



# OLYMPIC TIN HỌC SINH VIÊN LẦN THỨ XVIII, 2009

## Khối thi: Siêu cúp

Thời gian làm bài: 180 phút

Ngày thi: 08-10-2009

Nơi thi: ĐẠI HỌC NHA TRANG

Tên bài	Tên file chương trình	Tên file dữ liệu	Tên file kết quả	Hạn chế thời gian cho mỗi test
Tổng trung vị	MEDSUM.???	MEDSUM.INP	MEDSUM.OUT	2 giây
Vì rút cúm	VIRUS.???	VIRUS.INP	VIRUS.OUT	2 giây
Tuỳ chọn	OPTION.???	OPTION.INP	OPTION.OUT	2 giây

### Chú ý:

- Dấu ??? được thay thế bởi đuôi ngầm định của ngôn ngữ được sử dụng để cài đặt chương trình.
- Thí sinh phải nộp cả file mã nguồn của chương trình và file chương trình thực hiện (chương trình đã được biên dịch ra file .exe).

*Hãy lập trình giải các bài sau đây:*

### Bài 1. Tổng trung vị

Cho  $N$  dãy số không giảm  $\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_N$ , mỗi dãy gồm  $L$  số nguyên (dãy số được gọi là không giảm nếu mỗi phần tử đứng sau là lớn hơn hoặc bằng phần tử đứng trước). Xét hai dãy  $\mathcal{A}_i$  và  $\mathcal{A}_j$  ( $1 \leq i, j \leq N$ ), ta gọi dãy gộp (ký hiệu là  $\mathcal{A}_{ij}$ ) của hai dãy  $\mathcal{A}_i, \mathcal{A}_j$  là dãy gồm tất cả  $2L$  phần tử của hai dãy  $\mathcal{A}_i, \mathcal{A}_j$  được sắp xếp theo thứ tự không giảm và phần tử đứng ở vị trí thứ  $L$  trong dãy gộp được gọi là phần tử trung vị của nó.

**Ví dụ:** Xét hai dãy số

$$\mathcal{A}_i = (1 \ 3 \ 4 \ 5 \ 6); \mathcal{A}_j = (0 \ 1 \ 5 \ 6 \ 7).$$

Khi đó dãy gộp  $\mathcal{A}_{ij}$  từ hai dãy đã cho là

$$0 \ 1 \ 1 \ 3 \ 4 \ 5 \ 5 \ 6 \ 6 \ 7$$

có phần tử trung vị là 4.

**Yêu cầu:** Tính tổng của tất cả các phần tử trung vị của tất cả các dãy gộp  $\mathcal{A}_{ij}$  với  $1 \leq i < j \leq N$ .

**Dữ liệu:** Vào từ file văn bản MEDSUM.INP:

- Dòng đầu tiên chứa hai số  $N$  và  $L$  ( $2 \leq N \leq 200, 1 \leq L \leq 20000$ );
- Dòng thứ  $i$  trong số  $N$  dòng tiếp theo chứa  $L$  số nguyên là các phần tử của dãy thứ  $i$  trong số  $N$  dãy đã cho. Giả thiết là các phần tử của các dãy số là các số nguyên có trị tuyệt đối không vượt quá  $10^9$ .

Hai số liên tiếp trên cùng một dòng trong file dữ liệu được ghi cách nhau bởi ít nhất một dấu cách.

**Kết quả:** Ghi ra file văn bản MEDSUM.OUT giá trị  $\sigma \bmod 10^9$  (là phần dư trong phép chia  $\sigma$  cho  $10^9$ ), trong đó  $\sigma$  là tổng của tất cả các phần tử trung vị của tất cả các dãy gộp  $A_{ij}$  với  $1 \leq i < j \leq N$ .

**Lưu ý:** Có 50% số test thoả mãn:  $2 \leq N \leq 100$ ,  $1 \leq L \leq 300$ . Giải đúng các test này, thí sinh được ít nhất 50% số điểm tối đa cho toàn bộ bài toán.

**Ví dụ:**

MEDSUM . INP	MEDSUM . OUT
3 6	8
1 2 3 4 5 6	
3 4 5 6 7 8	
0 0 1 1 2 2	

## Bài 2. Vi rút cúm

Vi rút cúm đang lây lan trên toàn thế giới cũng như ở Việt Nam. Bản đồ gen của vi rút cúm đã được giải mã và được biểu diễn bởi một xâu kí tự  $S$  của bốn loại ARN: A, C, G, U.

Để sản xuất vắc-xin chống lại vi rút cúm, người ta nghiên cứu sử dụng  $k$  loại thuốc đánh số từ 1 đến  $k$ . Loại thuốc thứ  $i$  được biểu diễn bởi xâu kí tự  $D_i$  gồm các chữ cái từ tập  $\{A, C, G, U\}$ .

Ta nói loại thuốc thứ  $i$  áp dụng được lên virus cúm từ vị trí  $x$  đến vị trí  $y$ , nếu xâu  $D_i$  trùng với đoạn xâu con từ vị trí  $x$  đến vị trí  $y$  của xâu  $S$ . Khi áp dụng  $D_i$  lên  $S$  từ vị trí  $x$  đến vị trí  $y$ , đoạn xâu con từ vị trí  $x$  đến vị trí  $y$  của xâu  $S$  sẽ bị xóa đi, xâu  $S$  bị thu ngắn lại và độ dài giảm đi  $(y-x+1)$ . Như vậy thực hiện liên tiếp việc áp dụng các loại thuốc lên xâu  $S$  ta có thể rút ngắn độ dài của xâu  $S$ .

**Ví dụ:** Cho xâu  $S = \text{"GUGGACCA"}$  và 4 loại thuốc:  $D_1 = \text{"ACC"}$ ;  $D_2 = \text{"AA"}$ ;  $D_3 = \text{"CC"}$ ;  $D_4 = \text{"GUG"}$ . Khi đó ta có thể áp dụng các loại thuốc lên xâu  $S$  để rút ngắn độ dài của xâu này như sau:

$\text{GUGGACCA} (D_3) \rightarrow \text{GUGGAA} (D_2) \rightarrow \text{GUGG} (D_4) \rightarrow \text{G}.$

**Yêu cầu:** Hãy tìm cách áp dụng các loại thuốc lên vi rút cúm để độ dài xâu  $S$  còn lại là ngắn nhất. Lưu ý, một loại thuốc có thể không được sử dụng hoặc được sử dụng nhiều lần.

**Dữ liệu:** Vào từ file văn bản VIRUS.INP có cấu trúc như sau:

- Dòng đầu chứa xâu  $S$ , độ dài không vượt quá 1000.
- Dòng thứ 2 chứa số nguyên dương  $k$  ( $k \leq 20$ ).
- $k$  dòng tiếp theo, mỗi dòng chứa một xâu mô tả một loại thuốc, độ dài không vượt quá 1000.

**Kết quả:** Ghi ra file văn bản VIRUS.OUT một số nguyên dương duy nhất là độ dài nhỏ nhất của xâu  $S$  sau khi áp dụng các thuốc.

VIRUS . INP	VIRUS . OUT
GUGGACCA	1
4	
ACC	
AA	
CC	
GUG	

### Bài 3. Tuỳ chọn

Các hình thức khuyến mãi truyền thống đã phần nào trở thành nhàm chán, không thu hút khách hàng. Hãy tưởng tượng, ở nhà bạn đã có một ổ USB đủ các các loại, vậy mà khi mua một máy tính xách tay cực một Macbook trọng lượng 1250g với giá 30 triệu 500 ngàn đồng bạn được nhã nhận mời nhận khuyến mãi thêm một USB 4GB!

Siêu thị máy tính CMA (*Computer Machine for All – Máy tính cho tất cả mọi người*) đã đưa ra một phương thức khuyến mãi mới vừa lách được các qui định của luật khuyến mãi, vừa có sức thu hút lớn, đặc biệt là đối với giới trẻ sinh viên.

Nếu bạn mua một máy tính ở CMA giá từ 8 triệu 799 ngàn đồng trở lên, bạn sẽ được cấp một mã khóa P sử dụng một lần vạn năng và một số nguyên dương  $k$ . Bạn được quyền truy nhập vào trang WEB *CMA.Soft.com* của cửa hàng. Trang WEB này chứa  $n$  phần mềm, đánh số từ 1 đến  $n$ . Mỗi phần mềm được lưu trữ dưới dạng một file ZIP và được bảo mật bằng một khóa riêng. Khóa này vừa dùng để mở nén file vừa dùng để cài đặt phần mềm và đăng ký bản quyền sử dụng. Khóa thuộc loại sử dụng một lần: sau khi được dùng để mở file và cài đặt, khóa sẽ bị vô hiệu hóa. Trong một vài file ZIP còn chứa file DOC lưu khóa truy nhập file ZIP khác.

Thông tin trên trang WEB cho biết giá của mỗi phần mềm và khóa truy nhập của phần mềm này được giữ ở file ZIP nào. Bạn được quyền mở không quá  $k$  file ZIP, cài đặt phần mềm mở được và sử dụng khóa hoặc những khóa lưu trữ ở file này để truy nhập tới các file khác. Bạn không nhất thiết phải sử dụng hết các khóa nhận được. Ban đầu với khóa vạn năng P bạn có thể mở một file ZIP tùy chọn bất kỳ, cài đặt phần mềm đó vào máy của mình và dùng các khóa lưu trữ trong file này để truy nhập tới các file khác. Giá trị máy của bạn sẽ tăng thêm một lượng đúng bằng tổng giá trị phần mềm được cài đặt thêm. Nếu có cách lựa chọn sử dụng khóa đúng đắn, giá trị máy tính của bạn có thể tăng lên gấp đôi hay gấp ba!

Ví dụ, với  $n = 6$ ,  $k = 3$  và thông tin về các file ZIP như sau:

<i>File</i>	<i>Giá trị</i>	<i>Khóa truy nhập tới các file</i>
1	400	4
2	400	3 và 5
3	100	1
4	1000	
5	150	2
6	750	

Nếu dùng khóa vạn năng truy nhập vào file 2, bạn có thể cài đặt phần mềm 2, dùng khóa 3 nhận được để truy nhập và cài đặt phần mềm 3, sau đó dùng khóa 1 để truy nhập và cài đặt phần mềm 1. Tổng giá trị phần mềm cài đặt được là  $400+100+400 = 900$ . Nhưng nếu lúc đầu bạn truy nhập vào file 1, cài đặt và truy nhập tiếp đến file 4. Bạn chỉ cài được hai phần mềm, nhưng tổng giá trị của chúng sẽ là 1400. Có lẽ bạn sẽ chọn phương án sau, phải vậy không? Song đó vẫn chưa phải là cách có lợi nhất!

**Yêu cầu:** Cho  $n$ ,  $k$ , giá trị của từng phần mềm và khóa kèm theo tới các file khác (nếu có). Khóa truy nhập tới mỗi file được lưu giữ ở không quá một nơi. Hãy xác định tổng giá trị lớn nhất của các phần mềm bạn có thể cài đặt vào máy của mình.

**Dữ liệu:** Vào từ file văn bản OPTION.INP:

- Dòng đầu tiên chứa 2 số nguyên  $n$  và  $k$  ( $1 \leq k \leq n \leq 100$ ),
- Dòng thứ  $i$  trong  $n$  dòng sau chứa 2 số nguyên không âm  $v_i$  và  $m_i$ , trong đó  $v_i$  ( $v_i \leq 10^6$ ) – giá trị phần mềm thứ  $i$ ,  $m_i$  – số lượng khóa lưu trữ trong file thứ  $i$ . Nếu  $m_i > 0$  thì sau đó là  $m_i$  số nguyên dương khác nhau từng đôi một, mỗi số có giá trị không vượt quá  $n$  – là các chỉ số của các file có khóa truy nhập được lưu trong file thứ  $i$ .

Hai số liên tiếp trên cùng một dòng trong file dữ liệu được ghi cách nhau bởi ít nhất một dấu cách.

**Kết quả:** Ghi ra file văn bản OPTION.OUT một số nguyên – tổng giá trị lớn nhất của các phần mềm có thể cài đặt.

**Ví dụ:**

OPTION.INP	OPTION.OUT
6 3	1500
400 1 4	
400 2 3 5	
100 1 1	
1000 0	
150 1 2	
750 0	

----- HẾT -----