



ĐỀ CHÍNH THỨC

(Đề thi gồm 04 trang)

TỔNG QUAN ĐỀ THI

Bài	Tên bài	File chương trình	Input/Output	Điểm
1	Mật khẩu	pw.cpp	Thiết bị chuẩn	100
2	Mạng công ty	comnet.cpp	Thiết bị chuẩn	100
3	Mê cung ngoặc	bmzgame.cpp	Thiết bị chuẩn	100
4	Bán hàng tối ưu	opbying.cpp	Thiết bị chuẩn	100

Bài 1. Mật khẩu

Alice muốn đặt mật khẩu cho một ứng dụng mà cô mới xây dựng. Cô đã chọn xâu kí tự S (kí hiệu $|S|$ là độ dài xâu S) và dự định chọn K đoạn trên xâu S (các đoạn gồm ít nhất một kí tự và không nhất thiết rời nhau) rồi ghép các đoạn theo một thứ tự nào đó để nhận được xâu đối xứng. Nhắc lại, xâu đối xứng là xâu đọc từ trái qua phải cũng như đọc từ phải qua trái, ví dụ “abba”, “sos” là xâu đối xứng, còn xâu “abab” thì không phải là xâu đối xứng. Alice đã chọn $K - 1$ đoạn, đoạn thứ i ($1 \leq i < K$) gồm các kí tự thứ L_i đến kí tự thứ R_i của xâu S ($1 \leq L_i \leq R_i \leq |S|$). Khi chọn đoạn thứ K , Alice muốn chọn một đoạn có độ dài m mà với đoạn đó Alice có thể ghép với $K - 1$ đoạn đã chọn theo một thứ tự nào đó để nhận được một xâu đối xứng.

Yêu cầu: Cho xâu S và $(K - 1)$ cặp số L_i, R_i , hãy đếm số cách chọn đoạn thỏa mãn.

Dữ liệu: Vào từ thiết bị vào chuẩn (bàn phím) có khuôn dạng:

- Dòng đầu chứa số nguyên dương K, m ($m \leq |S|$);
- Dòng thứ hai chứa xâu S chỉ gồm các kí tự ‘a’ đến ‘z’ ($2^K \times |S| \leq 2 \times 10^5$);
- Dòng thứ i ($1 \leq i \leq K - 1$) trong $K - 1$ dòng tiếp theo chứa hai số nguyên dương L_i, R_i ($1 \leq L_i \leq R_i \leq |S|$).

Kết quả: Ghi ra thiết bị ra chuẩn (màn hình) một dòng chứa một số là số lượng cách chọn thỏa mãn.

Input	Output
1 1 abab	4

Input	Output
2 2 abab 2 3	2

Subtask 1 (20 điểm): $K = 1; |S| \leq 2000$;

Subtask 2 (20 điểm): $K = 1$;

Subtask 3 (20 điểm): $K \leq 7; |S| \leq 2000$;

Subtask 4 (20 điểm): $K \leq 7$.

Subtask 5 (20 điểm): $K = 14$.

Bài 2. Mạng công ty

Công ty của Alice có n máy tính, các máy tính được đánh số từ 1 đến n . Hiện tại đang có m ($m \geq n - 1$) dây nối giữa các máy tính, dây nối thứ k ($1 \leq k \leq m$) nối hai máy tính u_k, v_k ($u_k \neq v_k$) và giúp truyền tin theo cả hai chiều giữa hai máy, có thể có nhiều dây nối giữa hai máy tính. Hiện tại, n máy tính có thể không liên thông với nhau, Alice có thể tháo dây nối để đầu nối lại với mong muốn làm cho n máy tính liên thông, cụ thể, cô có thể thực hiện:

- Tháo một đầu nối của dây thứ k để đầu nối sang máy tính khác, hành động này mất chi phí c_k ;
- Tháo cả hai đầu nối của dây thứ k để đầu nối sang hai máy tính khác, hành động này mất chi phí $2 \times c_k$.

Yêu cầu: Hãy giúp Alice tính chi phí ít nhất cần thực hiện để liên thông được n máy tính.

Dữ liệu: Vào từ thiết bị vào chuẩn (bàn phím) có khuôn dạng:

- Dòng đầu chứa hai số nguyên dương n, m ($n \leq 10^5; n - 1 \leq m \leq 2 \times 10^5$);
- Dòng thứ k ($1 \leq k \leq m$) trong m dòng tiếp theo chứa ba số nguyên dương u_k, v_k, c_k ($c_k \leq 10^6$).

Kết quả: Ghi ra thiết bị ra chuẩn (màn hình) một dòng chứa một số là chi phí ít nhất tìm được.

Input	Output
3 3 1 2 1 1 2 2 1 3 1	0

Input	Output
3 3 1 2 1 1 2 2 1 2 3	1

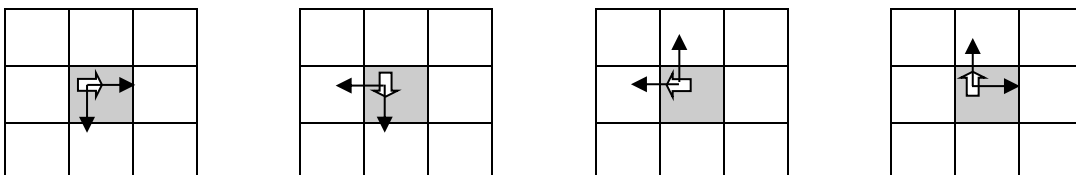
Subtask 1 (40 điểm): $c_i = 1$;

Subtask 2 (30 điểm): $m, n \leq 1000$;

Subtask 3 (30 điểm): Không có ràng buộc gì thêm.

Bài 3. Mê cung ngược

Một mê cung được mô tả bằng bảng chữ hình chữ nhật kích thước $m \times n$. Các hàng của bảng được đánh số từ 1 đến m , từ trên xuống dưới, các cột của bảng được đánh số từ 1 đến n , từ trái qua phải. Ô nằm trên giao của hàng i và cột j được gọi là ô (i, j) . Mỗi ô của lưới chứa một kí tự ngoặc mở '(' hoặc ngoặc đóng ')'



Người chơi sẽ xuất phát từ ô $(1, 1)$, quay hướng tới phía ô $(1, n)$ và di chuyển trên bảng. Việc di chuyển phải tuân thủ các quy tắc được mô tả trong hình trên, cụ thể: từ ô đang đứng, căn cứ vào hướng đang hướng tới được chỉ ra bởi mũi tên $\Rightarrow$, thực hiện bước di chuyển sang ô kề cạnh đang hướng tới, hoặc sang ô kề cạnh nằm bên phải (các hướng có thể di chuyển được chỉ ra bởi các mũi tên $\rightarrow$). Mỗi ô chỉ được đi qua nhiều nhất một lần. Người chơi có thể dừng di chuyển tại một ô nào đó để kết thúc trò chơi. Khi kết thúc trò chơi, người chơi

nhận được một xâu kí tự T gồm các kí tự trong các ô trên đường đi được xếp liên tiếp nhau. Người chơi giành chiến thắng nếu xâu T là một biểu thức ngoặc đúng bậc k .

Nhắc lại, định nghĩa biểu thức ngoặc đúng và bậc của biểu thức ngoặc.

- Biểu thức rỗng là biểu thức ngoặc đúng và có bậc bằng 0,
- Nếu A là biểu thức ngoặc đúng có bậc bằng k thì (A) cũng là một biểu thức ngoặc đúng có bậc bằng $k + 1$,
- Nếu A và B là hai biểu thức ngoặc đúng và có bậc tương ứng là k_1 và k_2 thì AB cũng là một biểu thức ngoặc đúng có bậc bằng $\max(k_1, k_2)$.

Ví dụ, $'()()'$ là một biểu thức ngoặc đúng có bậc bằng 2 còn $'(())()'$ là một biểu thức ngoặc đúng và có bậc bằng 3.

Yêu cầu: Cho bảng chữ và số nguyên dương k , đếm số lượng đường đi khác nhau giúp người chơi giành chiến thắng. Hai đường đi được gọi là khác nhau nếu tồn tại một ô thuộc đường đi này nhưng không thuộc đường đi kia.

Dữ liệu: Vào từ thiết bị vào chuẩn (bàn phím) có khuôn dạng:

- Dòng đầu tiên ghi ba số nguyên dương m, n, k ($m, n \leq 30; k \leq 10$);
- Tiếp theo là m dòng mô tả bảng chữ, mỗi dòng chứa một xâu gồm n kí tự, mỗi kí tự ngoặc mở '(' hoặc ngoặc đóng ')'

Kết quả: Ghi ra thiết bị ra chuẩn (màn hình) một dòng là số lượng đường đi đếm được chia dư cho $(10^9 + 7)$.

Input	Output
3 3 1 ()) ())	4

Subtask 1 (30 điểm): $m, n \leq 5$;

Subtask 2 (30 điểm): $k = 1$;

Subtask 3 (40 điểm): Không có ràng buộc gì thêm.

Bài 4. Bán hàng tối ưu

Alice có n món đồ cô và muốn bán tất cả trong m ngày. Cô đã tìm hiểu và biết được c_{it} là số tiền mà cô sẽ nhận được nếu bán món đồ i ($1 \leq i \leq n$) ở ngày t ($1 \leq t \leq m$) hoặc $c_{it} = -1$ nếu không ai muốn mua món đồ i ở ngày t . Một số món đồ phải được bán trước một số món đồ khác, có k ràng buộc như vậy, mỗi ràng buộc có dạng (u, v) và có nghĩa là món đồ u phải được bán trước ít nhất một ngày so với ngày bán món đồ v . Alice muốn lên kế hoạch bán tối ưu để có thể thu được nhiều tiền nhất. Chú ý rằng, mỗi ngày Alice có thể bán nhiều hơn một món đồ miễn là các món đồ liên quan đến ràng buộc bán trước các món đồ này đều đã được bán ở các ngày trước.

Yêu cầu: Giúp Alice tính số tiền nhiều nhất có thể nhận được khi bán tối ưu tất cả các món đồ.

Dữ liệu: Vào từ thiết bị vào chuẩn (bàn phím) có khuôn dạng:

- Dòng đầu tiên ghi ba số nguyên dương n, m, k ($n, m, k \leq 100$);

- Tiếp theo là n dòng, dòng thứ i ($1 \leq i \leq n$) chứa m số nguyên $c_{i1}, c_{i2}, \dots, c_{im}$ ($-1 \leq c_{it} \leq 100$);
- Tiếp theo là k dòng, mỗi dòng chứa hai số nguyên dương u, v ($1 \leq u, v \leq n$) cho biết món đồ u phải trước ít nhất một ngày so với ngày bán món đồ v .

Dữ liệu đảm bảo có cách bán hết cả n món đồ.

Kết quả: Ghi ra thiết bị ra chuẩn (màn hình) một dòng là số tiền nhiều nhất có thể nhận được khi bán tối ưu tất cả các món đồ.

Input	Output
3 2 2	270
100 -1	
100 80	
100 90	
1 2	
1 3	

Subtask 1 (25 điểm): $n \leq 15$;

Subtask 2 (35 điểm): Trong k ràng buộc không có hai ràng buộc nào có cùng giá trị v ;

Subtask 3 (40 điểm): Không có ràng buộc gì thêm.

----- **HẾT** -----

Họ và tên thí sinh: Số báo danh:

Lưu ý:

- Thí sinh **không** được sử dụng tài liệu.
- Cán bộ coi thi **không** giải thích gì thêm.