SỞ GIÁO DỤC VÀ ĐÀO TẠO
HÀ NỘI

KÌ THI CHỌN HỌC SINH GIỎI LỚP 9 CẤP THÀNH PHỐ
NĂM HỌC 2022 – 2023

ĐÁP ÁN

Môn thi: TIN HỌC
Ngày thi: *08 tháng 1 năm 2023*
Thời gian làm bài: *150 phút*

Hướng dẫn thuật toán

Bài I. Thời gian *(5,0 điểm)*

Nếu X < 60, chỉ cần in ra thời gian là 08:X. Nếu $60 \le X \le 100$ thì kết quả sẽ là 09:(X%60).

Bài II. Mật mã *(5,0 điểm)*

Duyệt xâu, tạo ra các xâu con chứa các chữ số có thể lưu vào mảng hoặc set để đếm số phần tử khác nhau.

Bài III. Trạm phát sóng *(4,0 điểm)*

Dùng thuật toán tìm kiếm nhị phân.

- Sắp xếp dãy vị trí N trạm thu sóng theo thứ tự tăng dần.
- Thực hiện tìm kiếm nhị phân giá trị khoảng cách i để tính số trạm phát sóng cần thiết nếu khoảng cách tối đa cho phép là i.

Bài IV. Triển lãm *(3,0 điểm)*

- Sub 1: Duyệt trâu toàn bộ cách chọn trong 2^{16}.
- Sub 2: Sort các bức tranh theo kích thước, có thể thấy rằng nếu chọn 2 tác phẩm i và j ($j \ge i$) thì ta có thể chọn toàn bộ bức tranh từ i->j mà không thay đổi A_max và A_min. Ta có thể duyệt trâu toàn bộ các cặp i,j rồi tính giá trị các tác phẩm từ i đến j. Độ phức tạp sẽ là $O(N^3)$.
- Sub 3: Có thể giảm độ phức tạp xuống $O(N^2)$ bằng cách dùng tổng cộng dồn giá trị các bức tranh.
- Sub 4: Nhận ra giá trị trưng bày nếu chọn các bức tranh từ L -> R sẽ có công thức là (S[R] - A[R]) - (S[L - 1] - A[L - 1]) với S là mảng cộng dồn. Vì vậy thay vì phải duyệt N^2 các cặp, ta có thể duyệt bức tranh có A_max và duy trì một giá trị để lưu lại min(S[i] - A[i]) trước đó với độ phức tạp O(N). Độ phức tạp sẽ là O(NLogN) của sort.

Bài V. Dãy đẹp *(3,0 điểm)*

- Sub 1: Duyệt trâu toàn bộ các dãy con.
- Sub 2: Do dãy phân biệt, gọi mx là phần tử lớn nhất và mn là phần tử nhỏ nhất, dãy độ dài N đẹp khi mx – mn + 1 = N, độ phức tạp là $O(n^2)$.
- Sub 3: Dùng chia để trị, f(l, r) là số lượng số dãy con đẹp có 2 đầu trong đoạn l, r

  m = (l + r) / 2

  Cần tính số lượng dãy đẹp (i, j) mà i <= m và m < j.

  Gọi L là phần tử nhỏ nhất của dãy cần tìm, R là phần tử lớn nhất.

  Có 4 trường hợp:

  - L, R nằm cùng phía [l, m] (1)
  - L, R nằm cùng phía [m + 1, r] (2)
  - L nằm ở [l, m] và R nằm ở [m + 1, r] (3)
  - L nằm ở [m + 1, r] và R nằm ở [l, m] (4)

Lặp qua đầu bên phải từ m + 1 đến r. Bây giờ ta chỉ cần quan tâm đến trường hợp (2) và (4) sau đó ta có thể làm tương tự cho phần bên trái.

Ta đang ở vị trí m + 1<= i <= r, đặt cmin = min(p[m + 1…i]), cmax = max(p[m + 1…i]).

Đối với trường hợp 2 dễ dàng nhận thấy j = m + 1 - (R - L + 1) - (i - m), để kiểm tra xem p[j…i] có phải là dãy đẹp hay không thì tìm max, min trên một đoạn, hoặc dung hash….

Đối với trường hợp 4, min(p[j…m]) không thể < cmin. Vì min(p[m + 1…i + 1]) <= cmin -> $j_{i + 1}$ <= j_i trong đó j_i là vị trí nhỏ nhất có thể của đầu bên trái khi đầu bên phải là i.

Ta có phương trình: R - L = i - j -> R + j = i + L. Ta đã biết i và L nên tính mọi vị trí k sao cho j <= k và R = max(p[k…m]) > cmax và k + R = i + L. (***)

- Sub 4: Gán một số nguyên bất kì cho mỗi giá trị từ 1 đến N là h(i)

  H(l…r) là h(l) + h(l + 1) + … + h(r)

  P(l…r) là h(a[l]) + h(a[r])+…+h(a[r])

  Dãy con a[l..r] đẹp khi H(L…R) = P(l…r)

  Làm giống hệt sub 3, ở (***) ta có H(L…R) = P(l…r) biến đổi tương tự.

---------------**HẾT**---------------