SOLUTION 26/12 - C

Bài 1. NUMBER

Subtask 1: $n, q \leq 1000$: Sử dụng thuật toán DFS/BFS, khi xác định được quan hệ cha con $u - v$ thì $d[v] = 10.d[u] + w$ với w là trọng số cạnh $u - v$. Độ phức tạp $O(n.q)$;

Subtask 2: với mọi truy vấn $u - v$ thì đỉnh v luôn thuộc đường đi từ u tới đỉnh 1: Sử dụng thuật toán Binary Jumping (bảng thưa) với u đi lên vì v là tổ tiên của u. Gọi $T[u][i]$ là tổ tiên đời 2^i, $A[u][i]$ là số tạo thành khi đi từ u lên 2^i cạnh, công thức bước nhảy là $ans = ans * 10^{2^i} + A[u][i]$. Độ phức tạp $O(n + q)\log(n)$;

Subtask 3: $n, q \leq 100000$: Sử dụng thuật toán Binary Jumping từ hai phía u, v. Tuy nhiên với phía v cần duy trì biến chứa số chữ số của đường đi, phía u thì không cần vì là lũy thừa của 2. Gọi $ans1, ans2$ lần lượt là số tạo thành khi nhảy từ u, v lên $LCA(u, v)$, công thức tính đáp số là $ans = ans1 * 10^{scs} + ans2$. Độ phức tạp $O(n + q)\log(n)$.

Bài 2. SHOPPING

Subtask 1: giới hạn $b = 0$ nên mỗi món đồ không cần quan tâm tới z. Đây là bài toán quy hoạch động cái túi cơ bản: gọi $dp[i][j]$ là tổng giá trị lớn nhất nếu xét trong i món đồ đầu và số tiền không quá j. Công thức quy hoạch động tính $dp[i][j]$ là max của hai trường hợp:

- $dp[i-1][j]$: không mua món đồ thứ i
- $dp[i-1]\big[j - y[i]\big] + x[i]$: mua món đồ thứ i

Đáp số là $dp[n][a]$. Độ phức tạp $O(n.a)$;

Subtask 2: giới hạn $n \leq 5$ nên có thể duyệt tất cả các tập con của tập n món đồ, với mỗi tập con, voucher sử dụng tối ưu là sử dụng vào món đồ có z nhỏ. Độ phức tạp $O(2^n)$;

Subtask 3: $n, a, b \leq 50$: Thuật toán quy hoạch động: gọi $dp[i][j][k]$ là tổng giá trị lớn nhất nếu xét trong i món đồ đầu tiên, sử dụng j đồng và sử dụng k voucher. Công thức quy hoạch động tính $dp[i][j][k]$ là max của các trường hợp:

- $dp[i-1][j][k]$: không mua món đồ thứ i
- $dp[i-1][j - (y[i] - t/z[i])][k - t]$ $\forall t = 0,1, \dots, \max(k, y[i] * z[i])$: mua món đồ thứ i bằng cách sử dụng t voucher.

Đáp số là $dp[n][a][b]$. Độ phức tạp $O(n.a.b^2)$;

Subtask 4: $n, a, b \leq 200$: Nhận xét rằng khi biết tập các món đồ được mua thì sử dụng voucher tối ưu nhất là sử dụng vào món đồ có z nhỏ nhất. Do đó sắp xếp các món đồ theo thứ tự món đồ nào có z_i lớn thì đứng trước. Khi đó, khi xét tới món đồ thứ i, nó là món đồ có z_i nhỏ nhất, nên dùng tối đa voucher vào đây nên $t = \max(k/z[i] * z[i], y[i] * z[i])$.

Đáp số là $dp[n][a][b]$. Độ phức tạp $O(n.a.b)$.

Bài 3. GEN

Subtask 1: $n_1, m_1, n_2, m_2 \leq 200$: sử dụng thuật toán BFS hoặc Dijkstra với đồ thị tích. Độ phức tạp là $O(n_1 * n_2 + m_1 * m_2)$;

Subtask 2: Trên mỗi gen, nếu một trạng có thể đi tới vào thời điểm x, thì có thể đi tới vào các thời điểm $x + 2, x + 4$, ... bằng cách sử lặp lại 1 cạnh nào đó. Để đạt được một trạng thái ngoại hình nào đó có hai cách là đi tới vào thời điểm chẵn hoặc đi tới vào thời điểm lẻ. Sử dụng thuật toán Dijkstra trên 2 đồ thị ban đầu để tính các thông tin: $d[u][0]$ /$d[u][1]$ là thời điểm chẵn/ lẻ nhỏ nhất để tới u trên đồ thị thứ nhất, $g[u][0]$ /$g[u][1]$ là thời điểm chẵn/ lẻ nhỏ nhất để tới u trên đồ thị thứ hai. Với mỗi trạng thái (x, y) thời gian để đến vào thời điểm chẵn là $a = \max(d[x][0], g[y][0])$, thời gian để đến vào thời điểm lẻ là $b = \max(d[x][1], g[y][1])$, thời gian tối thiểu là $\min(a, b)$. Độ phức tạp $O\big((n_1 + n_2 + m_1 + m_2)\log(n_1 + n_2) + n_1 * n_2\big)$;

Subtask 3: Tính sum_{chan} là tổng thời gian tối thiểu để tới các trạng thái vào thời điểm chẵn, sum_{le} là tổng thời gian tối thiểu để tới các trạng thái vào thời điểm lẻ. Tuy nhiên một trạng thái có

thể đến cả vào thời điểm chẵn lẫn lẻ, nên tính thêm sum_{max} là tổng thời gian để tới các trạng thái không kể thời điểm chẵn lẫn lẻ. Đáp số là $sum_{chan} + sum_{le} - sum_{max}$;

Công thức tính tổng $sum_{chan} = \sum_{i\ chẵn}(cnt1[i] * cnt2[i] - cnt1[i-2] * cnt2[i-2]) * i$ trong đó $cnt1[i]$ là số đỉnh trên đồ thị thứ nhất có thời gian tối thiểu là chẵn và không quá i, $cnt2[i]$ là số đỉnh trên đồ thị thứ hai có thời gian tối thiểu là chẵn và không quá i. Tương tự với hai tổng còn lại. Độ phức tạp $O(n_1 + m_1 + n_2 + m_2)$.

----------Hết----------