Nội dung

* Stack - ngăn xếp
* Hàng đợi Queue

Container – kiểu cấu trức dữ liệu trong đó việc lưu trữ và lấy phần tử nó chỉ phụ thuộc vào thứ tự, không liên quan đến phần tử mà nó lưu trữ

VD. Mảng, danh sách liên kết - linkedlist, danh sách tuyến tính, Stack, Queue, deque (hàng đợi 2 đầu), hàng đợi ưu tiên - priority queue,…

Ngăn xếp - stack

* LIFO – vào sau ra trước
* Lưu trữ và lấy dữ liệu chỉ tại 1 đầu - top
* Đảo thứ tự --> cần đảo thứ tự danh sách
* Cài đặt dùng mảng hoặc dùng danh sách liên kết đơn

Cài đặt stack dùng danh sách liên kết đơn

* Đỉnh stack = đầu danh sách --> việc thêm và lấy phần tử chỉ diễn ra tại đầu --> chỉ cần 1 con trỏ

Bài toán kiểm tra cặp ngoặc hợp lệ

* Có 3 kiểu dấu ngoặc () [] và {}
* Hợp lệ khi 1 ngoặc mở phải có 1 ngoặc đóng tương ứng

VD.

|  |  |
| --- | --- |
| ()()([]{}) | OK |
| ()([)] | FAIL |
| {[](}) | FAIL |

Thuật toán kiểm tra

* Dùng 1 ngăn xếp để chứa các dấu mở ngoặc
* Lần lượt duyệt chuỗi dấu ngoặc
  + Nếu gặp mở ngoặc --> đẩy vào stack
  + Nếu gặp đóng ngoặc --> lấy ra 1 mở ngoặc ở đỉnh stack, check xem đóng ngoặc và mở ngoặc đó có phải là 1 cặp hay không
    - Nếu là 1 cặp --> check tiếp
    - Ngược lại --> dừng và báo lỗi

()([)]

|  |  |  |
| --- | --- | --- |
| TT | STACK | Giải thích |
| ( | ( | Mở ngoặc --> đẩy vào stack |
| ) | Rỗng | () là hợp lệ --> duyệt tiếp |
| ( | ( | Mở ngoặc --> đẩy vào stack |
| [ | ( [ | Mở ngoặc --> đẩy vào stack (đỉnh stack là bên phải) |
| ) | ( | Cặp [) là không hợp lệ --> FAIL |
| ] |  |  |

Chú ý:

* Khi dùng danh sách liên kết thì mỗi con trỏ được coi là bộ nhớ lãng phí
* KHÔNG nên dùng danh sách liên kết để lưu trữ phần tử mà kích thước phần tử chỉ là dữ liệu đơn giản (int, char, float,…). Nếu dữ liệu chỉ là kiểu đơn giản --> nên biểu diễn bằng mảng để tiết kiệm bộ nhớ

Bài tập 1. Mở rộng bài toán check dấu ngoặc cho biểu thức tổng quát dạng : **A=6+7\*(c/2\*(6-3\*c[2]))**

* Vẫn có 3 loại dấu ngoặc () [] {}
* Khi duyệt, nếu không phải dấu ngoặc --> bỏ qua
* In ra thứ tự ký tự bị lỗi (trong trường hợp thiếu dấu ngoặc mở cũng cần in ra thông báo đầy đủ)
* Biểu thức được nhập vào từ bàn phím hoặc từ file văn bản

Trong C++ stack được cài đặt trong header stack

Khi dùng chỉ cần include file header ở trên

Pair là 1 bộ

* phần tử đầu tiên sẽ là first
* phàn tử tiếp sẽ là second

|  |
| --- |
| #include <iostream>  #include <utility>  **using** **namespace** std;    **int** main()  {  pair <**int**, **char**> PAIR1 ;  pair <string, **double**> PAIR2 ("GeeksForGeeks", 1.23) ;  pair <string, **double**> PAIR3 ;    PAIR1.first = 100;  PAIR1.second = 'G' ;    PAIR3 = make\_pair ("GeeksForGeeks is Best",4.56);    cout << PAIR1.first << " " ;  cout << PAIR1.second << endl ;    cout << PAIR2.first << " " ;  cout << PAIR2.second << endl ;    cout << PAIR3.first << " " ;  cout << PAIR3.second << endl ;    **return** 0;  } |

Dùng C++ stack và pair để làm lại bài toán check dấu ngoặc của biểu thức hợp lệ

Bài tập về nhà

**Bài 1**. Viết chương trình kiểm tra 1 xâu có phải đối xứng - palindrome **dùng stack**

ABBA --> OK

ABA --> OK

AbbA --> OK

ABbA --> Fail

Xâu được nhập vào từ file hoặc bàn phím

**Bài 2**. Viết chương trình kiểm tra các tag html có hợp lệ hay không

* 1 thẻ mở phải cóp 1 thẻ đóng tương ứng
* Có 1 số thẻ KHÔNG cần thể đóng
* 1 số thẻ có thể vừa mở và đóng đồng thời

|  |
| --- |
| <h1 class="dt-news\_\_title">  Phó Ch? t?ch TPHCM: Có tr??ng h?p tiêm 2 m?i vaccine Covid-19 v?n t? vong  </h1>  <div class="dt-news\_\_audio fl-center">  <div class="dt-audio"><div class="dta-main"><a href="javascript:void(0);" class="aud-btn "><div class="icon40 play"></div></a><a href="javascript:void(0);" class="aud-btn hidden"><div class="icon40 pause"></div></a><div class="dt-graph prel"><div class="aud-text">0:00/0:00</div><div class="aud-bg"><div class="seek" style="width:0;"></div></div></div><a class="aud-btn speech doc-tiep" href="javascript:void(0);"><span class="doc-tiep-img"></span></a><div class="speech-section"><a class="speech-section-initial " href="javascript:void(0);"><div class="spico speech-female"></div><div class="speech-section-initial-description">Phía B?c</div><div class="arrow-down"></div></a></div></div></div>  </div> |

Cách làm giống với check dấu ngoặc hợp lệ, nhưng số lượng trường hợp lớn hơn nhiều

Nên dùng java hoặc C++

**Bài 3**. Cho 1 mảng số nguyên, với mỗi phần tử hãy tìm phần tử lớn hơn phần tử hiện tại đầu tiên bên phải - Next Greater Element

VD. 1,5,8,7,6,8

1 thì NGE là 5

5 sẽ có NGE là 8

8 KHÔNG có NGE, trả về -1

7 có NGE là 8

6 có NGE là 8

8 cuối dãy, nên không có NGE

đầu vào : 1,5,8,7,6,8

Tạo ra bảng kết quả là 5,8,-1,8,8,-1