ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

BÀI TẬP MÔN HỌC
PHÂN TÍCH THIẾT KẾ THUẬT TOÁN

Sinh viên: Đỗ Phương Duy - 23520362

Sinh viên: Nguyễn Nguyên Khang - 22520623

Ngày 26 tháng 10 năm 2024

Mục lục

1 Bài toán 1: Giải pháp cho Bài toán Tìm Tất cả Dãy Con Liên Tiếp của Chuỗi Số

Cho một chuỗi đầu vào gồm các số. Bài toán yêu cầu tìm tất cả các tổ hợp của các chữ số có thể được tạo ra bằng cách sử dụng các chữ số theo thứ tự liên tiếp như trong chuỗi ban đầu.

1.1 Giải pháp

Để tìm tất cả các dãy con liên tiếp của chuỗi đầu vào, chúng ta cần xét mọi tập hợp con của các chữ số có thể chọn theo thứ tự liên tiếp. Đối với một chuỗi có độ dài n, tổng số dãy con có thể có là $2^n - 1$, vì mỗi chữ số trong chuỗi có thể được chọn hoặc không chọn.

1.2 Độ phức tạp thời gian

Độ phức tạp thời gian của thuật toán này là $O(2^n)$, vì phải tạo ra tất cả các dãy con có thể có, dẫn đến độ phức tạp theo cấp số nhân.

1.3 Phương pháp giải

Ta có thể sử dụng phương pháp đệ quy hoặc quay lui để tạo ra tất cả các dãy con theo thứ tự xuất hiện của các chữ số trong chuỗi ban đầu. Một cách khác là triển khai theo phương pháp lặp, xem xét từng vị trí của chuỗi là bao gồm hoặc không bao gồm vào dãy con hiện tại.

1.4 Hàm đệ quy tìm dãy con

Hàm đệ quy sau đây sẽ tìm tất cả các dãy con liên tiếp của chuỗi đầu vào:

$$\text{Subsequences}(s, \text{index}, \text{current\_subseq}) = \begin{cases} \text{In ra current\_subseq} & \text{nếu i} \\ \text{Gọi Subsequences}(s, \text{index}+1, \text{current\_subseq}) & \text{(bỏ q} \\ \text{Gọi Subsequences}(s, \text{index}+1, \text{current\_subseq} + s[\text{index}]) & \text{(bao} \end{cases}$$

2 Bài toán 3: Viết chương trình in ra tất cả các tổ hợp các ước của một số cho trước n.

2.1 Giải pháp đệ quy

Để tìm tất cả các tổ hợp của các ước của n, chúng ta có thể thực hiện như sau:

1. Duyệt qua từng ước của n.
2. Đối với mỗi ước, chúng ta có hai lựa chọn: hoặc thêm ước vào tổ hợp hiện tại hoặc bỏ qua nó.
3. Khi hoàn thành tổ hợp (tức là không còn ước nào để xét), in ra tổ hợp đó.

2.2 Phân tích độ phức tạp

Giả sử k là số lượng ước của n. Vì mỗi ước có thể được chọn hoặc không chọn, tổng số tổ hợp sẽ là:

$$2^k$$

Do đó, độ phức tạp thời gian của thuật toán là:

$$O(2^k)$$

trong đó $k \approx \sqrt{n}$ vì các ước của n chỉ cần xét đến căn bậc hai của n.

3 Bài Toán 5: Giải pháp đệ quy cho Bài toán Tháp Hà Nội

Tháp Hà Nội là một câu đố toán học với ba cọc và n đĩa. Mục tiêu là di chuyển toàn bộ chồng đĩa từ cọc nguồn A sang cọc đích C, sử dụng cọc trung gian B, tuân theo các quy tắc sau:

- Chỉ một đĩa có thể di chuyển tại một thời điểm.
- Mỗi lần di chuyển bao gồm việc lấy đĩa trên cùng từ một cọc và đặt nó lên cọc khác.
- Không có đĩa nào có thể đặt trên đĩa nhỏ hơn.

3.1 Giải pháp đệ quy

Giải pháp đệ quy cho bài toán Tháp Hà Nội bao gồm các bước sau:

1. Di chuyển $n-1$ đĩa từ cọc nguồn A sang cọc trung gian B.
2. Di chuyển đĩa thứ n từ cọc A trực tiếp sang cọc đích C.
3. Di chuyển $n-1$ đĩa từ cọc trung gian B sang cọc đích C.

3.2 Độ phức tạp thời gian

Độ phức tạp thời gian của thuật toán này là $O(2^n)$, vì mỗi bước liên quan đến việc di chuyển đệ quy $n-1$ đĩa. Tổng số lần di chuyển tối thiểu để hoàn thành bài toán là:

$$T(n) = 2^n - 1$$

3.3 Hàm đệ quy

Hàm đệ quy để thực hiện bài toán Tháp Hà Nội có thể được định nghĩa như sau:

$$\text{Hanoi}(n, A, C, B) = \begin{cases} \text{Di chuyển 1 đĩa từ } A \text{ sang } C & \text{nếu } n = 1 \\ \text{Di chuyển } n-1 \text{ đĩa từ } A \text{ sang } B & \text{nếu } n > 1 \\ \text{Di chuyển đĩa } n \text{ từ } A \text{ sang } C & \\ \text{Di chuyển } n-1 \text{ đĩa từ } B \text{ sang } C & \end{cases}$$

Tài liệu