

## Bài tập 7: Phương pháp quy hoạch động

### I. Lí thuyết phương pháp

Ý tưởng của phương pháp quy hoạch động là tận dụng tính chất tái sử dụng của các phương án con của một bài toán để giảm thiểu thời gian tính toán.

Các yếu tố cơ bản của thiết kế giải thuật bằng phương pháp quy hoạch động bao gồm:

1. Tính tối ưu: Giải thuật quy hoạch động được thiết kế để tìm kiếm lời giải tối ưu cho một bài toán cụ thể.
2. Tính chia nhỏ: Giải thuật quy hoạch động sẽ chia bài toán ban đầu thành các bài toán con nhỏ hơn, từ đó tìm lời giải tối ưu cho các bài toán con này.
3. Tính lập trình động: Giải thuật quy hoạch động thường được thực hiện thông qua một hệ thống lập trình động, cho phép lưu trữ và sử dụng lại các kết quả tính toán trước đó để giảm thiểu thời gian tính toán.

Các bước trong thiết kế giải thuật quy hoạch động bao gồm:

1. Xác định cấu trúc tối ưu của bài toán: Các bài toán quy hoạch động thường có một cấu trúc đặc biệt, và bước đầu tiên của thiết kế giải thuật là xác định cấu trúc này.
2. Xác định phương án con: Sau khi xác định được cấu trúc của bài toán, giải thuật quy hoạch động sẽ chia bài toán thành các bài toán con nhỏ hơn.
3. Xác định hàm trạng thái và hàm quyết định: Bước này liên quan đến việc xác định các hàm để tính toán các giá trị tối ưu cho các bài toán con.
4. Lập trình động: Giải thuật quy hoạch động thường được thực hiện thông qua một hệ thống lập trình động, cho phép lưu trữ và sử dụng lại các kết quả tính toán trước đó để giảm thiểu thời gian tính toán.
5. Xác định lời giải tối ưu: Cuối cùng, giải thuật quy hoạch động sẽ tính toán và trả về lời giải tối ưu cho bài toán ban đầu.

### II. Lập trình

Coding trong folder `./src/bai_2`

+ Bài toán tìm dãy con đơn điệu tăng dài nhất (longest increasing subsequence) có độ phức tạp là  $O(n \log n)$  theo đoạn code trên

+ Bài toán xếp balo 0-1 (0-1 knapsack)

Trong đó:

`val` và `wt` lần lượt là mảng giá trị và trọng lượng của các vật phẩm

`W` là trọng lượng tối đa của balo

`n` là số lượng vật phẩm

Hàm `knapsack` trả về giá trị lớn nhất có thể đạt được với trọng lượng không vượt quá `W`

Độ phức tạp của thuật toán này là  $O(nW)$ , trong đó `n` là số lượng vật phẩm và `W` là trọng lượng tối đa của balo. Vì vậy, nếu `W` rất lớn, thì thuật toán này có thể trở nên chậm và không hiệu quả.

### III. Đặt bài toán, thiết kế, phân tích và triển khai thuật toán

*Đề bài toán: Cho một dãy số nguyên dương gồm  $n$  phần tử, hãy tìm một dãy con liên tiếp có tổng lớn nhất.*

Phân tích: Để giải quyết bài toán này, chúng ta có thể sử dụng phương pháp quy hoạch động. Để tìm dãy con liên tiếp có tổng lớn nhất, ta cần xác định dãy con có tổng lớn nhất bắt đầu từ mỗi phần tử của dãy. Với mỗi phần tử, ta sẽ tính tổng các phần tử kế tiếp của nó và lưu trữ kết quả này vào một mảng mới. Sau đó, ta tìm phần tử lớn nhất trong mảng mới này để xác định tổng lớn nhất của dãy con.

Thiết kế giải thuật:

1. Khởi tạo mảng  $dp$  gồm  $n$  phần tử, trong đó  $dp[i]$  là tổng lớn nhất của dãy con kết thúc tại vị trí  $i$ .
2. Gán  $dp[0] = a[0]$ .
3. Duyệt qua từng phần tử của dãy, với mỗi phần tử  $a[i]$ , ta tính tổng các phần tử kế tiếp của nó và lưu trữ kết quả này vào  $dp[i]$ .
4. Tìm phần tử lớn nhất trong mảng  $dp$  để xác định tổng lớn nhất của dãy con.

Độ phức tạp thời gian của thuật toán này là  $O(n)$ , với  $n$  là số lượng phần tử của dãy. Cụ thể, việc tính tổng các phần tử kế tiếp và lưu trữ kết quả vào mảng  $dp$  được thực hiện một lần duy nhất qua từng phần tử của dãy, nên độ phức tạp của quá trình tính toán là  $O(n)$ . Sau đó, việc tìm phần tử lớn nhất trong mảng  $dp$  cũng được thực hiện trong  $O(n)$  bằng cách duyệt qua từng phần tử của mảng. Do đó, tổng độ phức tạp thời gian của thuật toán là  $O(n)$ .

Coding trong folder `./src/bai_3/`