

Bài tập thực hành

Cấu trúc dữ liệu và thuật toán

Ngăn xếp (B)

1. Bài 1

Viết chương trình đọc một chuỗi từ file và sử dụng ngăn xếp để đảo ngược chuỗi đó.

INPUT.TXT: Mỗi dòng chứa một chuỗi cần đảo.

OUTPUT.TXT: Dòng i chứa chuỗi sau khi đã đảo ngược tương ứng với chuỗi cần đảo ở dòng i trong file INPUT.TXT

Ví dụ:

INPUT.TXT

ABCDEF

AB

OUTPUT.TXT

FEDCBA

BA

2. Bài 2

Cho một Stack lưu các ký tự chữ cái, được lưu trữ bởi một mảng ký tự, ban đầu Stack rỗng. Có hai loại thao tác được ký hiệu như sau:

1: là lấy một ký tự ra khỏi Stack, nếu Stack rỗng thì bỏ qua

2 c: là thêm một ký tự chữ cái c vào trong Stack, nếu Stack đầy thì bỏ qua

Dữ liệu đầu vào cho trong file INPUT.TXT: Dòng đầu tiên gồm một số nguyên dương n tương ứng với có n hành động. Tiếp theo có n dòng, mỗi dòng lưu thông tin của một hành động.

Kết quả đầu ra ghi vào file OUTPUT.TXT: Mỗi dòng in ra một ký tự khi thực hiện thao tác lấy ra trong Stack cho đến khi hết.

Ví dụ:

INPUT.TXT

6

2 a

2 b

2 c

1

OUTPUT.TXT

b

a

3. Bài 3

Cho một biểu thức chỉ chứa các dấu ngoặc $()$, $[]$, $\{\}$. Kiểm tra xem biểu thức có hợp lệ hay không, tức là các dấu ngoặc được mở và đóng đúng thứ tự.

INPUT.TXT: Mỗi dòng chứa một chứa biểu thức cần kiểm tra.

OUTPUT.TXT: Dòng i ghi kết quả kiểm tra của biểu thức trong dòng i của file INPUT.TXT. Ghi YES nếu biểu thức hợp lệ, NO nếu không.

Ví dụ 1:

INPUT.TXT

$([]\{\})$

$([])$

OUTPUT.TXT

YES

NO

4. Bài 4

Cho một biểu thức hậu tố gồm các toán tử $+$, $-$, $*$, $/$ và các số nguyên. Hãy tính giá trị biểu thức đó bằng ngăn xếp.

INPUT.TXT: Mỗi dòng chứa một biểu thức hậu tố, các phần tử cách nhau bởi dấu cách.

OUTPUT.TXT: Dòng i là giá trị kết quả (số nguyên) của biểu thức ở dòng i trong file INPUT.TXT

Ví dụ:

INPUT.TXT

45+

5 1 2 + 4 * + 3 -

OUTPUT.TXT

9

14

5. Bài 5

Sử dụng ngăn xếp để chuyển đổi một số nguyên dương sang hệ nhị phân.

INPUT.TXT: Mỗi dòng chứa một số nguyên dương n.

OUTPUT.TXT: Dòng i chứa biểu diễn nhị phân của số nguyên dương ở dòng i trong file INPUT.TXT.

Ví dụ:

INPUT.TXT

2

19

OUTPUT.TXT

10

10011

6. Bài 6 (*)

Viết chương trình thực hiện đổi biểu thức trung tố thành biểu thức hậu tố sử dụng cấu trúc dữ liệu ngăn xếp.

Khai báo của hàm chuyển đổi như sau:

void infixToPostfix(char* exp, char* postfix)

Input:

- exp là string của biểu thức trung tố.
- postfix là địa chỉ của string chứa biểu thức hậu tố.

Output:

- postfix là string của biểu thức hậu tố.

Dữ liệu đầu vào cho trong file INPUT.TXT: Mỗi dòng chứa một biểu thức trung tố (một string, không có dấu cách).

Kết quả đầu ra ghi vào file OUTPUT.TXT: Dòng i chứa chuỗi biểu thức hậu tố tương ứng với chuỗi biểu thức trung tố trong dòng i của file INPUT.TXT

Ví dụ:

INPUT.TXT

$a+b*(c^d-e)^{(f+g*h)}-i$

$a+b$

OUTPUT.TXT

$abcd^e-fgh*+^*+i-$

$ab+$

7. Bài 7 (*)

Mô phỏng chức năng UNDO và REDO của trình soạn thảo văn bản bằng hai ngăn xếp. Mỗi thao tác là một chuỗi ký tự DO x, UNDO, hoặc REDO.

INPUT.TXT:

Dòng đầu: Số nguyên N là số thao tác. N dòng tiếp theo là các thao tác DO x, UNDO, REDO.

OUTPUT.TXT: Một dòng là chuỗi thao tác đã thực hiện cuối cùng (cách nhau bởi dấu cách).

Ví dụ:

INPUT.TXT

6

DO A

DO B

UNDO

DO C

REDO

DO D

OUTPUT.TXT

A C D