Bài fibsseq.

Ta cho f (u, v) là độ dài dãy fibonacci mà phần tử chót là phần tử a[v], phần tử áp chót là phần tử a[u] (u<v).

Với mỗi f (i, j) (i, j chạy từ 1->n) ta QHĐ bằng công thức f(i, j)=f(k, i)+1 với a[k] = a[j] – a[i] với k max & k<i

Đáp án cần tìm là f (i,j) max và để có dãy ta truy vết f(i, j) (dùng mảng đánh dấu).

Độ phức tạp O (n^2\*log­­­2 (n))

1. Bài Subpair

Ta xét tất cả vị trí ngắt x và y của 2 tập ban đầu dãy ban đầu a, b thành 4 dãy con a1, a2, b1, b2 rồi lần lượt tìm dãy con chung liên tiếp dài nhất giữa dãy a1 và b1 ( giả sử là c1 độ dài d(c1) ), giữa a2 và b2 (c2, d(c2) ). Ta chọn x, y sao d1+d2 dài nhất. Xuất c1 và c2.

VD Test đề bài

abacabadabacaba

acabacadacabaca

Ta lấy x=7, y=5 thì ta có 4 dãy con a1 abacaba a2 dabacaba b1 acaba b2 cadacabaca. Rồi ta tìm dãy con chung dài nhất a1 và b1 là acaba

Và a2, b2 là abaca. d1 + d2 max = 10.

CT QHĐ Tìm dãy con chung dài nhất giữa 2 dãy

F(i, j) là dãy con chung liền kề dài nhất của 2 dãy, ký tự cuối dãy 1 là ai, ký tự cuối dãy 2 là bj.

* F(i,j) = max(F(i-1 ,j), F(i, j-1), F(i-1,j-1)+1(nếu ai=bj) )

Độ phức tạp

1. Bài Longdom

Dễ thấy số dòng hoặc số cột ít nhất 1 trong 2 số <=9. Nếu số dòng > số cột thì ta xoay bảng.

Ta dùng QHĐ trạng thái.

+ Đối với domino 2x1

F(n,s,s’) là bảng đã được xếp đômino phủ kín n cột, ta cần xét 2 cột n+1 & n+2 của bảng có trạng thái s & s’.

Ta xét cột n+1, khi ta thấy 1 ô trống nào ta xét 2 trường hợp là nhét ô domino theo chiều ngang hoặc domino theo chiều dọc đè lên ô đấy rồi cứ thế QHĐ. Nếu dòng n+1 đã được phủ kín ta xét bảng đã phủ kín n+1 dòng và xét dòng n+2 & n+3.

Cụ thể F(i,s,s’) = F(i,s’,s’’) nếu dòng i+1 đã phủ kín

= F(i,s1,s’1) + F(I,s2,s’2) (các s & s’ mới là trạng thái mới khi phủ them 1 domino theo chiều dọc hoặc ngang.

+ Đối với domino 3x1

Ta làm tương tự nhưng thay vì xét 2 cột tiếp theo của bảng ta xét 3 cột tiếp theo.

1. Bài orderenc

Ta nhận thấy khi mã hóa các ký tự theo dãy nhị phân theo từ điển, tại một dãy ký tự ban đầu ta có thể ngắt thành 2 dãy con, dãy con phía trước đoạn ngắt khi mã hóa ký tự ban đầu là 0, tập con phía sau đoạn ngắt ký tự là 1. Rồi từ các tập con đó lại tiếp tục phân chia. Cứ như thế khi nào tập đó còn 1 ký tự thì đó là 1 ký tự mã hóa. Bài toán trở thành tìm cách ngắt để có được các mã hóa cho các ký tự sao cho tổng mã hóa của dãy ký tự là bé nhất.

VD dãy ký tự có 5 ký tự A, 7 B, 13 C, 4 D, 2 E, 7F.

Dãy ban đầu gộp tất cả các ký tự con là ABCDEF.

Đây là 1 cách ngắt để có mã hóa ký tự

(rỗng) ABCDEF (ngắt C)->Nhánh 0: ABC (ngắt B)->Nhánh 0: AB(ngắt A)->Nhánh 0: A. Mã 000 (3)

->Nhánh 1: B. Mã 001(3)

->Nhánh 1:C -> Mã 01(2)

->Nhánh 1: DEF(ngắt D)-> Nhánh 0: D -> Mã 10(2)

-> Nhánh 1:EF(ngắt E) -> Nhánh 0: E. Mã 110(3)

-> Nhánh 1: F. Mã 111(3)

Tổng độ dài mã = 5\*3+7\*3+13\*2+4\*2+2\*3+7\*3.

Nhận thấy với mỗi dãy con khi phân tách độ dài mã sẽ tăng lên một cho mỗi ký tự -> độ dài mã chung sẽ tăng lên là tổng số lần xuất hiện của các ký tự. Như trong ví dụ khi ngắt DEF thì Mã của D,E,F đều tăng 1 -> Độ dại đoạn mã tăng lên 4 (ký tự D) + 2 (ký tự E) + 7 (ký tự F).

* Công thức QHĐ

F(c) là độ dài tăng thêm từ đoạn c sau khi mã hóa

If(số ký tự trong dãy là 1) return 0.

Else return min(F(dãy con 1) + F(dãy con 2) + mỗi ký tự trong dãy\*số lần xuất hiện ký tự đó) với mỗi cách ngắt dãy