

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



Bài tập lớn 1 - Phần 2

HỆ THỐNG XE BUS TẠI CÁC THÀNH PHỐ LỚN

TP. HỒ CHÍ MINH, THÁNG 11/2020

ĐẶC TẢ BÀI TẬP LỚN 1 (Phần 2)

Phiên bản 1.0

1 Chuẩn đầu ra

Sau khi hoàn thành bài tập lớn này, sinh viên sẽ có khả năng:

- Ứng dụng một cấu trúc dữ liệu đã biết vào xử lý bài toán thực tế.
- Phân tích độ phức tạp của giải thuật đã hiện thực dưới góc độ thực tế thí nghiệm.

2 Dẫn nhập

Ở các thành phố lớn, giao thông công cộng là một đối tượng quan trọng trong tổng hợp cơ sở hạ tầng của thành phố. Giao thông công cộng là một trong những phương thức di chuyển của người hiện đại tại các thành phố lớn, đặc biệt là xe bus.

Hệ thống giao thông bằng xe bus có nhiều tuyến. Mỗi tuyến sẽ có được đánh bằng một mã số cố định, bến xuất phát (bến đầu), bến kết thúc (bến cuối) và thời gian di chuyển để di chuyển trên toàn tuyến. Hằng ngày, mỗi tuyến sẽ có nhiều chuyến xe thực hiện đi từ bến đầu đến bến cuối. Để phân biệt giữa các chuyến của cùng một tuyến, người ta sử dụng biển kiểm soát của phương tiện vận chuyển và thời điểm xuất bến.

3 Nhiệm vụ

Sinh viên được yêu cầu xây một chương trình trên C++ để hiện thực việc lưu trữ thông tin về các tuyến xe bus này với giả định rằng, số lượng chuyến tối đa của một chuyến trên một ngày là bằng nhau, ký hiệu là N . Tất cả các lệnh xử lý dữ liệu trên chương trình này có độ phức tạp là $O(N)$.

Sinh viên có thể hiện thực lại một cấu trúc dữ liệu khác phù hợp, nhưng được khuyến khích sử dụng cấu trúc **FragmentLinkedList** đã được hiện thực ở phần 1 để lưu trữ cho chương trình này.

4 Lệnh xử lý dữ liệu

4.1 Hướng dẫn chung

Mỗi lệnh xử lý dữ liệu là chuỗi các ký tự. Mỗi lệnh bắt đầu bằng một từ khoá (từ in đậm trong mô tả) và theo sau là các thông số (từ đặt trong dấu < và > trong mô tả). Giữa lệnh và các thông số cách nhau đúng một khoảng trắng. Không có khoảng trắng nào trước từ khoá lệnh và cũng không có khoảng trắng nào đi sau thông số cuối cùng. Một số thông số cuối có thể tùy chọn (có hoặc không có trong lệnh), các thông số này sẽ được đặt trong dấu [và] trong mô tả. Khi một lệnh không cung cấp đúng số thông số hoặc không có kiểu thông số đúng như mô tả hoặc có các khoảng trắng không như mô tả thì lệnh sẽ không được xử lý và kết quả trả về là chuỗi ký tự "-1". Ngược lại, lệnh sẽ được xử lý và một chuỗi ký tự sẽ được trả về theo mô tả trong mục 4.2 (nếu được mô tả ở dạng số thì chuyển về kiểu chuỗi để trả về).

Ý nghĩa của từ viết tắt và kiểu dữ liệu của các thông số được mô tả như sau:

- CODE: một chuỗi ký tự thể hiện mã tuyến.
- LP: một chuỗi ký tự lưu trữ giá trị của biến kiểm soát của chuyến xe.
- CASE: một giá trị nhị phân, nếu 0 tức là tuyến xe đi từ bến đầu đến bến cuối (chiều đi), ngược lại tức là chuyến xe đi theo chiều ngược lại (chiều về).
- TIME: một số nguyên (biểu diễn thời gian theo chuẩn ISO) thể hiện thời điểm.
- N: số chuyển tối đa cho tất cả các tuyến.

Trong các lệnh có hai thông số thời gian <TIME_A> và <TIME_B> tùy chọn (biểu diễn [<TIME_A> [<TIME_B>]]), chương trình sẽ thực thi lệnh với các thông số khác và

- <TIME_A> ≤ <TIME> ≤ <TIME_B> nếu có cả <TIME_A> và <TIME_B> trong lệnh.
- <TIME> = <TIME_A> nếu chỉ có <TIME_A> trong lệnh.
- mọi <TIME> nếu không có thông số <TIME_A> và <TIME_B>.

Mỗi lệnh cần phải được xử lý với độ phức tạp về tập thời gian không vượt quá qui định ở cột "Độ phức tạp". Sau khi xử lý xong tất cả các lệnh và trả về giá trị tương ứng với từng lệnh, chương trình phải đảm bảo huỷ tất cả các đối tượng dữ liệu được cấp phát động, không để lại rác trong bộ nhớ trước khi kết thúc chương trình.

4.2 Danh sách lệnh

Yêu cầu	Độ phức tạp	Mô tả
SQ <N>	O(1)	Thiết lập số chuyến xe tối đa cho tất cả các tuyến. Lệnh này chỉ thực hiện đúng một lần duy nhất trong chương trình và thực hiện đầu tiên. Nếu đúng như mô tả trả về 1, ngược lại trả về -1.
INS <CODE> <LP> [<CASE>] <TIME_A> <TIME_B>	O(N)	Thêm một chuyến xe bus có biển kiểm soát <LP> thuộc tuyến có mã số <CODE> với thời điểm xuất bến đầu là <TIME_A>, đến bến cuối là <TIME_B>. Lệnh xử lý thành công khi không có chuyến xe bus cùng tuyến và giờ xuất bến. Trong trường hợp biển kiểm soát đã tồn tại trước đó, thời điểm xuất phát <TIME_A> phải có lớn hơn thời điểm đến bến trước đó. Khi lệnh giao dịch thành công, trả về số xe đã lưu trữ của tuyến, ngược lại trả về -1. Giả định rằng, một biển kiểm soát chỉ thực hiện chuyến cho một tuyến cố định.
DEL <CODE> [<TIME_A> [<TIME_B>]]	O(N)	Xoá các chuyến xe bus đã lưu trữ của tuyến <CODE> có thời điểm xuất bến nằm trong đoạn <TIME_A> và <TIME_B>. Lệnh trả về số chuyến bị xóa ra khỏi tập dữ liệu đã lưu trữ.
CS <CODE> <TIME> [<CASE>]	O(N)	Trả về số chuyến đã bắt đầu đi nhưng chưa đến bến của tuyến bus có mã <CODE> tại thời điểm xem xét là <TIME>. Khi có giá trị <CASE>, ta cần trả về đúng số chuyến tương ứng với chiều đi hoặc chiều về, ngược lại, ta trả về tất cả chuyến thỏa mãn ở cả hai chiều.
CE <CODE> <TIME> [<CASE>]	O(N)	Trả về số chuyến đã kết thúc của tuyến bus có mã <CODE> tại thời điểm xem xét là <TIME>. Khi có giá trị <CASE>, ta cần trả về đúng số chuyến tương ứng với chiều đi hoặc chiều về, ngược lại, ta trả về tất cả chuyến thỏa mãn ở cả hai chiều.

GS <CODE> <TIME> [<CASE>]	O(N)	Trả về biển kiểm soát của chuyến xe đã xuất phát của tuyến bus có mã <CODE> gần nhất thời điểm xem xét là <TIME>. Khi có giá trị <CASE>, ta cần trả về đúng chuyến tương ứng với chiều đi hoặc chiều về, ngược lại, ta xem xét cả hai chiều. Trong trường hợp không có chuyến nào thỏa mãn, trả về -1. Trong trường hợp ta đang xem xét hai chiều, nếu có cùng 2 chuyến đều thỏa mãn, ta ưu tiên trả về theo chiều đi.
GE <CODE> <TIME> [<CASE>]	O(N)	Trả về biển kiểm soát của chuyến xe đã kết thúc của tuyến bus có mã <CODE> gần nhất thời điểm xem xét là <TIME>. Khi có giá trị <CASE>, ta cần trả về đúng chuyến tương ứng với chiều đi hoặc chiều về, ngược lại, ta xem xét cả hai chiều. Trong trường hợp không có chuyến nào thỏa mãn, trả về -1. Trong trường hợp ta đang xem xét hai chiều, nếu có cùng 2 chuyến đều thỏa mãn, ta ưu tiên trả về theo chiều đi.

5 Quy định về thắc mắc và nộp bài

Sinh viên sẽ được cung cấp khung code khởi tạo trên site chung của môn học. Sinh viên làm bài trực tiếp trên hệ thống của BKeL và bài làm cuối cùng của sinh viên được sử dụng để chấm điểm.

Sinh viên được giải đáp thắc mắc trên diễn đàn trên site môn học, **TUYỆT ĐỐI KHÔNG GỬI EMAIL** cho giảng viên phụ trách để đặt câu hỏi.

—————**HẾT**—————