

Giải đề QG 2007

Nguyễn Tấn Phát

Bài 1: Dãy con không giảm dài nhất (MAXISEQ)

- **Tóm tắt đề**

- Cho n và dãy A gồm n số: a_1, \dots, a_n ;
- Dãy u_k được định nghĩa: $u_1=1$;
 $u_k=u_{k-1}+k$;
- Tìm dãy con không giảm của dãy A
mà các phần tử thuộc dãy u_k ;

- **Ràng buộc**

- $n < 10^4$
- $a_i < 10^8$

MAXISEQ.INP	MAXISEQ.OUT
8	3
2	
2007	
6	
6	
15	
16	
3	
21	

Giải thích: 6 6 15 là dãy cần tìm

Ví dụ dãy $u_k = u_{k-1} + k$

- $u_1 = 1;$
- $u_2 = u_1 + 2 = 3;$
- $u_3 = u_2 + 3 = 6;$
- $u_4 = u_3 + 4 = 10;$
- $u_5 = u_4 + 5 = 15;$
- $u_6 = u_5 + 6 = 21$
-

$$\Rightarrow u_n = 1 + 2 + 3 + \dots + n = n(n+1)/2$$

Phân tích

- Vấn đề của bài toán là nếu kiểm tra được một số có thuộc u_k hay không? \Rightarrow Bài được giải quyết duyệt tuyến tính để tìm dãy dài nhất là dãy đơn điệu dài nhất;
- Kiểm tra một số có thuộc u_k có nhiều cách
- **Cách 1**: Đánh dấu ($u:\text{array}[0..\text{maxn}]$ of Boolean)

fillchar(u,sizeof(u),false);

$u[1]:=true;$

$i:=1; k:=1;$

while $k < \text{maxn}$ do begin

$\text{inc}(i); k:=k+i;$

 if $k < \text{maxn}$ then $u[k]:=true;$

end;

Phân tích

- Kiểm tra một số có thuộc u_k có nhiều cách
- Cách 2: (nếu $u_k > 10^8$)
 - Sinh dãy u_k ;
 - chắt nhị phân để kiểm tra một số có thuộc hay không
- Cách 3: Sử dụng công thức

$$\Rightarrow u_n = 1 + 2 + 3 + \dots + n = n(n+1)/2$$

Để tìm $a[i]$ có thuộc dãy u hay không?

Chắt nhị phân có tồn tại giá trị k để $a[i] = k(k+1)/2; \rightarrow \text{exit}(\text{true})$
ngược lại $\text{exit}(\text{false});$

Bài 2: Siêu thị may mắn(SMARKET)

- **Tóm tắt đề**
- Cho n mặt hàng; mỗi mặt hàng có giá trị c_i và có số lượng m_i ; Có số tiền S . Hỏi số cách có thể mua hàng sao cho tổng giá trị bằng S ; đưa ra một cách mua;

SMARKET.INP	SMARKET.OUT
12 3	2
4 1	0 2 0
6 2	
2 1	

Ràng buộc dữ liệu

$$1 \leq n \leq 500; 1 \leq s \leq 10^5; 1 \leq c_i \leq 10^4; 1 \leq m_i \leq 100$$

Phân tích

- Subtask1: Duyệt vét cạn như sau
- Mỗi món hàng có các khả năng: $j=0...m_i$; nghĩa là 0 chọn hoặc chọn 1 hoặc... hoặc chọn m_i ;
- Duyệt đệ qui cho tới khi $T=S$ thì đếm số cấu hình thỏa ĐK; (T là tổng hiện tại)
- **Đặt cận để giảm các nhánh thừa**

Xét bài toán con

- Cho dãy $A: a_1, a_2, \dots, a_n$ và S . Đếm số cách chọn các phần tử trong A để tổng bằng S .
- **Hướng dẫn**
- Gọi $f[i, j]$ là tổng số cách khi xét tới i và tổng bằng S
- CSQHĐ: $f[0, 0] = 1$; $f[0, i] = 0$ với mọi i
- Xét 2 TH:
 - Không chọn i : $t1 = f[i-1, j]$;
 - Chọn i : $t2 = f[i-1, j-a[i]]$;
 - $\Rightarrow F[i, j] = t1 + t2$;

Ví dụ: $S=6$; $n=5$: 4 6 2 1 5

i	1	2	3	4	5	6
1	0	0	0	1	0	0
2	0	0	0	1	0	1
3	0	1	0	1	0	2
4	1	1	1	1	1	2
5	1	1	1	1	2	3

Mở rộng bài toán xét mỗi vật có $m[i]$

- Xét vật i có các TH:
 - Không chọn i : $t1=f[i-1,j]$;
 - Chọn 1: $t2=f[i-1,j-a[i]]$;
 - Chọn 2: $t2= f[i-1,j-2*a[i]]$;
 -
 - Chọn $m[i]$: $t_k= f[i-1,j-m[i]*a[i]]$;
 - $\Rightarrow F[i,j]=t1+t2+...+t_k$;
 - Độ phức tạp: $O(n*s*m[i])$

- Subtask2: QHĐ <https://vietcodes.github.io/code/55/index.html>

Gọi f_i là số cách mua cho tổng số tiền là i , ban đầu $f_0 = 1$. Ta duyệt qua mỗi mặt hàng, cập nhật lại f như sau:

$$f'_i = f_i + f_{i-c} + f_{i-2c} + f_{i-3c} + \dots + f_{i-mc}$$

Với c, m là giá và số lượng của mặt hàng đang xét.

Đặt

$$\Delta_i = f'_i - f_i = f_{i-c} + f_{i-2c} + f_{i-3c} + \dots + f_{i-mc}$$

Ta thấy rằng $\Delta_i = f_{i-c} + \Delta_{i-c} - f_{i-(m+1)c}$, vì vậy ở mỗi bước ta có thể tính Δ trong $O(s)$ rồi cập nhật lại f trong $O(s)$.

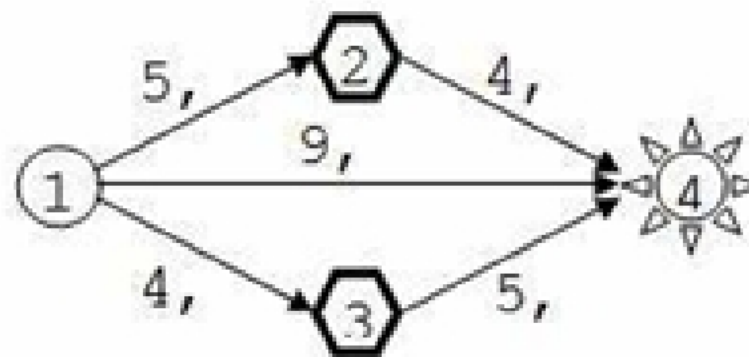
Độ phức tạp thuật toán: $O(ns)$.

Cần chú ý là kết quả có thể vượt giới hạn số 64-bit. Bản cài đặt ở dưới sử dụng số 128-bit để tính toán.

QBROBOT - VOI07 Robot cứu hỏa

- Tóm tắt

QBROBOT.INP	QBROBOT.OUT
4 0 1 1 0 5 1 2 5 4 1 3 4 3 1 4 9 4 2 4 4 1 3 4 5 2	3



Nút 2 và nút 3 có trạm tiếp năng lượng

QBROBOT - VOI07 Robot cứu hỏa

- Tóm tắt
- Cho n nút giao thông. m là số đường đi. Trên mỗi đường đi (i,j) mất thời gian $t[i,j]$ và chi phí xăng $c[i,j]$. Ban đầu robot chứa đầy xăng(w). Ở mỗi nút có thể có trạm đổ xăng robot có thể đổ đầy bình. Tìm w nhỏ nhất thỏa ĐK:
 1. Đường đi từ s đến t thời gian nhỏ nhất
 2. Robot đủ năng lượng để đi.

Thuật toán

- `dijkstra(1, d1);` //Xuất phát từ 1
- `dijkstra(n, dn);` //Xuất phát từ n
- Chặt nhị phân w;
- Với mỗi w \rightarrow `kt(w)` có thể đi được với đường đi ngắn nhất?
- Viết hàm `kt(w)` bằng BFS
 - Với mỗi u xác định v là kề với u khi thỏa ĐK:
 $(d1[u] + t[u,v] + dn[v] = d1[n])$ and $(w \geq c[u,v])$