

Dynamic Programming

(Quy hoạch động)

Họ tên: Phạm Vũ Anh Quân

MSV: 19000470

1. Lý thuyết phương pháp

1.1. Ý tưởng

- Phương pháp Quy hoạch động giải quyết bài toán theo nguyên tắc sau:
 - Giải các bài toán con nhỏ hơn hoặc các bài toán con gộp nhau
 - Ghi lại kết quả của các bài toán con
 - Kết hợp các kết quả đã được ghi lại để có được giải pháp cuối cùng

1.2. Những yếu tố cơ bản trong thiết kế giải thuật bằng phương pháp QHĐ:

- Công thức truy hồi
- Cơ sở Quy hoạch động
- Bảng phương án
- Phương án tối ưu
- Truy vết tìm nghiệm

1.3. Các bước xây dựng giải thuật bằng phương pháp Quy hoạch động:

- B1: Nhận dạng bài toán giải bằng Quy hoạch động
- B2: Xây dựng công thức truy hồi
- B3: Xây dựng cơ sở Quy hoạch động
- B4: Dựng bảng phương án
- B5: Tìm phương án tối ưu
- B6: Truy vết liệt kê thành phần của nghiệm

2. Lập trình

- Làm 2 bài: Dãy con tăng dài nhất và Sắp xếp balo.
- Dãy con tăng dài nhất: lis.py
- Sắp xếp balo: knapsack01.py

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

- Đặt bài toán: Bài toán coin-row. Có 1 hàng gồm n đồng xu với giá trị lần lượt là c_1, c_2, \dots, c_n ($c_j > 0, j = 1..n$) không nhất thiết là phải khác nhau. Ta phải chọn được tổng số tiền lớn nhất sao cho không có 2 đồng xu nào cạnh nhau cùng được chọn.
- Phân tích:
 - Gọi $F(n)$ là tổng số tiền lớn nhất có thể chọn từ n đồng xu. Ta chia tất cả đồng xu có thể được chọn thành 2 nhóm: Nhóm chứa đồng xu cuối cùng và nhóm không chứa đồng xu cuối cùng.
 - Tổng số tiền lớn nhất có thể thu được từ nhóm thứ nhất là:
$$c_n + F(n - 2)$$
 - Tổng số tiền lớn nhất có thể thu được từ nhóm thứ hai là: $F(n - 1)$

- Từ phân tích ở trên, ta có công thức truy hồi là:
$$F(n) = \max\{c_n + F(n - 2), F(n - 1)\}, n > 1$$
 - Với cơ sở Quy hoạch động là:
 - $F(0) = 0$
 - $F(1) = c_1$
- Triển khai thuật toán: coinrow.py

(*)

Source code đặt tại thư mục 'src'.

Thực thi chương trình: ./filename hoặc python filename

Cài đặt thư viện

- Matplotlib: `pip install matplotlib`