Nguyễn Xuân Huy

**Greedy Method**

**Các chuyên đề chọn lọc**

Hà Nội - 2023

# Phương pháp tham

*Phương pháp tham* gợi ý chúng ta tìm một trật tự hợp lí để duyệt dữ liệu nhằm đạt được mục tiêu một cách chắc chắn và nhanh chóng. Thông thường, dữ liệu được duyệt theo một trong hai trật tự là tăng hoặc giảm dần theo một chỉ tiêu nào đó. Một số bài toán đòi hỏi những dạng thức cải biên của hai dạng nói trên.

### Packing

Australia

Anh David muốn xếp các bao hàng lên xe có tải trọng w. Có n bao hàng và biết trọng lượng mỗi bao. Anh muốn chở số lượng bao hàng ít nhất. Theo ý bạn, anh David nên làm thế nào?

Giới hạn: n < 100, w < 1000 kg.

|  |  |  |
| --- | --- | --- |
| PACKING.INP | Giải thích | OUTPUT |
| 200 10  12 40 30 25 37 10 22 17 14 14 | Tải trọng xe 200, 10 bao hàng  Trọng lượng mỗi bao hàng  Output  Tổng trọng lượng (199), số bao (8) | 199 8 |

#### Thuật toán

Bước 1. Đọc dữ liệu rồi sắp giảm. Độ phức tạp nlog(n)

Bước 2. Lấy lần lượt các bao hàng từ nặng trở xuống đến khi đủ thì dừng.

#### Chương trình

// Packing

#include <bits/stdc++.h>

using namespace std;

const char \* FN = "PACKING.INP";

void Print(int a[], int d, int c, const char \* msg = "") { ... }

void Packing() {

ifstream f(FN);

int w, n;

f >> w >> n;

cout << "\n Tai trong xe: " << w << " So bao hang: " << n;

int b[n]; // trong luong cac bao hang

for(int i = 0; i < n; ++i)

f >> b[i];

Print(b, 0, n-1, "\n Cac bao hang: ");

sort(b, b+n, greater<int>());

Print(b, 0, n-1, "\n Sap giam: ");

// Xep

int s = 0, t = 0, i;

for(i = 0; i < n; ++i) {

t += b[i];

if (t > w) { t -= b[i]; break; }

}

cout << "\n Result: ";

cout << "\n . Tong trong luong xe: " << t;

cout << "\n . So bao: " << i;

}

main () {

Packing(); // 199, 8

cout << "\n T h e E n d";

return 0;

}

## Luyện tập Phương pháp tham

### Music box

Người ta cần ghi N bài hát được mã số từ 1 đến N vào một băng nhạc có thời lượng tính theo phút đủ chứa toàn bộ các bài đã cho. Với mỗi bài hát ta biết thời lượng phát của bài đó. Băng sẽ được lắp vào một máy phát nhạc kiểu cổ đặt tại một phố đi bộ. Khách hàng muốn nghe bài hát nào chỉ việc nhấn phím ứng với bài đó. Để tìm và phát bài thứ i trên băng, máy xuất phát từ đầu cuộn băng, quay băng để bỏ qua i – 1 bài ghi trước bài đó. Tiếp đến, máy sẽ phát bài i rồi lại chuyển sang trạng thái đợi ở đầu cuộn băng. Thời gian quay băng bỏ qua mỗi bài và thời gian phát bài đó được tính là như nhau. Thời gian máy chuyển về đầu cuộn băng là không đáng kể. Tính trung bình, các bài hát trong một ngày được khách hàng lựa chọn với số lần (tần suất) như nhau. Hãy tìm cách ghi các bài trên băng sao cho tổng thời gian quay băng trong mỗi ngày là ít nhất.

Dữ liệu vào được ghi trong tệp văn bản tên MUSIC.INP.

Dòng đầu tiên là số tự nhiên N cho biết số lượng bài hát.

Tiếp đến là N số nguyên dương thể hiện dung lượng tính theo phút của mỗi bài. Mỗi đơn vị dữ liệu được ghi cách nhau qua dấu cách.

Ví dụ

|  |  |  |
| --- | --- | --- |
| MUSIC.INP  3  7 2 3 | MUSIC.OUT  2 2  3 5  1 12  19 | Có N = 3 bài hát:  Bài 1 phát trong thời gian 7 phút.  Bài 2 phát trong thời gian 2 phút.  Bài 3 phát trong thời gian 3 phút. |

*Dữ liệu ra được ghi trong tệp văn bản MUSIC.OUT theo dạng thức sau:*

* *N dòng đầu tiên thể hiện trật tự ghi bài hát trên băng: mỗi dòng gồm hai số nguyên dương j và d cách nhau, trong đó j là mã số của bài hát cần ghi, d là thời gian tìm và phát bài đó theo trật tự ghi này.*
* *Dòng thứ n + 1 ghi tổng số thời gian quay băng nếu mỗi bài hát được phát một lần trong ngày.*

*Với ví dụ trên, kết quả thu được sẽ như sau:*

* *Cần ghi lần lượt trên băng các bài theo trật tự : bài 2, bài 3, bài 1;*
* *Để tìm và phát bài 2 cần 2 phút;*
* *Để tìm và phát bài 3 cần 5 phút;*
* *Để tìm và phát bài 1 cần 12 phút;*
* *Tổng thời gian để tìm và phát mỗi bài một lần là: 19 phút.*

### Testing

Có N phần mềm (PM) mã số 1..N cần được kiểm định trên một máy tính, mỗi PM đòi hỏi đúng 1 giờ test. Với mỗi PM ta biết thời hạn phải nộp kết quả test sau khi hoàn thành kiểm định và tiền thưởng nếu nộp kết quả trước hoặc đúng thời điểm quy định. Chỉ có một máy tính trong tay, hãy lập lịch thực hiện đủ N test cho từng PM trên máy tính sao cho tổng số tiền thưởng thu được là lớn nhất và thời gian hoạt động của máy là nhỏ nhất. Giả thiết rằng máy được khởi động vào đầu ca, thời điểm t = 0 và chỉ tắt máy sau khi đã hoàn thành đủ N công việc.

Dữ liệu vào: tệp văn bản TEST.INP:

Dòng đầu tiên là số N.

N dòng tiếp theo: mỗi PM cần test được mô tả bằng hai số tự nhiên, số thứ nhất là thời hạn giao nộp kết quả kiểm định, số thứ hai là tiền thưởng. Các số cách nhau qua dấu cách.

Ví dụ

|  |  |  |
| --- | --- | --- |
| TEST.INP  4  1 15  3 10  5 100  1 27 | TEST.OUT  4  2  3  1  137 | PM 1 phải nộp kết quả test không muộn hơn thời điểm 1 (giờ)  với tiền thưởng 15 V (đơn vị tiền tệ);  PM 2 phải nộp kết quả test không muộn hơn thời điểm 3 (giờ)  với tiền thưởng 10 V;  PM 3 phải nộp kết quả test không muộn hơn thời điểm 5 (giờ)  với tiền thưởng 100 V;  PM 4 phải nộp kết quả test không muộn hơn thời điểm 1 (giờ)  với tiền thưởng 27. |

Dữ liệu ra: tệp văn bản TEST.OUT:

N dòng đầu tiên, dòng thứ t ghi một số tự nhiên i cho biết việc thứ i được làm trong giờ t.

Dòng cuối cùng ghi tổng số tiền thu được.

Ý nghĩa của dữ liệu ra:

Giờ thứ 1 test PM 4 và nộp đúng hạn nên được thưởng 27 V;

Giờ thứ 2 test PM 2 và nộp trước hạn nên được thưởng 10 V;

Giờ thứ 3 test PM 3 và nộp trước hạn nên được thưởng 100 V;

Giờ thứ 4 test PM 1 không có thưởng;

Tổng tiền thưởng thu được do đã hoàn thành đúng hạn ba PM 4, 2 và 3 là 27+10+100=137 (V).

### Knapsack

Có N vật (mặt hàng) mã số 1..N, với mỗi vật ta biết trọng lượng và giá trị của nó. Hãy xác định trọng lượng cần lấy ở một số vật để xếp vào một ba lô có sức chứa tối đa là M kg sao cho giá trị chứa trong ba lô là lớn nhất. Giả thiết là có thể lấy một số kg ở mỗi vật.

Dữ liệu vào: Tệp văn bản KNAPSACK.INP:

Dòng đầu tiên: hai giá trị nguyên dương N và M.

N dòng tiếp theo, mỗi dòng chứa hai giá trị nguyên dương d v cho mỗi vật, trong đó d là trọng lượng, v là giá trị tính theo một đơn vị trọng lượng của vật đó (đơn giá). Các số cách nhau qua dấu cách.

|  |  |  |
| --- | --- | --- |
| KNAPSACK.INP  5 30  8 5  5 4  4 2  3 8  16 6 | *N = 5 vật, sức chứa tối đa của ba lô là  M = 30 (kg).*  *Vật thứ nhất có trọng lượng 8, đơn giá 5*  *Vật thứ hai có trọng lượng 5, đơn giá 4*  *Vật thứ ba có trọng lượng 4, đơn giá 2,*  *Vật thứ tư có trọng lượng 3, đơn giá 8*  *Vật thứ năm có trọng lượng 16, đơn giá 6*  *Trọng lượng: kg*  *Đơn giá V / kg (V là đơn vị tiền)* | KNAPSACK.OUT    8  3  0  3  16  172 |

*Dữ liệu ra: Tệp văn bản tên KNAPSACK.OUT:*

* *N dòng, dòng thứ i cho biết trọng lượng cần lấy ở vật thứ i.*
* *Dòng cuối cùng ghi tổng giá trị thu được.*