

BÀI 9: THỰC HÀNH CHIẾN LƯỢC THAM LAM, QUI HOẠCH ĐỘNG

I - Chiến lược tham lam

- Chiến lược thiết kế thuật toán tìm lời giải tối ưu cho bài toán theo từng bước.
- Lấy tiêu chuẩn toàn cục làm tiêu chuẩn cục bộ: Tại mỗi bước, ta lựa chọn một khả năng tốt nhất tại điểm đó (tối ưu cục bộ) mà không quan tâm tới tương lai.
- Hi vọng tập hợp các lời giải tối ưu cục bộ chính là lời giải tối ưu cần tìm.
- Chiến lược tham lam có thể áp dụng tốt cho các bài toán tối ưu có 2 đặc điểm sau:
 - o Greedy-choice property: Một nghiệm tối ưu toàn cục có thể được xây dựng bằng cách lựa chọn các nghiệm tối ưu địa phương.
 - o Optimal substructure: Một nghiệm tối ưu của bài toán lại chứa một nghiệm tối ưu của các bài toán con của nó.

II - Chiến lược quy hoạch động

- Quy hoạch động là một chiến lược nhằm đơn giản hóa việc tính toán các công thức truy hồi bằng cách lưu trữ toàn bộ hay một phần kết quả tính toán tại mỗi bước với mục đích sử dụng lại.
- Bản chất của quy hoạch động là thay thế mô hình tính toán top – down bằng mô hình tính toán bottom – up.
- Quy hoạch động giúp giải quyết các bài toán tối ưu mang bản chất đệ quy.
- Đặc điểm chung của quy hoạch động:
 - o Quy hoạch động bắt đầu từ việc giải tất cả các bài toán nhỏ nhất (bài toán cơ sở) để từ đó từng bước giải quyết những bài toán lớn hơn cho tới khi giải được bài toán lớn nhất (bài toán ban đầu).
 - o Quy hoạch động cần phải có bảng phương án.

Ý tưởng cơ bản của phương pháp quy hoạch động là tránh tính toán lại các bài toán con đã xét.

BÀI TẬP THỰC HÀNH TRÊN LỚP

Bài tập 1: COIN CHANGING.

Input:

- Cho trước n loại tiền xu có mệnh giá lần lượt là x_1, x_2, \dots, x_n và một số tiền m .
- Giả sử rằng mỗi loại mệnh giá tiền xu có số lượng đồng xu là vô hạn.

Output:

- Tính p số lượng đồng xu ít nhất cần lấy để có tổng mệnh giá bằng m .

Thực hiện các yêu cầu sau:

- Thiết kế thuật toán A1 để tính và trả về p số lượng đồng xu ít nhất cần lấy để có tổng mệnh giá bằng m bằng chiến lược tham lam.
- Cài đặt chương trình:
 - o Cài đặt hàm F1 biểu diễn thuật toán A1.
 - o Khởi tạo n, m và $\{x_1, x_2, \dots, x_n\}$.
 - o Sử dụng thuật toán A1 để tính và đưa ra p .

Bài tập 2: Chiết nước.

Input:

- Cho một bình có dung tích d chứa đầy nước và n chiếc chai rỗng có dung tích khác nhau để chiết nước từ bình vào đầy các chai.
- Dung tích lần lượt của các chai là x_1, x_2, \dots, x_n .

Output:

- Tính q là số lượng chai tối đa được đổ đầy nước.

Thực hiện các yêu cầu sau:

- Thiết kế thuật toán A2 để tính q là số lượng chai tối đa được đổ đầy nước bằng chiến lược tham lam.
- Cài đặt chương trình:
 - o Cài đặt hàm F2 biểu diễn thuật toán A2.
 - o Khởi tạo d và $\{x_1, x_2, \dots, x_n\}$.
 - o Sử dụng thuật toán A2 để tính và đưa ra q .

Bài tập 3: INTERVAL SCHEDULING.

Input:

- Có n công việc, thời điểm bắt đầu của các công việc lần lượt là s_1, s_2, \dots, s_n và thời điểm kết thúc của các công việc lần lượt là f_1, f_2, \dots, f_n .

- Hai công việc i và j được gọi là tương hợp nếu thời gian thực hiện của chúng không giao nhau ($f_i \leq s_j$ hoặc $f_j \leq s_i$).

Output:

- Tính c là một tập cực đại các công việc mà chúng tương hợp với nhau.

Thực hiện các yêu cầu sau:

- Thiết kế thuật toán A3 để tính và trả về c là tập cực đại các công việc tương hợp với nhau bằng chiến lược tham lam.
- Cài đặt chương trình:
 - o Cài đặt hàm F3 biểu diễn thuật toán A3.
 - o Khởi tạo n , $\{s_1, s_2, \dots, s_n\}$ và $\{f_1, f_2, \dots, f_n\}$.
 - o Sử dụng thuật toán A3 để tính và đưa ra c .

Bài tập 4: Xếp ba lô

Input:

- Cho một kho chứa n gói hàng được ghi số thứ tự từ 1 đến n .
- Các gói hàng có khối lượng lần lượt là m_1, m_2, \dots, m_n và kích thước lần lượt là k_1, k_2, \dots, k_n .
- Cho một chiếc ba lô có kích thước k .

Output:

- Tính p và d lần lượt là tổng khối lượng và danh sách các gói hàng được chọn để đưa vào ba lô sao cho tổng khối lượng là nhỏ nhất và tổng kích thước của các gói hàng được chọn không vượt quá kích thước của ba lô.

Thực hiện các yêu cầu sau:

- Thiết kế thuật toán A6 để tính và trả về p và d lần lượt là tổng khối lượng và danh sách các gói hàng được chọn để đưa vào ba lô sao cho tổng khối lượng là nhỏ nhất và tổng kích thước của các gói hàng được chọn không vượt quá kích thước của ba lô.
- Cài đặt chương trình:
 - o Cài đặt hàm F6 biểu diễn thuật toán A6.
 - o Khởi tạo $k, n, \{m_1, m_2, \dots, m_n\}$ và $\{k_1, k_2, \dots, k_n\}$.
 - o Sử dụng thuật toán A6 để tính và đưa ra p và d .

○

Bài tập 5: Phân tích số thành tổng

Input: Cho số tự nhiên $n \leq 100$.

Output: Tính p là số cách phân tích số n thành tổng của dãy các số nguyên dương.

Yêu cầu thực hiện:

- Thiết kế thuật toán Q1 để tính và trả về p bằng chiến lược quy hoạch động.
- Cài đặt chương trình ứng dụng:
 - Cài đặt hàm biểu diễn thuật toán Q1.
 - Khởi tạo n .
 - Sử dụng thuật toán Q1 để tính và đưa ra p .

Bài tập 6: Bài toán cái túi

Input:

- Cho danh sách n gói hàng được đánh số thứ tự từ 1 đến n , các gói hàng có trọng lượng lần lượt là w_1, w_2, \dots, w_n và giá trị lần lượt là v_1, v_2, \dots, v_n .
- Cho một chiếc túi có thể chứa được khối lượng tối đa là m .

Output:

- Tính u và d lần lượt là số gói hàng và danh sách các gói hàng cần xếp vào cái túi sao cho giá trị lấy được là lớn nhất mà không vượt quá kích thước của cái túi.

Yêu cầu thực hiện:

- Thiết kế thuật toán Q2 để tính và trả về u và d bằng chiến lược quy hoạch động.
- Cài đặt chương trình ứng dụng:
 - Cài đặt hàm biểu diễn thuật toán Q2.
 - Khởi tạo $m, n, \{w_1, w_2, \dots, w_n\}$ và $\{v_1, v_2, \dots, v_n\}$.
 - Sử dụng thuật toán Q2 để tính và đưa ra u và d .

Bài tập 7: Dãy con đơn điệu tăng dài nhất.

Input:

- Cho số tự nhiên n và dãy a gồm n số thực a_1, a_2, \dots, a_n .

- Một dãy con của dãy a là một cách chọn ra trong dãy a một số phần tử giữ nguyên thứ tự (dãy a có 2^n dãy con như vậy).

Output:

- Tính b là một dãy con của dãy a với b gồm các phần tử có thứ tự tăng với số phần tử là nhiều nhất (dãy b được gọi là dãy con đơn điệu tăng dài nhất).

Yêu cầu thực hiện:

- Thiết kế thuật toán Q3 để tính và trả về b bằng chiến lược quy hoạch động.
- Cài đặt chương trình ứng dụng:
 - o Cài đặt hàm biểu diễn thuật toán Q3.
 - o Khởi tạo n và dãy a .
 - o Sử dụng thuật toán Q3 để tính và đưa ra b .

Giải thích:

s là mảng đánh dấu loại tiền nào cần dùng cuối cùng để đạt số tiền i .

Truy vết để tìm lại các loại tiền cần dùng bằng mảng s như sau:

```
j = s[m];
i = m;
while (j khác 0) {
    print(a[j]);
    i = i - j;
    j = s[i];
}
```

BÀI TẬP THỰC HÀNH SAU KHI LÊN LỚP

Bài tập 1: Vận tải 1

Input:

- Cho n chiếc xe tải với các xe có tải trọng khác nhau lần lượt là t_1, t_2, \dots, t_n .
- Kho hàng k chứa m tấn hàng cần vận chuyển đến địa điểm khác.

Output:

- Tính d là danh sách tối thiểu các xe tải cần sử dụng để vận chuyển hết m tấn hàng sao cho mỗi xe tải chỉ được sử dụng 1 lần.

Thực hiện các yêu cầu sau:

- Thiết kế thuật toán A4 để tính và trả về d là danh sách tối thiểu các xe tải được sử dụng để vận chuyển hết m tấn hàng bằng chiến lược tham lam.
- Cài đặt chương trình:
 - o Cài đặt hàm F4 biểu diễn thuật toán A4.
 - o Khởi tạo m , n và $\{t_1, t_2, \dots, t_n\}$.
 - o Sử dụng thuật toán A4 để tính và đưa ra d .

Bài tập 2: Vận tải 2

Input:

- Cho một chiếc xe tải với kích thước thùng xe là k .
- Cho n kiện hàng khác nhau, các kiện hàng có khối lượng lần lượt là m_1, m_2, \dots, m_n và kích thước lần lượt là k_1, k_2, \dots, k_n .

Output:

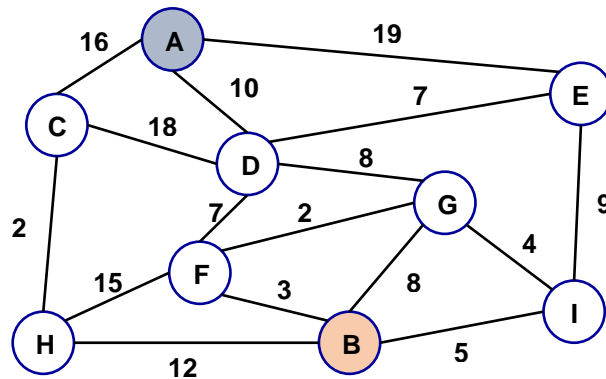
- Tính u, v lần lượt là số kiện hàng và danh sách các kiện hàng được xếp lên thùng xe tải sao cho tổng khối lượng xếp lên xe đạt được là lớn nhất mà không vượt quá kích thước của thùng xe.

Thực hiện các yêu cầu sau:

- Thiết kế thuật toán A5 để tính và trả về u và v lần lượt là số lượng và danh sách các kiện hàng được xếp lên thùng xe tải sao cho tổng khối lượng xếp lên xe đạt được là lớn nhất mà không vượt quá kích thước của thùng xe.
- Cài đặt chương trình:
 - o Cài đặt hàm F5 biểu diễn thuật toán A5.
 - o Khởi tạo $k, n, \{m_1, m_2, \dots, m_n\}$ và $\{k_1, k_2, \dots, k_n\}$.
 - o Sử dụng thuật toán A5 để tính và đưa ra u và v .

Bài tập 3: Travelling Sale Man

Input: Cho đồ thị vô hướng có trọng số như hình vẽ bên dưới.



Output: Tìm đường đi ngắn nhất từ một đỉnh A đến một đỉnh B trên đồ thị bằng chiến lược tham lam.

Yêu cầu thực hiện:

- Thiết kế thuật toán theo yêu cầu.
- Cài đặt chương trình ứng dụng thuật toán và biểu diễn kết quả.

Bài tập 4: Coin changing.

Input:

- Cho n loại tiền có mệnh giá lần lượt là c_1, c_2, \dots, c_n với số lượng tờ tiền mỗi loại mệnh giá không giới hạn.
- Cho một số tiền m .

Output:

- Tính $s = \{s_1, s_2, \dots, s_n\}$ là số tờ tiền mỗi loại cần lấy để chi trả vừa đủ số tiền m sao cho tổng số lượng tờ tiền phải trả t là ít nhất.

Yêu cầu thực hiện:

- Thiết kế thuật toán Q4 để tính và trả về s và t bằng chiến lược quy hoạch động.
- Cài đặt chương trình ứng dụng:
 - o Cài đặt hàm biểu diễn thuật toán Q4.
 - o Khởi tạo n và $\{c_1, c_2, \dots, c_n\}$.
 - o Sử dụng thuật toán Q4 để tính và đưa ra s và t .

Hướng dẫn:

- Gọi $f(i)$ là số lượng tờ ít nhất để trả số tiền i nên $f(m)$ là số tờ ít nhất để trả số tiền m .
- Để được số tiền là i có các cách để tạo thành số tiền đó khi chúng ta dùng thêm một tờ là: $i - c[k_1]$, $i - c[k_2]$, ..., $i - c[k_j]$, trong đó k_j là số thoả mãn mà $c[k_j] < i$.
- Số tờ tiền tối ưu nhất cần tìm là giá trị nhỏ nhất trong các giá trị:
 $f(i - c[k_1]) + 1$, $f(i - c[k_2]) + 1$, ..., $f(i - c[k_j]) + 1$.
- Công thức quy hoạch động như sau:
$$f(i) = \min\{f(i - c[j]) + 1, j \text{ thoả mãn: } c[j] < i\}$$
- Thuật toán quy hoạch động:

```
coin_changing() {  
    f[0] = 0;  
    for (i = 1 -> m) {  
        min = maxint;  
        for (j = 1 -> n) {  
            if (f[i - a[j]] + 1 < min && a[j] < i) {  
                min = f[i - a[j]] + 1;  
                s[i] = j;  
            }  
        }  
        f[i] = min;  
    }  
}
```