#\}, \{0,1,\#, \emptyset, \}, \delta, q_o, q_y, q_n\}$$

$$\text{Trong đó } Q = \{q_o, q_y, q_n, q_1, q_2, q_3\}$$

Hàm chuyển được viết như sau:

$$\square(q_o, (\emptyset, \emptyset)) = (q_n, (\emptyset, \emptyset), (S, S))$$

$$\square(q_o, (1, \emptyset)) = (q_1, (1, \#), (S, R))$$

$$\square(q_o, (0, \emptyset)) = (q_1, (0, \#\emptyset), (S, R))$$

$$\square(q_o, (\#, \emptyset)) = (q_n, (\#, \emptyset), (S, S))$$

$$\square(q_1, (1, \emptyset)) = (q_1, (1, 1), (R, R))$$

$$\square(q_1, (0, \emptyset)) = (q_1, (0, 0), (R, R))$$

$$\square(q_1, (\#, \emptyset)) = (q_2, (\#, \emptyset), (S, L))$$

$$\square(q_1, (\emptyset, \emptyset)) = (q_n, (\emptyset, \emptyset), (S, S))$$

$$\square(q_2, (\#, 1)) = (q_2, (\#, 1), (S, L))$$

$$\square(q_2, (\#, 0)) = (q_2, (\#, 0), (S, L))$$

$$\square(q_2, (\#, \#)) = (q_3, (\#, \#), (R, R))$$

$$\square(q_3, (1, 1)) = (q_3, (1, 1), (R, R))$$

$$\square(q_3, (1, \emptyset)) = (q_n, (1, \emptyset), (S, S))$$

$$\square(q_3, (1, 0)) = (q_n, (1, 0), (S, S))$$

$$\square(q_3, (0, 0)) = (q_3, (0, 0), (R, R))$$

$$\square(q_3, (0, 1)) = (q_N, (0, 1), (S, S))$$

$$\square(q_3, (0, \emptyset)) = (q_N, (0, \emptyset), (S, S))$$

$$\square(q_3, (\emptyset, 1)) = (q_N, (\emptyset, 1), (S, S))$$

$$\square(q_3, (\emptyset, 0)) = (q_N, (\emptyset, 0), (S, S))$$

$$\square(q_3, (\emptyset, \emptyset)) = (q_y, (\emptyset, \emptyset), (S, S))$$

$$\square(q_3, (\#, 1)) = (q_N, (\emptyset, \emptyset), (S, S))$$

$$\square(q_3, (\#, 0)) = (q_N, (\emptyset, \emptyset), (S, S))$$

$$\square(q_3, (\#, \emptyset)) = (q_N, (\emptyset, \emptyset), (S, S))$$

Độ phức tạp thời gian:

Quá trình hoạt động của máy: đầu tiên, máy ghi ký tự # vào đầu băng 2 để đánh dấu điểm bắt đầu của băng. Tiếp theo, máy đọc từng ký tự của xâu phía trước dấu # của từ vào trên băng 1 và ghi xuống băng 2. Sau đó, Máy chuyển đầu đọc của băng 2 về phía đầu và bắt đầu so sánh từng ký tự của xâu sau dấu # trên băng 1 và xâu trước dấu # được ghi lại ở băng 2.

Trong quá trình tính toán của máy trên mỗi từ vào, việc bác bỏ những xâu không có dạng  $w\#w$  được thực hiện trong suốt quá trình soi xét và so sánh các ký tự trên 2 băng. Do đó, độ phức tạp thời gian  $t_A(n)$  của máy được xác định bởi thời gian thực hiện ghi dấu # đầu tiên trên băng 2, chuyển xâu trước dấu # xuống băng 2, thời gian di chuyển của đầu đọc về đầu băng 2 và thời gian so sánh các ký tự trên 2 băng.

không biết vẽ :) bỏ đi các bạn, ôn lý thuyết để kiểm điểm hơn, kệ các bạn dùng phao.

**TH:**  $w\#w^R$ ,  $w \in \{0, 1\}^+$

Máy turing đoán nhận xâu vào  $w\#w^R$  trên bộ chữ  $\{0, 1\}$  được xây dựng như sau:

$$M = \{Q, \{0,1,\#\}, \{0,1,\#, \emptyset\}, \delta, q_0, q_y, q_n\}$$

Trong đó  $Q = \{q_0, q_y, q_n, q_1, q_2, q_3\}$

Hàm chuyển được viết như sau:

1.  $\square(q_0, (\emptyset, \emptyset)) = (q_n, (\emptyset, \emptyset), (S, S))$
2.  $\square(q_0, (1, \emptyset)) = (q_1, (1, \#), (S, R))$
3.  $\square(q_0, (0, \emptyset)) = (q_1, (0, \#), (S, R))$
4.  $\square(q_0, (\#, \emptyset)) = (q_n, (\#, \emptyset), (S, S))$
5.  $\square(q_1, (1, \emptyset)) = (q_1, (1, 1), (R, R))$
6.  $\square(q_1, (0, \emptyset)) = (q_1, (0, 0), (R, R))$
7.  $\square(q_1, (\#, \emptyset)) = (q_2, (\#, \emptyset), (R, L))$
8.  $\square(q_1, (\emptyset, \emptyset)) = (q_n, (\emptyset, \emptyset), (S, S))$
9.  $\square(q_2, (1, 1)) = (q_2, (1, 1), (R, L))$
10.  $\square(q_2, (1, 0)) = (q_n, (1, 0), (S, S))$
11.  $\square(q_2, (1, \#)) = (q_n, (1, \#), (S, S))$
12.  $\square(q_2, (0, 0)) = (q_2, (0, 0), (R, L))$
13.  $\square(q_2, (0, 1)) = (q_n, (0, 1), (S, S))$
14.  $\square(q_2, (0, \#)) = (q_n, (0, \#), (S, S))$
15.  $\square(q_2, (\emptyset, 1)) = (q_n, (\emptyset, 1), (S, S))$
16.  $\square(q_2, (\emptyset, 0)) = (q_n, (\emptyset, 0), (S, S))$
17.  $\square(q_2, (\emptyset, \#)) = (q_y, (\emptyset, \#), (S, S))$
18.  $\square(q_2, (\#, 1)) = (q_n, (\#, 1), (S, S))$
19.  $\square(q_2, (\#, 0)) = (q_n, (\#, 0), (S, S))$
20.  $\square(q_2, (\#, \#)) = (q_n, (\#, \#), (S, S))$

**TH:  $w\#0w^R, w \in \{0, 1\}^+$**

Máy turing đoán nhận xâu vào  $w\#0w^R$  trên bộ chữ  $\{0, 1\}$  được xây dựng như sau:

$$M = \{Q, \{0,1,\#\}, \{0,1,\#, \emptyset\}, \delta, q_0, q_y, q_n\}$$

Trong đó  $Q = \{q_0, q_y, q_n, q_1, q_2, q_3\}$

Hàm chuyển được viết như sau:

$$\square(q_0, (\emptyset, \emptyset)) = (q_n, (\emptyset, \emptyset), (S, S))$$

$$\square(q_0, (1, \emptyset)) = (q_1, (1, \#), (S, R))$$

$$\square(q_0, (0, \emptyset)) = (q_1, (0, \#), (S, R))$$

$$\square(q_0, (\#, \emptyset)) = (q_N, (\#, \emptyset), (S, S))$$

$$\square(q_1, (1, \emptyset)) = (q_1, (1, 1), (R, R))$$

$$\square(q_1, (0, \emptyset)) = (q_1, (0, 0), (R, R))$$

$$\square(q_1, (\#, \emptyset)) = (q_2, (\#, \emptyset), (R, S))$$

$$\square(q_1, (\emptyset, \emptyset)) = (q_N, (\emptyset, \emptyset), (S, S))$$

$$\square(q_2, (\emptyset, \emptyset)) = (q_N, (\emptyset, \emptyset), (S, S))$$

$$\square(q_2, (1, \emptyset)) = (q_N, (1, \emptyset), (S, S))$$

$$\square(q_2, (0, \emptyset)) = (q_3, (0, \emptyset), (RS, L))$$

$$\square(q_2, (\#, \emptyset)) = (q_N, (\#, \emptyset), (S, S))$$

$$\square(q_3, (1, 1)) = (q_3, (1, 1), (R, L))$$

$$\square(q_3, (1, 0)) = (q_n, (1, 0), (S, S))$$

$$\square(q_3, (1, \#)) = (q_n, (1, \#), (S, S))$$

$$\square(q_3, (0, 1)) = (q_n, (0, 1), (S, S))$$

$$\square(q_3, (0, 0)) = (q_3, (0, 0), (R, L))$$

$$\square(q_3, (0, \#)) = (q_n, (0, \#), (S, S))$$

$$\square(q_3, (\#, 1)) = (q_n, (\#, 1), (S, S))$$

$$\square(q_3, (\#, 0)) = (q_n, (\#, 0), (S, S))$$

$$\square(q_3, (\#, \#)) = (q_n, (\#, \#), (S, S))$$

$$\square(q_3, (\emptyset, 1)) = (q_n, (\emptyset, 1), (S, S))$$

$$\square(q_3, (\emptyset, 0)) = (q_n, (\emptyset, 0), (S, S))$$

$$\square(q_3, (\emptyset, \#)) = (q_y, (\emptyset, \#), (S, S))$$

**TH:  $w\#0w$ ,  $w \in \{0, 1\}^+$**

Máy turing đoán nhận xâu vào  $w\#0w$  trên bộ chữ  $\{0, 1\}$  được xây dựng như sau:

$$M = \{Q, \{0, 1, \#\}, \{0, 1, \#, \emptyset\}, \delta, q_o, q_y, q_n\}$$

Trong đó  $Q = \{q_o, q_y, q_n, q_1, q_2, q_3, q_4\}$

Hàm chuyển được viết như sau:

$$\square(q_o, (\emptyset, \emptyset)) = (q_n, (\emptyset, \emptyset), (S, S))$$

$$\square(q_o, (1, \emptyset)) = (q_1, (1, \#), (S, R))$$

$$\square(q_o, (0, \emptyset)) = (q_1, (0, \#), (S, R))$$

$$\square(q_o, (\#, \emptyset)) = (q_n, (\#, \emptyset), (S, S))$$

$$\square(q_1, (1, \emptyset)) = (q_1, (1, 1), (R, R))$$

$$\square(q_1, (0, \emptyset)) = (q_1, (0, 0), (R, R))$$

$$\square(q_1, (\#, \emptyset)) = (q_2, (\#, \emptyset), (R, S))$$

$$\square(q_1, (\emptyset, \emptyset)) = (q_n, (\emptyset, \emptyset), (S, S))$$

$$\square(q_2, (\emptyset, \emptyset)) = (q_n, (\emptyset, \emptyset), (S, S))$$

$$\square(q_2, (1, \emptyset)) = (q_n, (1, \emptyset), (S, S))$$

$$\square(q_2, (0, \emptyset)) = (q_3, (0, \emptyset), (S, L))$$

$$\square(q_2, (\#, \emptyset)) = (q_N, (\#, \emptyset), (S, S))$$

$$\square(q_3, (0, 1)) = (q_3, (0, 1), (S, L))$$

$$\square(q_3, (0, 0)) = (q_3, (0, 0), (S, L))$$

$$\square(q_3, (0, \#)) = (q_4, (0, \#), (R, R))$$

$$\square(q_4, (0, 1)) = (q_n, (0, 1), (S, S))$$

$$\square(q_4, (0, 0)) = (q_4, (0, 0), (R, R))$$

$$\square(q_4, (0, \emptyset)) = (q_n, (0, \emptyset), (S, S))$$

$$\square(q_4, (1, 1)) = (q_4, (1, 1), (R, R))$$

$$\square(q_4, (1, 0)) = (q_n, (1, 0), (S, S))$$

$$\square(q_4, (1, \emptyset)) = (q_n, (1, \emptyset), (S, S))$$

$$\square(q_4, (\#, 1)) = (q_n, (\#, 1), (S, S))$$

$$\square(q_4, (\#, 0)) = (q_n, (\#, 0), (S, S))$$

$$\square(q_4, (\#, \emptyset)) = (q_n, (\#, \emptyset), (S, S))$$

$$\square(q_4, (\emptyset, \emptyset)) = (q_y, (\emptyset, \emptyset), (S, S))$$

$$\square(q_4, (\emptyset, 1)) = (q_n, (\emptyset, 1), (S, S))$$

$$\square(q_4, (\emptyset, 0)) = (q_n, (\emptyset, 0), (S, S))$$

**TH:**  $w\#00w, w \in \{0, 1\}^+$

Máy turing đoán nhận xâu vào  $w\#00w$  trên bộ chữ  $\{0, 1\}$  được xây dựng như sau:

$$M = \{Q, \{0,1,\#\}, \{0,1,\#, \emptyset\}, \delta, q_o, q_y, q_n\}$$

$$\text{Trong đó } Q = \{q_o, q_y, q_n, q_1, q_2, q_3\}$$

Hàm chuyển được viết như sau:

$$\square(q_o, (\emptyset, \emptyset)) = (q_N, (\emptyset, \emptyset), (S, S))$$

$$\square(q_o, (1, \emptyset)) = (q_1, (1, \#), (S, R))$$

$$\square(q_o, (0, \emptyset)) = (q_1, (0, \#), (S, R))$$

$$\square(q_o, (\#, \emptyset)) = (q_N, (\#, \emptyset), (S, S))$$

$$\square(q_1, (1, \emptyset)) = (q_1, (1, 1), (R, R))$$

$$\square(q_1, (0, \emptyset)) = (q_1, (0, 0), (R, R))$$

$$\square(q_1, (\#, \emptyset)) = (q_2, (\#, \emptyset), (R, S))$$

$$\square(q_1, (\emptyset, \emptyset)) = (q_N, (\emptyset, \emptyset), (S, S))$$

$$\square(q_2, (\emptyset, \emptyset)) = (q_N, (\emptyset, \emptyset), (S, S))$$

$$\square(q_2, (1, \emptyset)) = (q_N, (1, \emptyset), (S, S))$$

$$\square(q_2, (0, \emptyset)) = (q_3, (0, \emptyset), (S, L))$$

$$\square(q_2, (\#, \emptyset)) = (q_N, (\#, \emptyset), (S, S))$$

$$\square(q_3, (\emptyset, \emptyset)) = (q_N, (\emptyset, \emptyset), (S, S))$$

$$\square(q_3, (1, \emptyset)) = (q_N, (1, \emptyset), (S, S))$$

$$\square(q_3, (0, \emptyset)) = (q_4, (0, \emptyset), (S, L))$$

$$\square(q_3, (\#, \emptyset)) = (q_N, (\#, \emptyset), (S, S))$$



$$\square(q_4, (0, 1)) = (q_4, (0, 1), (S, L))$$

$$\square(q_4, (0, 0)) = (q_4, (0, 0), (S, L))$$

$$\square(q_4, (0, \#)) = (q_5, (0, \#), (R, R))$$

$$\square(q_5, (0, 1)) = (q_n, (0, 1), (S, S))$$

$$\square(q_5, (0, 0)) = (q_5, (0, 0), (R, R))$$

$$\square(q_5, (0, \emptyset)) = (q_n, (0, \emptyset), (S, S))$$

$$\square(q_5, (1, 1)) = (q_5, (1, 1), (R, R))$$

$$\square(q_5, (1, 0)) = (q_n, (1, 0), (S, S))$$

$$\square(q_5, (1, \emptyset)) = (q_n, (1, \emptyset), (S, S))$$

$$\square(q_5, (\#, 1)) = (q_n, (\#, 1), (S, S))$$

$$\square(q_5, (\#, 0)) = (q_n, (\#, 0), (S, S))$$

$$\square(q_5, (\#, \emptyset)) = (q_n, (\#, \emptyset), (S, S))$$

$$\square(q_5, (\emptyset, \emptyset)) = (q_y, (\emptyset, \emptyset), (S, S))$$

$$\square(q_5, (\emptyset, 1)) = (q_n, (\emptyset, 1), (S, S))$$

$$\square(q_5, (\emptyset, 0)) = (q_n, (\emptyset, 0), (S, S))$$

**TH:**  $w\#00w^R, w \in \{0, 1\}^+$

Máy turing đoán nhận xâu vào  $w\#00w^R$  trên bộ chữ  $\{0, 1\}$  được xây dựng như sau:

$$M = \{Q, \{0, 1, \#\}, \{0, 1, \#, \emptyset\}, \delta, q_o, q_y, q_n\}$$

Trong đó  $Q = \{q_o, q_y, q_n, q_1, q_2, q_3, q_4\}$

Hàm chuyển được viết như sau:

$$\square(q_0, (\emptyset, \emptyset)) = (q_N, (\emptyset, \emptyset), (S, S))$$

$$\square(q_0, (1, \emptyset)) = (q_1, (1, \#), (S, R))$$

$$\square(q_0, (0, \emptyset)) = (q_1, (0, \#), (S, R))$$

$$\square(q_0, (\#, \emptyset)) = (q_N, (\#, \emptyset), (S, S))$$

$$\square(q_1, (1, \emptyset)) = (q_1, (1, 1), (R, R))$$

$$\square(q_1, (0, \emptyset)) = (q_1, (0, 0), (R, R))$$

$$\square(q_1, (\#, \emptyset)) = (q_2, (\#, \emptyset), (R, S))$$

$$\square(q_1, (\emptyset, \emptyset)) = (q_N, (\emptyset, \emptyset), (S, S))$$

$$\square(q_2, (\emptyset, \emptyset)) = (q_N, (\emptyset, \emptyset), (S, S))$$

$$\square(q_2, (1, \emptyset)) = (q_N, (1, \emptyset), (S, S))$$

$$\square(q_2, (0, \emptyset)) = (q_3, (0, \emptyset), (R, S))$$

$$\square(q_3, (0, \emptyset)) = (q_4, (0, \emptyset), (R, L))$$

$$\square(q_3, (1, \emptyset)) = (q_N, (1, \emptyset), (S, S))$$

$$\square(q_4, (\#, \emptyset)) = (q_N, (\#, \emptyset), (S, S))$$

$$\square(q_4, (1, 1)) = (q_4, (1, 1), (R, L))$$

$$\square(q_4, (1, 0)) = (q_n, (1, 0), (S, S))$$

$$\square(q_4, (1, \#)) = (q_n, (1, \#), (S, S))$$

$$\square(q_4, (0, 1)) = (q_n, (0, 1), (S, S))$$

$$\square(q_4, (0, 0)) = (q_4, (0, 0), (R, L))$$

$$\square(q_4, (0, \#)) = (q_n, (0, \#), (S, S))$$

$$\square(q_4, (\#, 1)) = (q_n, (\#, 1), (S, S))$$

$$\square(q_4, (\#, 0)) = (q_n, (\#, 0), (S, S))$$

$$\square(q_4, (\#, \#)) = (q_n, (\#, \#), (S, S))$$

$$\square(q_4, (\emptyset, 1)) = (q_n, (\emptyset, 1), (S, S))$$

$$\square(q_4, (\emptyset, 0)) = (q_n, (\emptyset, 0), (S, S))$$

$$\square(q_4, (\emptyset, \#)) = (q_y, (\emptyset, \#), (S, S))$$

XXX

### **Máy Turing 1 bảng tính hàm $f(n, m)$**

$$\text{TH: } f(n, m) = n * m$$

máy turing 1 bảng tính hàm  $f(n, m) = n * m$  được xây dựng như sau:

$$M = \{Q, \{0, 1\}, \{0, 1, \#, \emptyset, \otimes\}, \square, q_0, q_y, q_n\}$$

$$\text{TH: } f(n, m) = n * m - m + 1$$

máy turing 1 bảng tính hàm  $f(n, m) = n * m - m + 1$  được xây dựng như sau:

$$M = \{Q, \{0, 1\}, \{0, 1, \#, \emptyset, \otimes\}, \square, q_0, q_y, q_n\}$$

Trong đó  $Q = \{q_0, q_y, q_n, q_1, q_2, q_3, q_4, \dots\}$

**Hàm chuyển được viết như sau:**

$$\square(q_0, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_0, 1) = (q_N, 1, S)$$

$$\square(q_0, 0) = (q_1, \emptyset, R)$$

$$\square(q_1, 0) = (q_1, 0, R)$$

$$\square(q_1, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_1, 1) = (q_2, 1, R)$$

$$\square(q_2, 1) = (q_N, 1, S)$$

$$\square(q_2, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_2, 0) = (q_3, 0, R)$$

$$\square(q_3, 1) = (q_N, 1, S)$$

$$\square(q_3, 0) = (q_3, 0, R)$$

$$\square(q_3, \emptyset) = (q_4, \#, L)$$

$$\square(q_4, 0) = (q_4, 0, L)$$

$$\square(q_4, 1) = (q_5, 1, R)$$

$$\square(q_5, 0) = (q_6, \otimes, R)$$

$$\square(q_6, 0) = (q_6, 0, R)$$

$$\square(q_6, \#) = (q_6, \#, R)$$

$$\square(q_6, \emptyset) = (q_4, 0, L)$$

$$\square(q_4, \#) = (q_4, \#, L)$$

$$\square(q_4, \otimes) = (q_5, \otimes, R)_{++}$$

$$\square(q_5, \#) = (q_7, \#, L)$$

$$\square(q_7, \otimes) = (q_7, 0, L)$$

$$\square(q_7, 1) = (q_7, 1, L)$$

$$\square(q_7, 0) = (q_7, 0, L)$$

$$\square(q_7, \emptyset) = (q_0', \emptyset, R)$$

$$\square(q_0', 0) = (q_8, \emptyset, R)$$

$$\square(q_8, 0) = (q_8, 0, R)$$

$$\square(q_8, 1) = (q_5, 1, R)$$

$$0\square(q_0', 1) = (q_9, \emptyset, R)$$

$$\square(q_9, 0) = (q_{10}, \emptyset, R)$$

$$\square(q_{10}, 0) = (q_{10}, 0, R)$$

$$\square(q_{10}, \#) = (q_{10}, \#, R)$$

$$\square(q_{10}, \emptyset) = (q_{11}, \emptyset, L)$$

$$\square(q_{11}, 0) = (q_{12}, \emptyset, L)$$

$$\square(q_{12}, 0) = (q_{12}, 0, L)$$

$$\square(q_{12}, \#) = (q_{12}, \#, L)$$

$$\square(q_{12}, \emptyset) = (q_9, \emptyset, R)$$

$$\square(q_9, \#) = (q_y, 0, R)$$

**Độ phức tạp không gian:**

Sau khi tính được  $n \times m$ , 1 ký tự # và  $n \times m$  ký tự 0 được thêm vào sau từ vào.  
Sau quá trình tính toán, số ô sử dụng trên bảng giảm đi  $m$  ký tự cuối cùng nên việc tính không gian bộ nhớ sử dụng lớn nhất sau khi tính được  $n \times m$ .

**Để thấy, Trường hợp  $n = m$  sẽ cho kết quả lớn nhất tương ứng với số ký tự 0 được thêm vào là lớn nhất. Trong trường hợp đó, coi độ dài đầu vào là  $a$ , ta có  $n = m = (a - 1) / 2$ . Số ô nhớ cần cho việc tính toán của máy là  $s_A(a) = a + 1 + (a - 1)^2 / 4 + 1$ ; với 1 ký tự trống ở cuối sử dụng để đánh dấu điểm cuối. Như vậy, độ phức tạp không gian của máy là  $O(n^2)$ .**

**không đúng đâu :>>**

**TH:  $f(n, m) = n * m + m + 1$**

**máy turing 1 bằng tính hàm  $f(n, m) = n * m + m + 1$  được xây dựng như sau:**

**$M = \{Q, \{0, 1\}, \{0, 1, \#, \emptyset, \otimes\}, \square, q_0, q_y, q_n\}$**

**Trong đó  $Q = \{q_0, q_y, q_n, q_1, q_2, q_3, q_4, \dots\}$**

**Hàm chuyển được viết như sau:**

**$\square(q_0, \emptyset) = (q_N, \emptyset, S)$**

**$\square(q_0, 1) = (q_N, 1, S)$**

**$\square(q_0, 0) = (q_1, \emptyset, R)$**

**$\square(q_1, 0) = (q_1, 0, R)$**

**$\square(q_1, \emptyset) = (q_N, \emptyset, S)$**

**$\square(q_1, 1) = (q_2, 1, R)$**

**$\square(q_2, 1) = (q_N, 1, S)$**

**$\square(q_2, \emptyset) = (q_N, \emptyset, S)$**

**$\square(q_2, 0) = (q_3, 0, R)$**

**$\square(q_3, 1) = (q_N, 1, S)$**

**$\square(q_3, 0) = (q_3, 0, R)$**

**$\square(q_3, \emptyset) = (q_4, \#, L)$**

$$\square(q_4, 0) = (q_4, 0, L)$$

$$\square(q_4, 1) = (q_5, 1, R)$$

$$\square(q_5, 0) = (q_6, \otimes, R)$$

$$\square(q_6, 0) = (q_6, 0, R)$$

$$\square(q_6, \#) = (q_6, \#, R)$$

$$\square(q_6, \emptyset) = (q_4, 0, L)$$

$$\square(q_4, \#) = (q_4, \#, L)$$

$$\square(q_4, \otimes) = (q_5, \otimes, R)$$

$$\square(q_5, \#) = (q_7, \#, L)$$

$$\square(q_7, \otimes) = (q_7, 0, L)$$

$$\square(q_7, 1) = (q_7, 1, L)$$

$$\square(q_7, 0) = (q_7, 0, L)$$

$$\square(q_7, \emptyset) = (q_0', \emptyset, R)$$

$$\square(q_0', 0) = (q_8, \emptyset, R)$$

$$\square(q_8, 0) = (q_8, 0, R)$$

$$\square(q_8, 1) = (q_5, 1, R)$$

$$\square(q_0', 1) = (q_9, \emptyset, R)$$

$$\square(q_9, 0) = (q_{10}, \emptyset, R)$$

$$\square(q_{10}, 0) = (q_{10}, 0, R)$$

$$\square(q_{10}, \#) = (q_{10}, \#, R)$$

$$\square(q_{10}, \emptyset) = (q_{11}, 0, L)$$

$$\square(q_{11}, 0) = (q_{11}, 0, L)$$

$$\square(q_{11}, \#) = (q_{11}, \#, L)$$

$$\square(q_{11}, \emptyset) = (q_9, \emptyset, R)$$

$$\square(q_9, \#) = (q_y, 0, R)$$

$$TH: f(n, m) = n * m - n + 1$$

**máy turing 1 bảng tính hàm  $f(n, m) = n * m - n + 1$  được xây dựng như sau:**

$$M = \{Q, \{0, 1, \Omega\}, \{0, 1, \#, \emptyset, \otimes, \Omega\}, \square, q_0, q_y, q_n\}$$

$$\text{Trong đó } Q = \{q_0, q_y, q_n, q_1, q_2, q_3, q_4, \dots\}$$

**Hàm chuyển được viết như sau:**

$$\square(q_0, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_0, 1) = (q_N, 1, S)$$

$$\square(q_0, 0) = (q_1, \emptyset, R)$$

$$\square(q_1, 0) = (q_1, 0, R)$$

$$\square(q_1, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_1, 1) = (q_2, 1, R)$$

$$\square(q_2, 1) = (q_N, 1, S)$$

$$\square(q_2, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_2, 0) = (q_3, 0, R)$$

$$\square(q_3, 1) = (q_N, 1, S)$$

$$\square(q_3, 0) = (q_3, 0, R)$$

$$\square(q_3, \emptyset) = (q_4, \#, L)$$

$$\square(q_4, 0) = (q_4, 0, L)$$



$$\square(q_4, 1) = (q_5, 1, R)$$

$$\square(q_5, 0) = (q_6, \otimes, R)$$

$$\square(q_6, 0) = (q_6, 0, R)$$

$$\square(q_6, \#) = (q_6, \#, R)$$

$$\square(q_6, \emptyset) = (q_4, 0, L)$$

$$\square(q_4, \#) = (q_4, \#, L)$$

$$\square(q_4, \otimes) = (q_5, \otimes, R)$$

$$\square(q_5, \#) = (q_7, \#, L)$$

$$\square(q_7, \otimes) = (q_7, 0, L)$$

$$\square(q_7, 1) = (q_7, 1, L)$$

$$\square(q_7, 0) = (q_7, 0, L)$$

$$\square(q_7, \emptyset) = (q_0', \emptyset, R)$$

$$\square(q_0', 0) = (q_8, \emptyset, R)$$

$$\square(q_8, 0) = (q_8, 0, R)$$

$$\square(q_8, 1) = (q_5, 1, R)$$

$$\square(q_0', 1) = (q_9, 1, L)$$

$$\square(q_9, \emptyset) = (q_{10}, 0, R)$$

$$\square(q_{10}, 0) = (q_{10}, 0, R)$$

$$\square(q_{10}, 1) = (q_{10}, 1, R)$$

$$\square(q_{10}, \#) = (q_{10}, \#, R)$$

$$\square(q_{10}, \emptyset) = (q_{11}, \emptyset, L)$$

$$\square(q_{11}, 0) = (q_{12}, \emptyset, L)$$

$$\square(q_{12}, 0) = (q_{12}, 0, L)$$

$$\square(q_{12}, \#) = (q_{12}, \#, L)$$

$$\square(q_{12}, 1) = (q_{12}, 1, L)$$

$$\square(q_{12}, \emptyset) = (q_{10}, 0, R)$$

$$\square(q_{12}, \Omega) = (q_{13}, \emptyset, R)$$

$$\square(q_{13}, 0) = (q_{13}, \emptyset, R)$$

$$\square(q_{13}, 1) = (q_{13}, \emptyset, R)$$

$$\square(q_{13}, \#) = (q_y, 0, R)$$

$$\text{TH: } f(n, m) = n * m + n + 1$$

máy turing 1 bằng tính hàm  $f(n, m) = n * m + n + 1$  được xây dựng như sau:

$$M = \{Q, \{0, 1\}, \{0, 1, \#, \emptyset, \Omega\}, \square, q_0, q_y, q_n\}$$

$$\text{Trong đó } Q = \{q_0, q_y, q_n, q_1, q_2, q_3, q_4, \dots\}$$

Hàm chuyển được viết như sau:

$$\square(q_0, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_0, 1) = (q_N, 1, S)$$

$$\square(q_0, 0) = (q_1, \emptyset, R)$$

$$\square(q_1, 0) = (q_1, 0, R)$$

$$\square(q_1, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_1, 1) = (q_2, 1, R)$$

$$\square(q_2, 1) = (q_N, 1, S)$$

$$\square(q_2, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_2, 0) = (q_3, 0, R)$$

$$\square(q_3, 1) = (q_N, 1, S)$$

$$\square(q_3, 0) = (q_3, 0, R)$$

$$\square(q_3, \emptyset) = (q_4, \#, L)$$

$$\square(q_4, 0) = (q_4, 0, L)$$

$$\square(q_4, 1) = (q_5, 1, R)$$

$$\square(q_5, 0) = (q_6, \otimes, R)$$

$$\square(q_6, 0) = (q_6, 0, R)$$

$$\square(q_6, \#) = (q_6, \#, R)$$

$$\square(q_6, \emptyset) = (q_4, 0, L)$$

$$\square(q_4, \#) = (q_4, \#, L)$$

$$\square(q_4, \otimes) = (q_5, \otimes, R)$$

$$\square(q_5, \#) = (q_7, \#, L)$$

$$\square(q_7, \otimes) = (q_7, 0, L)$$

$$\square(q_7, 1) = (q_7, 1, L)$$

$$\square(q_7, 0) = (q_7, 0, L)$$

$$\square(q_7, \emptyset) = (q_0', \emptyset, R)$$

$$\square(q_0', 0) = (q_8, \emptyset, R)$$

$$\square(q_8, 0) = (q_8, 0, R)$$

$$\square(q_8, 1) = (q_5, 1, R)$$

$$\square(q_0', 1) = (q_9, 1, L)$$

$$\square(q_9, \emptyset) = (q_{10}, 0, R)$$

$$\square(q_{10}, 0) = (q_{10}, 0, R)$$

$$\square(q_{10}, 1) = (q_{10}, 1, R)$$

$$\square(q_{10}, \#) = (q_{10}, \#, R)$$

$$\square(q_{10}, \emptyset) = (q_{11}, 0, L)$$

$$\square(q_{11}, 0) = (q_{11}, 0, L)$$

$$\square(q_{11}, \#) = (q_{11}, \#, L)$$

$$\square(q_{11}, 1) = (q_{11}, 1, L)$$

$$\square(q_{11}, \emptyset) = (q_{10}, 0, R)$$

$$\square(q_{11}, \Omega) = (q_{12}, \emptyset, R)$$

$$\square(q_{12}, 0) = (q_{12}, \emptyset, R)$$

$$\square(q_{12}, 1) = (q_{12}, \emptyset, R)$$

$$\square(q_{12}, \#) = (q_y, 0, R)$$

$$\text{TH: } f(n, m) = n * m - m - 1$$

máy turing 1 bảng tính hàm  $f(n, m) = n * m - m - 1$  được xây dựng như sau:

$$M = \{Q, \{0, 1\}, \{0, 1, \#, \emptyset, \otimes\}, \square, q_0, q_y, q_n\}$$

$$\text{Trong đó } Q = \{q_0, q_y, q_n, q_1, q_2, q_3, q_4, \dots\}$$

Hàm chuyển được viết như sau:

$$\square(q_0, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_0, 1) = (q_N, 1, S)$$

$$\square(q_0, 0) = (q_1, \emptyset, R)$$

$$\square(q_1, 0) = (q_1, 0, R)$$

$$\square(q_1, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_1, 1) = (q_2, 1, R)$$

$$\square(q_2, 1) = (q_N, 1, S)$$

$$\square(q_2, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_2, 0) = (q_3, 0, R)$$

$$\square(q_3, 1) = (q_N, 1, S)$$

$$\square(q_3, 0) = (q_3, 0, R)$$

$$\square(q_3, \emptyset) = (q_4, \#, L)$$

$$\square(q_4, 0) = (q_4, 0, L)$$

$$\square(q_4, 1) = (q_5, 1, R)$$

$$\square(q_5, 0) = (q_6, \otimes, R)$$

$$\square(q_6, 0) = (q_6, 0, R)$$

$$\square(q_6, \#) = (q_6, \#, R)$$

$$\square(q_6, \emptyset) = (q_4, 0, L)$$

$$\square(q_4, \#) = (q_4, \#, L)$$

$$\square(q_4, \otimes) = (q_5, \otimes, R)$$

$$\square(q_5, \#) = (q_7, \#, L)$$

$$\square(q_7, \otimes) = (q_7, 0, L)$$

$$\square(q_7, 1) = (q_7, 1, L)$$

$$\square(q_7, 0) = (q_7, 0, L)$$

$$\square(q_7, \emptyset) = (q_0', \emptyset, R)$$

$$\square(q_0', 0) = (q_8, \emptyset, R)$$

$$\square(q_8, 0) = (q_8, 0, R)$$

$$\square(q_8, 1) = (q_5, 1, R)$$

$$\square(q_0', 1) = (q_9, \emptyset, R)$$

$$\square(q_9, 0) = (q_{10}, \emptyset, R)$$

$$\square(q_{10}, 0) = (q_{10}, 0, R)$$

$$\square(q_{10}, \#) = (q_{10}, \#, R)$$

$$\square(q_{10}, \emptyset) = (q_{11}, \emptyset, L)$$

$$\square(q_{11}, 0) = (q_{12}, \emptyset, L)$$

$$\square(q_{12}, 0) = (q_{12}, 0, L)$$

$$\square(q_{12}, \#) = (q_{12}, \#, L)$$

$$\square(q_{12}, \emptyset) = (q_9, \emptyset, R)$$

$$\square(q_9, \#) = (q_{13}, \emptyset, R)$$

$$\square(q_{13}, 0) = (q_y, \emptyset, R)$$

$$\text{TH: } f(n, m) = n \cdot m + m - 1$$

máy turing 1 bảng tính hàm  $f(n, m) = n \cdot m + m - 1$  được xây dựng như sau:

$$M = \{Q, \{0, 1\}, \{0, 1, \#, \emptyset, \otimes\}, \square, q_0, q_y, q_n\}$$

$$\text{Trong đó } Q = \{q_0, q_y, q_n, q_1, q_2, q_3, q_4, \dots\}$$

Hàm chuyển được viết như sau:

$$\square(q_0, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_0, 1) = (q_N, 1, S)$$

$$\square(q_0, 0) = (q_1, \emptyset, R)$$

$$\square(q_1, 0) = (q_1, 0, R)$$

$$\square(q_1, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_1, 1) = (q_2, 1, R)$$

$$\square(q_2, 1) = (q_N, 1, S)$$

$$\square(q_2, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_2, 0) = (q_3, 0, R)$$

$$\square(q_3, 1) = (q_N, 1, S)$$

$$\square(q_3, 0) = (q_3, 0, R)$$

$$\square(q_3, \emptyset) = (q_4, \#, L)$$

$$\square(q_4, 0) = (q_4, 0, L)$$

$$\square(q_4, 1) = (q_5, 1, R)$$

$$\square(q_5, 0) = (q_6, \otimes, R)$$

$$\square(q_6, 0) = (q_6, 0, R)$$

$$\square(q_6, \#) = (q_6, \#, R)$$

$$\square(q_6, \emptyset) = (q_4, 0, L)$$

$$\square(q_4, \#) = (q_4, \#, L)$$

$$\square(q_4, \otimes) = (q_5, \otimes, R)$$

$$\square(q_5, \#) = (q_7, \#, L)$$

$$\square(q_7, \otimes) = (q_7, 0, L)$$

$$\square(q_7, 1) = (q_7, 1, L)$$

$$\square(q_7, 0) = (q_7, 0, L)$$

$$\square(q_7, \emptyset) = (q_0', \emptyset, R)$$

$$\square(q_0', 0) = (q_8, \emptyset, R)$$

$$\square(q_8, 0) = (q_8, 0, R)$$

$$\square(q_8, 1) = (q_5, 1, R)$$

$$\square(q_0', 1) = (q_9, \emptyset, R)$$

$$\square(q_9, 0) = (q_{10}, \emptyset, R)$$

$$\square(q_{10}, 0) = (q_{10}, 0, R)$$

$$\square(q_{10}, \#) = (q_{10}, \#, R)$$

$$\square(q_{10}, \emptyset) = (q_{11}, 0, L)$$

$$\square(q_{11}, 0) = (q_{11}, 0, L)$$

$$\square(q_{11}, \#) = (q_{11}, \#, L)$$

$$\square(q_{11}, \emptyset) = (q_9, \emptyset, R)$$

$$\square(q_9, \#) = (q_{12}, \emptyset, R)$$

$$\square(q_{12}, 0) = (q_y, \emptyset, R)$$

$$\text{TH: } f(n, m) = n * m - n - 1$$

máy turing 1 bảng tính hàm  $f(n, m) = n * m - n - 1$  được xây dựng như sau:

$$M = \{Q, \{0, 1\}, \{0, 1, \#, \emptyset, \otimes, \Omega\}, \square, q_0, q_y, q_n\}$$

$$\text{Trong đó } Q = \{q_0, q_y, q_n, q_1, q_2, q_3, q_4, \dots\}$$

Hàm chuyển được viết như sau:

$$\square(q_0, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_0, 1) = (q_N, 1, S)$$

$$\square(q_0, 0) = (q_1, \emptyset, R)$$

$$\square(q_1, 0) = (q_1, 0, R)$$



$$\square(q_1, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_1, 1) = (q_2, 1, R)$$

$$\square(q_2, 1) = (q_N, 1, S)$$

$$\square(q_2, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_2, 0) = (q_3, 0, R)$$

$$\square(q_3, 1) = (q_N, 1, S)$$

$$\square(q_3, 0) = (q_3, 0, R)$$

$$\square(q_3, \emptyset) = (q_4, \#, L)$$

$$\square(q_4, 0) = (q_4, 0, L)$$

$$\square(q_4, 1) = (q_5, 1, R)$$

$$\square(q_5, 0) = (q_6, \otimes, R)$$

$$\square(q_6, 0) = (q_6, 0, R)$$

$$\square(q_6, \#) = (q_6, \#, R)$$

$$\square(q_6, \emptyset) = (q_4, 0, L)$$

$$\square(q_4, \#) = (q_4, \#, L)$$

$$\square(q_4, \otimes) = (q_5, \otimes, R)$$

$$\square(q_5, \#) = (q_7, \#, L)$$

$$\square(q_7, \otimes) = (q_7, 0, L)$$

$$\square(q_7, 1) = (q_7, 1, L)$$

$$\square(q_7, 0) = (q_7, 0, L)$$

$$\square(q_7, \emptyset) = (q_0', \emptyset, R)$$

$$\square(q_0', 0) = (q_8, \emptyset, R)$$

$$\square(q_8, 0) = (q_8, 0, R)$$

$$\square(q_8, 1) = (q_5, 1, R)$$

$$\square(q_0', 1) = (q_9, 1, L)$$

$$\square(q_9, \emptyset) = (q_{10}, 0, R)$$

$$\square(q_{10}, 0) = (q_{10}, 0, R)$$

$$\square(q_{10}, 1) = (q_{10}, 1, R)$$

$$\square(q_{10}, \#) = (q_{10}, \#, R)$$

$$\square(q_{10}, \emptyset) = (q_{11}, \emptyset, L)$$

$$\square(q_{11}, 0) = (q_{12}, \emptyset, L)$$

$$\square(q_{12}, 0) = (q_{12}, 0, L)$$

$$\square(q_{12}, \#) = (q_{12}, \#, L)$$

$$\square(q_{12}, 1) = (q_{12}, 1, L)$$

$$\square(q_{12}, \emptyset) = (q_{10}, 0, R)$$

$$\square(q_{12}, \Omega) = (q_{13}, \emptyset, R)$$

$$\square(q_{13}, 0) = (q_{13}, \emptyset, R)$$

$$\square(q_{13}, 1) = (q_{13}, \emptyset, R)$$

$$\square(q_{13}, \#) = (q_{14}, \emptyset, R)$$

$$\square(q_{14}, 0) = (q_y, \emptyset, R)$$

$$\text{TH: } f(n, m) = n * m + n - 1$$

máy turing 1 bảng tính hàm  $f(n, m) = n * m + n - 1$  được xây dựng như sau:

$$M = \{Q, \{0, 1\}, \{0, 1, \#, \emptyset, \square, \Omega\}, \square, q_0, q_y, q_n\}$$

Trong đó  $Q = \{q_0, q_y, q_n, q_1, q_2, q_3, q_4, \dots\}$

Hàm chuyển được viết như sau:

$$\square(q_0, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_0, 1) = (q_N, 1, S)$$

$$\square(q_0, 0) = (q_1, \emptyset, R)$$

$$\square(q_1, 0) = (q_1, 0, R)$$

$$\square(q_1, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_1, 1) = (q_2, 1, R)$$

$$\square(q_2, 1) = (q_N, 1, S)$$

$$\square(q_2, \emptyset) = (q_N, \emptyset, S)$$

$$\square(q_2, 0) = (q_3, 0, R)$$

$$\square(q_3, 1) = (q_N, 1, S)$$

$$\square(q_3, 0) = (q_3, 0, R)$$

$$\square(q_3, \emptyset) = (q_4, \#, L)$$

$$\square(q_4, 0) = (q_4, 0, L)$$

$$\square(q_4, 1) = (q_5, 1, R)$$

$$\square(q_5, 0) = (q_6, \square, R)$$

$$\square(q_6, 0) = (q_6, 0, R)$$

$$\square(q_6, \#) = (q_6, \#, R)$$

$$\square(q_6, \emptyset) = (q_4, 0, L)$$

$$\square(q_4, \#) = (q_4, \#, L)$$

$$\square(q_4, \square) = (q_5, \square, R)$$

$$\square(q_5, \#) = (q_7, \#, L)$$

$$\square(q_7, \square) = (q_7, 0, L)$$

$$\square(q_7, 1) = (q_7, 1, L)$$

$$\square(q7, 0) = (q7, 0, L)$$

$$\square(q7, \emptyset) = (q0', \emptyset, R)$$

$$\square(q0', 0) = (q8, \emptyset, R)$$

$$\square(q8, 0) = (q8, 0, R)$$

$$\square(q8, 1) = (q5, 1, R)$$

$$\square(q0', 1) = (q9, 1, L)$$

$$\square(q9, \emptyset) = (q10, 0, R)$$

$$\square(q10, 0) = (q10, 0, R)$$

$$\square(q10, 1) = (q10, 1, R)$$

$$\square(q10, \#) = (q10, \#, R)$$

$$\square(q10, \emptyset) = (q11, 0, L)$$

$$\square(q11, 0) = (q11, 0, L)$$

$$\square(q11, \#) = (q11, \#, L)$$

$$\square(q11, 1) = (q11, 1, L)$$

$$\square(q11, \emptyset) = (q10, 0, R)$$

$$\square(q11, \Omega) = (q12, \emptyset, R)$$

$$\square(q12, 0) = (q12, \emptyset, R)$$

$$\square(q12, 1) = (q12, \emptyset, R)$$

$$\square(q12, \#) = (q13, \emptyset, R)$$

$$\square(q13, 0) = (qy, \emptyset, R)$$