

Bài tập Greedy – Tuần 2

Bài 1 – All in All

Bạn nghĩ ra một kỹ thuật mã hoá mới để mã hoá thông điệp bằng cách chèn vào giữa các ký tự các ký tự được phát sinh ngẫu nhiên theo một cách thông minh. Bởi vì các vấn đề về bản quyền chúng ta sẽ không thảo luận chi tiết các chuỗi được phát sinh và chèn vào như thế nào trong thông điệp ban đầu. Tuy nhiên để kiểm chứng phương pháp này cần viết một chương trình kiểm tra xem thông điệp có thực sự được mã hoá trong chuỗi cuối cùng hay không.

Cho hai chuỗi s và t , bạn phải xác định xem s có là một chuỗi con của t , nghĩa là bạn có thể xoá các ký tự của t sao cho khi nối các ký tự còn lại sẽ ra s .

Dữ liệu

File dữ liệu **ALLALL.INP** chứa một số bộ test. Mỗi bộ được xác định bởi hai chuỗi ký tự chữ và số ASCII s , t cách nhau bởi khoảng trắng. Dữ liệu kết thúc bởi EOF.

Kết quả

Với mỗi dòng dữ liệu, cho biết s có phải là một chuỗi con của t hay không ('Yes' hay 'No'). Xuất tất cả kết quả ra file **ALLALL.OUT**.

Ví dụ

ALLALL.INP	ALLALL.OUT
sequence subsequence	Yes
person compression	No
VERDI vivaVittorioEmanueleReDiItalia	Yes
caseDoesMatter CaseDoesMatter	No

Bài 2 – Chuẩn bị thi Olympic SV

Để chuẩn bị cho kỳ thi Olympic Sinh viên, Ban huấn luyện đội tuyển Tin học trường đại học D giao cho mỗi thành viên đội tuyển n bài tập, các bài tập được đánh số từ 1 tới n . Thông thường, để giải được một bài tập sinh viên cần phải được trang bị một số kiến thức nào đó về thuật toán và cấu trúc dữ liệu và sau khi giải xong bài tập đó sinh

viên nhận thêm được một số kiến thức mới về hai lĩnh vực đó. Để giải bài tập thứ i sinh viên cần có chỉ số kiến thức tối thiểu về thuật toán và cấu trúc dữ liệu được đánh giá tương ứng bởi hai số nguyên không âm a_i , b_i và sau khi giải xong bài thứ i kiến thức về thuật toán và cấu trúc dữ liệu được tăng thêm một lượng c_i và d_i . Sinh viên Tuấn rất chăm chỉ trong quá trình tập huấn và rất mong muốn giải được càng nhiều bài tập càng tốt. Hiện tại Tuấn có chỉ số kiến thức về thuật toán là T và chỉ số kiến thức về cấu trúc dữ liệu P . Hãy tính số lượng nhiều nhất S các bài tập mà Tuấn có thể giải được.

Dữ liệu

Vào từ file văn bản **SVOL.INP** có $n+1$ dòng, trong đó dòng đầu chứa ba số n , T và P ($0 < n \leq 1000$; $0 \leq T, P \leq 10^6$). Dòng thứ i trong n dòng tiếp theo chứa bốn số nguyên không âm a_i , b_i , c_i và d_i ($0 \leq a_i, b_i, c_i, d_i \leq 10^6$). Các số trên cùng một dòng cách nhau bởi ít nhất một dấu cách.

Kết quả

Ghi ra file văn bản **SVOL.OUT** số lượng S các bài tập mà Tuấn giải được.

Ví dụ

SVOL.INP	SVOL.OUT	Ghi chú
5 1 3 2 1 1 0 1 0 1 0 1 4 2 2 5 4 3 3 2 3 1 2	5	Một phương án làm được cả 5 bài đó là lần lượt làm các bài: 2, 1, 5, 3 và 4.

Bài 3 – Ferry Loading II

Có một chiếc phà có thể chở n xe chạy ngang sông trong t phút và quay về trong t phút. m xe chạy đến bến sông theo một lịch cho trước. Thời gian sớm nhất mà tất cả các xe qua bên kia sông là bao nhiêu? Số lượng chuyến phà tối thiểu mà người lái phà cần phải chạy để chở hết tất cả các xe theo thời gian đó là bao nhiêu?

Dữ liệu

Vào từ file **FERRY.INP**. Dòng đầu tiên chứa n, t, m . m dòng tiếp theo mỗi dòng cho biết thời gian đến của một xe (theo phút tính từ bắt đầu của ngày). Nhân viên có thể chạy phà khi nào họ muốn, nhưng chỉ có thể chở những xe đã đến bến vào lúc đó.

Kết quả

Chương trình xuất ra kết quả vào file **FERRY.OUT** hai số nguyên trên một dòng cho biết: thời gian mà tất cả các xe được chở qua bên kia sông, tính theo phút tính từ bắt đầu của ngày, và số chuyến phà tối thiểu để chở các xe với thời gian đó.

Biết rằng $0 < n, t, m < 1440$. Thời gian đến được sắp theo thứ tự tăng dần.

Ví dụ

FERRY.INP	FERRY.OUT
2 10 10 0 10 20 30 40 50 60 70 80 90	100 5
2 10 3 10 30 40	50 2