

## Bài tập C++ nhóm cô Ngọc 2015 (lần 2)

21. Thực hiện cài đặt: stack, queue, danh sách liên kết đơn sử dụng

- a. Mảng
- b. Con trỏ

22. Viết chương trình nhập vào 1 dòng ký tự và sử dụng một stack để hiển thị dòng theo thứ tự ngược lại.

23. Viết chương trình sử dụng đối tượng stack để xác định 1 xâu có phải là một đối xứng hay không (xâu đọc xuôi và ngược giống nhau). Chương trình cần bỏ qua dấu cách và dấu câu.

24. Stacks thường dùng bởi trình biên dịch để xử lý việc tính biểu thức và sinh mã máy. Trong bài tập này và bài sau, chúng ta sẽ tìm hiểu trình biên dịch tính giá trị biểu thức số học bao gồm hằng, phép toán và dấu ngoặc như nào.

Con người thường viết biểu thức như  $3+4$  và  $7/9$  trong đó phép toán (+,/) được viết giữa 2 toán hạng, gọi là infix notation (trung tố). Máy tính “thích” postfix notation (hậu tố) hơn trong đó phép toán được viết bên phía phải của 2 toán hạng (ví dụ:  $3\ 4\ +$  và  $7\ 9\ /$ ).

### Viết chương trình chuyển biểu thức trung tố sang hậu tố

Gợi ý thuật toán:

Sử dụng 1 stack

b1. Push 1 dấu mở ngoặc '(' vào stack

b2. Thêm một dấu đóng ngoặc ')' vào biểu thức

b3. Khi stack chưa rỗng, đọc biểu thức trung tố từ trái qua phải và làm như sau:

- Nếu ký tự hiện thời là 1 ký hiệu số, copy nó vào phần tử tiếp của hậu tố.
- Nếu ký tự hiện thời là dấu mở ngoặc, đẩy nó vào stack
- Nếu ký tự hiện thời là 1 phép toán,
  - pop các phép toán (nếu có) ở đầu stack mà có mức ưu tiên cao hơn phép toán hiện thời và đẩy chúng vào biểu thức hậu tố.
  - push phép toán hiện thời vào stack
- Nếu ký tự hiện thời là dấu đóng ngoặc,
  - pop phép toán từ đỉnh stack và chèn vào biểu thức hậu tố cho tới khi xuất hiện dấu mở ngoặc ở đỉnh stack

- pop (và bỏ đi) dấu mở ngoặc khỏi đỉnh stack

Các phép toán cho phép trong biểu thức gồm: +, -, \*, /, ^ (mũ), %

**25. Tính giá trị biểu thức:** Viết chương trình tính giá trị biểu thức dạng hậu tố (giả sử chúng đúng định dạng). Ví dụ:  $6\ 2 + 5 * 8\ 4 / -$

Chương trình cần đọc 1 biểu thức dạng hậu tố bao gồm các chữ số và phép toán dưới dạng string. Sử dụng stack để duyệt biểu thức và tính giá trị. Thuật toán như sau:

Bước 1: Khi chưa duyệt tới cuối string, đọc biểu thức từ trái qua phải.

- Nếu kí tự hiện thời là 1 kí tự số:
  - **Push** giá trị integer tương ứng vào **stack** (bằng cách trừ đi kí tự '0')
- Ngược lại, kí tự hiện thời là 1 phép toán:
  - **Pop** 2 phần tử của **stack** vào biến x, y
  - Tính  $y <\text{phép toán}> x$ .
  - **Push** kết quả vào **stack**

Bước 2: Khi bạn đọc tới cuối string, **pop** phần tử đầu **stack** ra. Đây chính là kết quả của biểu thức hậu tố.

**26. “Ngôn ngữ teen”** thường khiến phụ huynh không hiểu được. Ví dụ (<3 nghĩa là love, hok -> không?). Viết chương trình nhập vào một văn bản text (không dấu) của teen và dịch sang văn bản phụ huynh hiểu được.

Gợi ý: Nên chuẩn bị 1 file từ điển. Rồi sau đó với file văn bản sẽ search xem có từ trong file không để dịch ra.

**27.** Viết chương trình thực hiện các phép toán cộng, trừ, nhân, chia trên đa thức. Các đa thức được biểu diễn bằng mảng động một chiều. Bậc của đa thức và các hệ số tương ứng được nhập từ bàn phím. Các đa thức được lưu trong file dathuc.dat và kết quả cần lưu vào file kqdathuc.dat

**28.** Viết chương trình thực hiện các phép toán cộng, trừ, nhân hai ma trận kích thước  $m \times n$ . Các ma trận được biểu diễn bằng mảng động hai chiều. Giá trị kích cỡ ma trận (m, n) và giá trị các phần tử của ma trận được lưu trong file matran.dat và kết quả cần lưu vào file kqmatran.dat

**29.** Viết 1 chương trình để kiểm tra trình độ hiểu biết của mọi người về Học viện BCVT gồm: 5 câu hỏi, mỗi câu hỏi có 4 lựa chọn đáp án. Các câu hỏi và đáp án được lưu ở file bcvt.txt.

Chương trình cần đếm số câu trả lời đúng và hiển thị thông báo cho người dùng. Ví dụ: nếu trả lời đúng hết 5 câu: xuất sắc; 4 câu: “Rất tốt”,..., 3 hoặc ít hơn: “Can phai tim hieu them ve truong”.

**30.** Tạo 1 struct sinh viên gồm các thông tin: họ tên, điểm, giới tính và thực hiện

- Thêm, Tìm kiếm, Sửa, Xoá vào danh sách sinh viên sao cho vẫn đúng thứ tự họ tên ABC
- Hiển thị danh sách theo điểm giảm dần
- Hiển thị danh sách theo giới tính
- Lưu danh sách sinh viên vào file.

Lưu ý: cần cài đặt bằng cả mảng và con trỏ

**31.** Nhập vào họ tên đầy đủ, tách riêng họ, đệm và tên.

Ví dụ: Nhập: Nguyen Thi Hong Anh

Ho: Nguyen

Dem: Thi Hong

Ten: Anh

**32. (Rùa và Thỏ).** Mô phỏng lại cuộc đua giữa Rùa và Thỏ sử dụng số random (ngẫu nhiên). Chiều dài đường đua là 70m (biểu diễn bằng 70 kí tự ‘.’), bắt đầu từ chân núi, kết thúc ở đỉnh núi. Kí hiệu rùa là R, thỏ là T. Mỗi giây đồng hồ, thỏ và rùa có thể đi tiếp, quay lui, hoặc đứng yên theo bảng bên dưới:

Con vật	Di chuyển	Tần suất	Giải thích
Rùa	Bò nhanh	50%	Tiến lên 3 m
	Trượt	20%	Trượt xuống 6m
	Bò chậm	30%	Tiến lên 1m
Thỏ	Ngủ	30%	Không di chuyển
	Nhảy xa	10%	Tiến lên 9m
	Trượt xa	10%	Trượt xuống 12m
	Nhảy gần	30%	Tiến lên 1m
	Trượt gần	20%	Trượt xuống 2m

Viết hàm moveTortoise, moveHare để điều chỉnh vị trí của Rùa và Thỏ theo luật ở bảng trên. Các hàm này cần sử dụng truyền tham chiếu sử dụng con trỏ (pointer-based pass-by-reference) để thay đổi vị trí của Rùa và Thỏ sau mỗi giây.

Sử dụng biến để lưu vị trí của các con vật. Vị trí bắt đầu là 1, nếu con nào bị trượt xuống từ 1 thì lại chuyển nó về vị trí 1 (không để giá trị âm). Sinh tần suất bằng 1 số integer  $i$  từ 1-10. Với Rùa, bò nhanh nếu  $i$  từ 1-5, Trượt nếu  $i$  từ 6-7, bò chậm nếu  $i$  từ 8-10. Làm tương tự với Thỏ.

Bắt đầu cuộc đua cần hiển thị “Bang!!!”

Với mỗi giây đồng hồ, hiển thị dòng dấu chấm tương ứng với 70m và vị trí của Rùa bằng R, Thỏ bằng T trên dòng này. Trường hợp Rùa và Thỏ ở cùng vị trí, Rùa sẽ cắn thỏ, chương trình cần hiển thị “Ouch!!!” bắt đầu từ vị trí trùng nhau.

Sau khi hiển thị mỗi dòng, cần kiểm tra xem đã con nào tới hoặc vượt mốc 70m chưa. Nếu Rùa thắng thì hiển thị “Rùa thắng”, nếu Thỏ thắng thì hiển thị “Thỏ thắng”. Nếu 2 con đến cùng lúc thì hiển thị “Đồng hàng”. Nếu chưa có con nào thắng thì thực hiện lặp lại.

### 33. Game TicTacToe:

a. Viết chương trình TicTacToe cho phép chơi game tic-tac-toe. Bảng điền sẽ là 1 mảng 2 chiều 3x3 kiểu integer. Ban đầu bảng trống. Cho phép 2 người chơi. Khi người 1 đi, điền 1 vào ô tương ứng. Khi người 2 đi, điền 2 vào ô tương ứng. Các lượt đi chỉ được điền vào ô còn trống. Sau mỗi lần đi, xác định game đã có người thắng hay cần tiếp tục.

Kết quả các bước đi cần lưu ở file tictactoe.dat

b. Cải thiện: cho phép máy tính chơi với người. Đồng thời, người chơi được chọn đi trước hay để máy đi trước.

*Game Tic-tac-toe: giống cờ caro, nhưng chỉ giới hạn bảng 3x3, ai được 3 ô thẳng hàng trước sẽ thắng.*

### 34. (Game Sudoku)

Cách chơi: Điền số từ 1 đến 9 vào những ô trống sao cho mỗi cột dọc, mỗi hàng ngang, mỗi phân vùng nhỏ (ô 3x3) có đủ các số từ 1 đến 9 mà không được lặp lại.

Bảng câu đố hình vuông, mỗi chiều có 9 ô nhỏ, hợp thành 9 cột, 9 hàng và được chia thành 9 ô lớn 3x3. Một vài ô nhỏ được đánh số, đó là những manh mối duy nhất để bạn tìm lời giải. Bảng câu đố lưu ở file text Sudoku.dat. File text này có 9 dòng, mỗi dòng có 9 số một chữ số: 0 tương ứng ô trên bàn cờ còn trống, 1-9 tương ứng với giá trị được điền ở vị trí tương ứng.

a. Viết chương trình kiểm tra xem người chơi điền có chính xác không.

b. Viết chương trình để tự động giải game Sudoku với 1 bảng câu đố cho trước. Ví dụ

003020600

900305001  
001806400  
008102900  
700000008  
006708200  
002609500  
800203009  
005010300

**35. (Phishing scanner)** Xác định các hành vi trộm cắp trong 1 email, trong đó người gửi thư yêu cầu bạn cung cấp các thông tin cá nhân như username, password, số thẻ credit card, số CMND. Email lừa đảo giả vờ là từ các ngân hàng thông dụng, các công ty thẻ tín dụng, các mạng xã hội và các dịch vụ thanh toán online trông có vẻ hợp pháp. Các thư này thường chứa links tới các websites giả mạo yêu cầu bạn cung cấp các thông tin riêng tư.

Truy cập [antiphishing.org](http://antiphishing.org), McAfee , Security Extra, ... và các trang web khác để tìm danh sách các tổ chức lừa đảo mới nhất.

Tạo ra 1 danh sách 30 từ, cụm từ và tên công ty thường xuất hiện trong các email giả mạo. Gán điểm cho từng phần tử trong danh sách trên ước lượng của bạn về cấp độ giả mạo (ví dụ: 1 điểm nếu hơi có vẻ như lừa đảo, 2 điểm nếu có vẻ như lừa đảo, 3 điểm nếu chắc chắn là lừa đảo). Viết chương trình kiểm tra 1 file text để tìm các từ và cụm từ trong danh sách. Đếm số lần xuất hiện của từng từ/cụm từ trong danh sách. Cuối cùng, hiển thị danh sách từ, cụm từ và số lần xuất hiện tương ứng, đồng thời hiển thị tổng điểm của cả file text.

- *Chương trình của bạn có gán điểm cao cho những email lừa đảo mà bạn đã từng nhận?*
- *Chương trình có gán điểm thấp cho những email bình thường (không giả mạo) mà bạn đã từng nhận?*