

Tổng Quan

Cấu Trúc Dữ Liệu và Giải Thuật

Mục tiêu

Bài tập thực hành này nhằm mục đích ôn lại cho sinh viên các kiến thức căn bản về ngôn ngữ lập trình Java và các cấu trúc dữ liệu cơ sở đã học. Bài thực hành cũng giúp sinh viên biết cách vận dụng các cấu trúc dữ liệu đã học để mô hình hóa bài toán thực tế và xây dựng các thuật toán trên cơ sở các chiến lược thiết kế để giải quyết vấn đề. Sau cùng, rèn luyện cho sinh viên khả năng phân tích, đánh giá độ phức tạp của thuật toán để đưa ra nhận định và lựa chọn được thuật toán hiệu quả để giải quyết vấn đề

Yêu cầu

Sinh viên hãy làm hoàn thiện các bài thực hành sau đây và nộp lại mã nguồn (source code) cho giảng viên vào buổi thực hành của tuần kế tiếp.

Nội dung

Câu 1: Viết chương trình nhập vào một số nguyên dương n và xuất ra màn hình tổng của tất cả các số lẻ dương nhỏ hơn n

Câu 2: Viết một hàm có tên là `isMultiple` nhận hai số nguyên m và n và trả về một số nguyên k nếu $m = n * k$, ngược lại, trả về giá trị -1 . Hãy xuất kết quả của hàm ra màn hình.

Câu 3: Viết một hàm có tên là `isEven` nhận một giá trị số nguyên n và trả về `true` nếu n là số chẵn, ngược lại, trả về `false`.

Yêu cầu: không được sử dụng phép chia lấy phần dư (%).

Hướng dẫn: dùng một vòng lặp để kiểm tra tính chẵn lẻ của n

Câu 4: Sử dụng các kiểu dữ liệu cơ bản trong Java, hãy xây dựng cấu trúc dữ liệu để lưu trữ trạng thái của một bàn cờ Caro có kích thước $m \times m$ ($0 \leq m, n \leq 20$). Với cấu trúc dữ liệu được xây dựng, hãy trình bày thuật toán và cài đặt chương trình để thực hiện các công việc sau:

- In ra màn hình bàn cờ Caro trong trạng thái hiện hành
- Kiểm tra xem có ai thắng hay không? Nếu có thì thông báo “Kết thúc”, nếu không có thì thông báo “Tiếp tục”

Câu 5: Cho một dãy số thực có n phần tử ($n \leq 10000$) và một số thực m . Hãy cho biết giá trị m có tồn tại trong dãy số trên hay không? Yêu cầu:

- a/ Xây dựng chương trình theo hướng thiết kế từ trên xuống (top-down design) để giải quyết bài toán trên
- b/ Xây dựng chương trình theo hướng thiết kế từ dưới lên (bottom-up design) để giải quyết bài toán trên
- c/ Dùng phương pháp thực nghiệm hãy tính xem chương trình xây dựng được ở **câu a** hoàn thành trong thời gian bao lâu ? Đơn vị tính: mili giây (milisecond)

Câu 6: Giả sử quy tắc tổ chức quản lý nhân viên của một công ty như sau:

Thông tin của một nhân viên bao gồm lý lịch và bản chấm công:

+ Lý lịch nhân viên:

- Mã nhân viên: chuỗi 8 ký tự
- Tên nhân viên: chuỗi 20 ký tự
- Tình trạng gia đình: 1 ký tự (M = Married – đã kết hôn, S = Single – còn độc thân)
- Số con: số nguyên ≤ 20
- Trình độ văn hóa: chuỗi 2 ký tự
(“C1” = cấp 1; “C2” = cấp 2; “C3” = cấp 3; “DH” = đại học; “CH” = cao học)
- Lương căn bản: số nguyên $\leq 10,000,000$

+ Bảng chấm công nhân viên:

- Số ngày nghỉ có phép trong tháng : số ≤ 28
- Số ngày nghỉ không phép trong tháng : số ≤ 28
- Số ngày làm thêm trong tháng : số ≤ 28
- Kết quả công việc : chuỗi 2 ký tự
(“T” = tốt; “TB” = đạt; “K” = kém)
- Lương thực lãnh trong tháng : số $\leq 20,000,000$

Quy tắc tính lương:

$$\text{Lương thực lãnh} = \text{lương căn bản} + \text{phụ cấp}$$

Trong đó tiền phụ cấp được tính theo qui tắc:

- Nếu số con > 2 : phụ cấp = 5% lương căn bản
- Trình độ văn hóa = CH: phụ cấp = 10% lương căn bản
- Làm thêm: phụ cấp = 4% lương căn bản/ngày
- Nghỉ không phép: phụ cấp = -5% lương căn bản/ngày

Chức năng yêu cầu:

- Cập nhật lý lịch, bảng chấm công cho nhân viên (thêm, xóa, sửa)
- Xem bảng lương hàng tháng



- Tìm thông tin của một nhân viên

Hãy tổ chức cấu trúc dữ liệu thích hợp cho chương trình để có thể biểu diễn được các thông tin trên, và cài đặt chương trình theo các chức năng đã mô tả.

Lưu ý:

- Nên phân biệt các thông tin mang tính chất tĩnh (lý lịch) và động (chấm công hàng tháng)
- Số lượng nhân viên cần quản lý là không quá nhiều (tối đa 1000 người)