

BÀI 03

CHIẾN LƯỢC THAM LAM

Design by Minh An

Email: anvanminh.hau@gmail.com

1

Nội dung

- **Giới thiệu**
- **Bài toán tối ưu**
- **Phương pháp tham lam**
- **Giải quyết bài toán bằng chiến lược tham lam**

Design by Minh An

2

3.1. Giới thiệu

- **Bài toán: Coin changing**

- Giả sử rằng ta đang có các đồng xu tiền mệnh giá 100 đồng, 25 đồng, 10 đồng, 5 đồng và 1 đồng. Cũng giả sử rằng số đồng xu mỗi loại mệnh giá là vô hạn.
- Cho trước một số tiền bất kỳ n . Hãy cho biết số lượng đồng xu ít nhất cần lấy để có tổng mệnh giá bằng n .

- **Ví dụ:**



Design by Minh An

3

Coin changing

- **Phương pháp giải:**

- Tại mỗi bước lặp, lấy các đồng xu có mệnh giá cao nhất với số lượng nhiều nhất có thể.



INPUT: $C[], m, n$
OUTPUT: $S[]$

Design by Minh An

4

Coin changing

▪ Phương pháp giải:

- Giả sử C được sắp giảm (nếu chưa, xin sắp C).
- Chuẩn bị mảng S[] và khởi gán các phần tử của S bằng 0.
- Duyệt C từ trái qua phải. Với mỗi C[i]:
 - + S[i] = số đồng tiền có mệnh giá C[i] nhiều nhất có thể lấy mà tổng giá trị không vượt quá n
 - + Tính lại n = số tiền còn lại.
- Nếu duyệt hết C mà n > 0: → No solution
- Ngược lại: → return S[].

Design by Minh An

5

Coin changing

▪ Thuật toán

```
bool CASHIERS_ALGORITHM(int *C, int m, long n, int * S)
{
    Khởi tạo S[]: S[i]=0 ∀i = 0..m;
    i=0;
    while (n>0 && i<m)
    {
        S[i] = Số_đồng_C[i]_nhiều_nhất_có_thể_lấy;
        n = Số_tiền_còn_lại ;
        i++;
    }
    if(n>0)      return false;
    else         return true;
}
```

Design by Minh An

6

Coin changing

▪ Nhận xét

Xét trường hợp có ba loại mệnh giá: 1, 7, 10
Cho số tiền $n = 15$.

Giải thuật tham lam cho kết quả: 6
Nghiem tối ưu toàn cục: 3

Xét trường hợp có ba loại mệnh giá: 3, 7, 10
Cho số tiền $n = 12$.

Giải thuật tham lam: không tìm thấy nghiệm
Nghiem tối ưu: 4

Design by Minh An

7

3.2. Bài toán tối ưu

$\min f(x)$

$x \in D$

f : hàm mục tiêu

x : biến

D : miền xác định

$x^* = \arg \min f(x)$ được gọi là nghiệm tối ưu

Design by Minh An

8

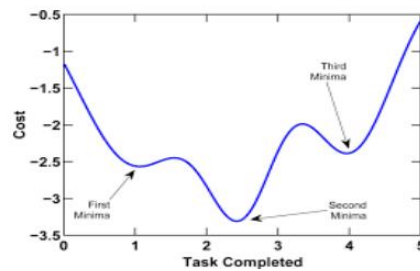
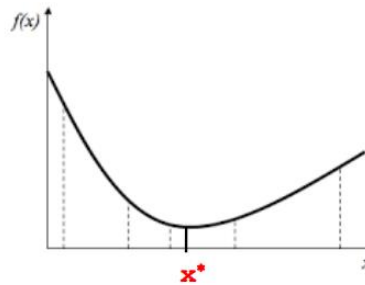
Bài toán tối ưu

Nghiệm tối ưu toàn cục

$$x^* : f(x) \geq f(x^*) \quad \forall x \in D$$

Nghiệm tối ưu địa phương

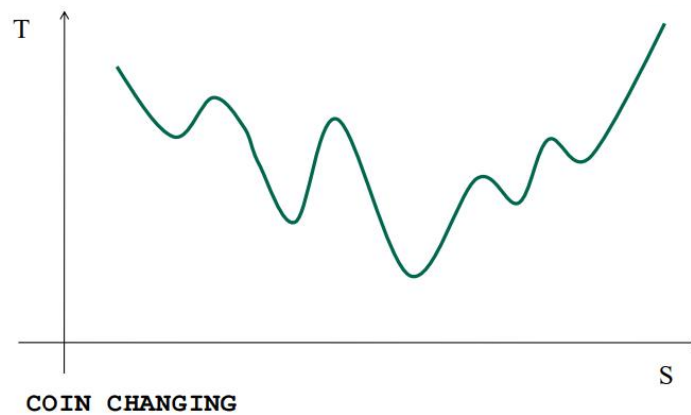
$$x^* : f(x) \geq f(x^*) \quad \forall x \in U \text{ với } U \text{ là lân cận của } x$$



Design by Minh An

9

Bài toán tối ưu



Design by Minh An

10

3.3. Giải thuật tham lam – Greedy Algorithm

- Tìm lời giải tối ưu cho bài toán theo từng bước
- 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
- **Trong một vài trường hợp** các giải thuật tham lam tìm được nghiệm tối ưu toàn cục bằng cách lặp đi lặp lại việc lựa chọn các khả năng tối ưu cục bộ
- **Trong nhiều trường hợp** phương pháp này có thể không cho lời giải tối ưu toàn cục hoặc thậm chí không đưa ra được lời giải

Design by Minh An

11

Giải thuật tham lam

- **Giải thuật tham lam là một chiến lược có thể áp dụng tốt cho các bài toán tối ưu có 2 đặc điểm sau:**
 - ✓ **Greedy-choice property:** Một nghiệm tối ưu toàn cục có thể được xây dựng bằng cách lựa chọn các nghiệm tối ưu địa phương
 - ✓ **Optimal substructure:** Một nghiệm tối ưu của bài toán lại chứa một nghiệm tối ưu của các bài toán con của nó

Design by Minh An

12

3. Giải thuật tham lam – Greedy Algorithm

Ưu điểm

- **Simplicity:** các giải thuật tham lam thường dễ dàng được mô tả và cài đặt hơn các giải thuật khác.
- **Efficiency:** các giải thuật tham lam thường có hiệu quả hơn các giải thuật khác.

Nhược điểm

- **Hard to design:** khi đã xác định được quy luật lựa chọn tham lam cho một bài toán, việc thiết kế giải thuật là dễ. Tuy nhiên, việc tìm ra quy luật tham lam là khó khăn.
- **Hard to verify:** khó để chứng minh một giải thuật tham lam là đúng.

Design by Minh An

13

3. Giải thuật tham lam – Greedy Algorithm

Bài tập 2: Một bình chứa chứa đầy nước với một lượng nước hữu hạn n . Cho m chiếc chai rỗng (dung tích các chai khác nhau) để chiết nước từ bình chứa vào đầy các chai. Hãy cho biết số lượng chai tối đa có thể được đổ đầy nước.

Input: dòng thứ nhất chứa hai số nguyên n và m ; dòng thứ 2 chứa m số nguyên là dung tích của các chai.

Output: một số nguyên là số chai tối đa được đổ đầy nước.

INPUT	OUTPUT
10 5 8 5 4 3 2	3

Design by Minh An

14

3. Giải thuật tham lam – Greedy Algorithm



Design by Minh An

15

3. Giải thuật tham lam – Greedy Algorithm

Thuật toán

```
pouring(jug, *bottle, n) {  
    //Sắp các chai theo thứ tự dung tích tăng dần  
    i = 1;  
    while (jug >= bottle[i] && i <= n) {  
        //Đổ nước từ bình vào đầy chai  
    }  
    //Nếu đổ được ít nhất 1 chai => có phương án  
    //Ngược lại: không có phương án  
}
```

Design by Minh An

16

3. Giải thuật tham lam – Greedy Algorithm

Bài tập 3: INTERVAL SCHEDULING

- Có n công việc, công việc j bắt đầu tại thời điểm s_j và kết thúc tại thời điểm f_j .
- Hai công việc được gọi là tương hợp nếu thời gian thực hiện chúng không giao nhau.
- Tìm một tập cực đại các công việc mà chúng tương hợp với nhau.

INPUT	OUTPUT
5	1 1 0 1 0
8 9 10 11 12	
8.5 11 11.5 12.5 13	

Design by Minh An

17

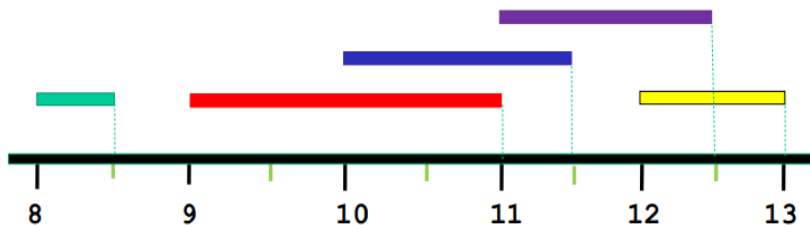
3. Giải thuật tham lam – Greedy Algorithm

Bài tập 3: INTERVAL SCHEDULING

INPUT	OUTPUT
5	1 1 0 1 0
8 9 10 11 12	
8.5 11 11.5 12.5 13	

Hai việc i và j không tương hợp nếu:

$f(i) \geq s(j)$ or $f(j) \geq s(i)$.



Design by Minh An

18

3. Giải thuật tham lam – Greedy Algorithm

Bài tập 3: INTERVAL SCHEDULING

`S[], F[]`: tập start time và finish time

`n`: số công việc

`schedule[]`: mảng kết quả

Duyệt danh sách công việc

Chọn công việc thứ `i` sao cho `F[i] → Min`

Thêm `i` vào `schedule[]`.

Xóa mọi công việc không tương hợp với `i`

return `schedule`

Design by Minh An

19

3. Giải thuật tham lam – Greedy Algorithm

Bài tập 3: INTERVAL SCHEDULING

Sắp xếp mảng đồng hành `S, F` theo chiều tăng dần của `F`

Khởi gán mảng `schedule` với các phần tử 0.

`last_finish = 0`

for `i = 1` to `n`:

if `s(i) >= last_finish`:

 Add `i` to `schedule`

`last_finish = f(i)`

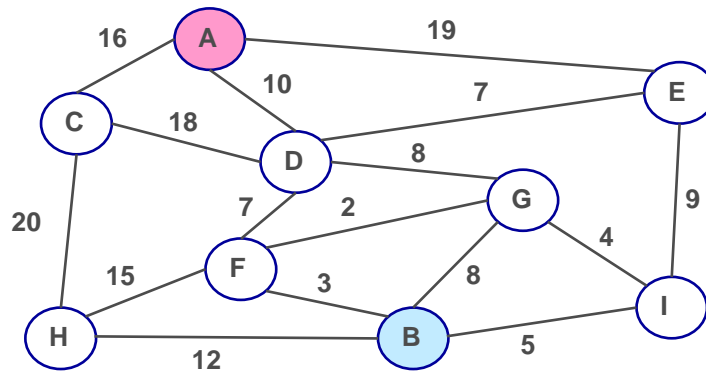
return `schedule`

Sử dụng cấu trúc dữ liệu phù hợp: `queue`

Design by Minh An

20

Bài tập 4



- Cho đồ thị vô hướng, có trọng số G như hình vẽ trên.
- Yêu cầu: Hãy tìm đường đi ngắn nhất từ đỉnh A đến đỉnh B (đường đi có tổng chi phí tốt nhất).

Design by Minh An

21

Greedy Traveling Sale Man

▪ Thuật giải

```
void Greedy(S, G) //S là tập ứng viên, G tập nghiệm
{
    G = ∅;
    while (S != ∅)
    {
        n = bestGet(S); //chọn phần tử tốt nhất trong S
        S = S - {n};
        if (G ∪ {n} là chấp nhận được)
            G = G ∪ {n} ;
    }
}
```

Design by Minh An

22

Bài tập 5

Một kho hàng gồm n gói hàng được ghi số thứ tự từ 1 đến n , gói hàng thứ i có kích thước s_i và khối lượng m_i ($1 \leq i \leq n$).

Ban đêm một tên trộm lẻn vào kho để lấy trộm hàng, tên trộm mang theo một chiếc ba lô có kích thước k . Vì sức yếu nên tên trộm chỉ chọn những gói hàng nhẹ nhàng.

Yêu cầu: Hãy chọn cho tên trộm những gói hàng ưng ý: tổng khối lượng các gói hàng được chọn là nhỏ nhất và tổng kích thước không vượt quá kích thước của ba lô.

Design by Minh An

23

Bài tập 6

Công ty vận tải A có n chiếc xe tải với các xe có tải trọng khác nhau, xe tải i có tải trọng là k_i (chở được tối đa k_i tấn). Công ty A cần vận chuyển m tấn hàng từ một kho đến địa điểm khác, hỏi công ty cần sử dụng bao nhiêu chiếc xe tải, gồm những xe nào (cho biết tải trọng của xe được chọn sử dụng) sao cho số xe tải cần sử dụng là ít nhất.

Cài đặt chương trình giải quyết bài toán (sử dụng chiến lược tham lam)

Design by Minh An

24

Bài tập 7

John có một chiếc xe tải với kích thước thùng xe là k (m^3). Công ty B có n kiện hàng khác nhau, kiện hàng k_i có khối lượng m_i (kg) và kích thước k_i (m^3). Công ty B thuê John chở một số kiện hàng trong số n kiện hàng. Hãy cho biết John xếp được bao nhiêu kiện hàng lên thùng xe, gồm những kiện hàng nào, sao cho tổng khối lượng xếp lên xe đạt được là lớn nhất?

Design by Minh An