# Bài 3: Thực hành với dữ liệu ngăn xếp

**Giải Chuyên đề Tin học 12 Bài 3: Thực hành với dữ liệu ngăn xếp**  
**Khởi động trang 13 Chuyên đề Tin học 12**: Trong bài trước, các em đã học cách thiết lập kiểu dữ liệu ngăn xếp. Kiểu dữ liệu ngăn xếp được sử dụng khá phổ biến trong các ứng dụng thực tế. Theo em, có thể sử dụng kiểu dữ liệu này để mô phỏng chức năng quay lại trang web đã duyệt trong các trình duyệt thông dụng như Google Chrome hay Bing được không?  
**Lời giải:**  
Trong bài trước, các em đã học cách thiết lập kiểu dữ liệu ngăn xếp. Kiểu dữ liệu ngăn xếp được sử dụng khá phổ biến trong các ứng dụng thực tế. Có thể sử dụng kiểu dữ liệu ngăn xếp để mô phỏng chức năng quay lại trang web đã duyệt trong các trình duyệt thông dụng như Google Chrome hay Bing.  
**Luyện tập 1 trang 15 Chuyên đề Tin học 12**: Trong bài trước, các em đã học cách thiết lập kiểu dữ liệu ngăn xếp. Kiểu dữ liệu ngăn xếp được sử dụng khá phổ biến trong các ứng dụng thực tế. Có thể sử dụng kiểu dữ liệu ngăn xếp để mô phỏng chức năng quay lại trang web đã duyệt trong các trình duyệt thông dụng như Google Chrome hay Bing.  
**Lời giải:**  
from Stack import \*  
backward = Stack()  
forward = Stack()  
option = 0  
while option != 4:  
 print("1: Đi đến trang web mới")  
 print("2: Trở về trang web trước đó")  
print("3: Đi đến trang web kế tiếp")  
 option = int(input(“Hãy nhập vào lựa chọn của bạn:\n”))  
 if option == 1:  
 website = input(“Hãy nhập vào địa chỉ website muốn đi đến:\n”)  
 push(backward,website)  
 print(“Đang đi đến trang web: ” + website)  
 elif option == 2:   
 if isEmptyStack(backward)  
 print(“Không tồn tại lịch sử duyệt web”)  
 else:  
 website = backward.pop()  
 forward.push(website)  
 if backward.isEmpty():  
 print ("Không tồn tại lịch sử duyệt web")  
 else:  
 print(“Đang đi đến trang web: ” + backward.peek())  
 elif optione == 3:  
 if forward.isEmpty():  
 print ("Không có trang web nào để quay lại")  
 else:  
 website = forward.pop()  
 backward.push(website)  
 print(“Đang đi đến trang web: ” + website)  
 else:   
 print(“Lựa chọn không hợp lệ. Vui lòng chọn lại.”)  
**Luyện tập 2 trang 15 Chuyên đề Tin học 12**: Sửa chương trình trong Nhiệm vụ 2 để in ra màn hình tổng số cặp đóng mở ngoặc của từng loại xuất hiện trong biểu thức.  
**Lời giải:**  
Thêm các biến:  
so\_ngoac\_tron\_mo = 0  
so\_ngoac\_tron\_dong = 0  
so\_ngoac\_vuong\_mo = 0  
so\_ngoac\_vuong\_dong = 0  
Thêm điều kiện vào dòng 7 (Theo như chương trình ở Nhiệm vụ 2):  
 if bieuthuc[i] == “(”  
 so\_ngoac\_tron\_mo += 1  
 else:  
 so\_ngoac\_vuong\_mo += 1  
Thêm điều kiện vào dòng 9 (Theo như chương trình ở Nhiệm vụ 2):  
 if bieuthuc[i] == “)”  
 so\_ngoac\_tron\_dong += 1  
 else:  
 so\_ngoac\_vuong\_dong += 1  
Thêm in ra màn hình vào dòng 23 (Theo như chương trình ở Nhiệm vụ 2):  
print(f"Số lượng ngoặc tròn: {dem\_ngoac\_mo\_tron} mở, {dem\_ngoac\_dong\_tron} đóng")   
print(f"Số lượng ngoặc vuông: {dem\_ngoac\_mo\_vuong} mở, {dem\_ngoac\_dong\_vuong} đóng")  
**Vận dụng 1 trang 15 Chuyên đề Tin học 12**: Hãy viết chương trình mô phỏng quá trình xếp và lấy sách ra khỏi một ngăn tủ. Cho trước một số quyển sách, lần lượt xếp các quyển sách này vào ngăn tủ. Khi lấy ra, sách sẽ được lấy ra theo thứ tự quyển nào đưa vào sau sẽ được lấy ra trước. Để lấy được một quyền sách, chúng ta phải lấy các quyền sách ở phía ngoài ra trước. Ví dụ các quyển sách được xếp vào tủ theo thứ tự như sau: [English, Physic, Maths, Chemistry, History, Biology]  
Nếu muốn lấy quyển sách Maths ra khỏi ngăn sách thì chúng ta cần lấy các quyển Biology, History, Chemistry ra trước.   
Cho trước tệp chứa tên các quyển sách. Hãy tạo một ngăn xếp và đưa các quyển sách trong tệp vào ngăn xếp. Sau đó cho người dùng nhập vào tên quyển sách muốn lấy ra và in ra màn hình số quyển sách cần lấy ra trước khi lấy được quyển sách muốn lấy.   
**Lời giải:**  
Hướng dẫn và gợi ý cho bài toán: Có thể sử dụng ngăn xếp (stack) - một cấu trúc dữ liệu hoạt động theo nguyên tắc "Last In, First Out" (LIFO). Các bước thực hiện:  
- Bước 1: Đọc danh sách các quyển sách từ tệp và đưa vào ngăn xếp.  
- Bước 2: Yêu cầu người dùng nhập tên quyển sách muốn lấy ra.  
- Bước 3: Duyệt qua ngăn xếp để tìm quyển sách đó, đồng thời đếm số quyển sách cần lấy ra trước nó.  
Mã Python mẫu:  
# Hàm để đọc danh sách các quyển sách từ tệp  
def read\_books\_from\_file(file\_path):  
 with open(file\_path, 'r') as file:  
 books = [line.strip() for line in file]  
 return books  
# Hàm để mô phỏng quá trình lấy sách ra khỏi ngăn tủ  
def simulate\_book\_retrieval(books, book\_to\_retrieve):  
 stack = books[:]  
 count = 0  
 while stack:  
 current\_book = stack.pop()  
 count += 1  
 if current\_book == book\_to\_retrieve:  
 return count  
 return -1 # Trường hợp không tìm thấy quyển sách cần lấy  
def main():  
 # Đường dẫn tới tệp chứa danh sách các quyển sách  
 file\_path = ‘books.txt’  
 # Đọc danh sách các quyển sách từ tệp  
 books = read\_books\_from\_file(file\_path)  
 # Yêu cầu người dùng nhập tên quyển sách muốn lấy ra  
 book\_to\_retrieve = input("Nhập tên quyển sách bạn muốn lấy ra: ")  
 # Mô phỏng quá trình lấy sách ra  
 count = simulate\_book\_retrieval(books, book\_to\_retrieve)  
 if count == -1:  
 print(f"Không tìm thấy quyển sách '{book\_to\_retrieve}' trong ngăn tủ.")  
 else:  
 print(f"Cần lấy {count-1} quyển sách trước khi lấy được quyển '{book\_to\_retrieve}'.")  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()  
**Giải thích mã:**  
 - Hàm read\_books\_from\_file: Đọc danh sách các quyển sách từ tệp và lưu vào một danh sách.  
 - Hàm simulate\_book\_retrieval:  
Tạo một ngăn xếp từ danh sách các quyển sách.  
Duyệt qua ngăn xếp bằng cách pop từng quyển sách ra và đếm số lượng quyển sách đã lấy.  
Nếu tìm thấy quyển sách cần lấy, trả về số lượng sách đã lấy.  
 - Hàm main:  
Đọc danh sách các quyển sách từ tệp.  
Yêu cầu người dùng nhập tên quyển sách muốn lấy.  
Gọi hàm mô phỏng quá trình lấy sách và in kết quả.  
**Lưu ý:** rằng bạn cần phải có tệp books.txt chứa danh sách các quyển sách, mỗi quyển sách trên một dòng. Dưới đây là một ví dụ về nội dung tệp books.txt:  
English  
Physic  
Maths  
Chemistry  
History  
Biology  
Chạy chương trình và nhập tên quyển sách muốn lấy ra để xem kết quả.  
**Vận dụng 2 trang 15 Chuyên đề Tin học 12**: Cải tiến chương trình trong Nhiệm vụ 2 để có thể kiểm tra biểu thức có chứa ba loại dấu đóng mở ngoặc "()", "[]", "{}"  
**Lời giải:**  
Để cải tiến chương trình kiểm tra biểu thức có chứa ba loại dấu đóng mở ngoặc "()", "[]", và "{}", chúng ta cần thêm hỗ trợ cho cặp ngoặc nhọn "{}". Điều này chỉ đòi hỏi một vài thay đổi nhỏ trong mã nguồn gốc để bao gồm các dấu ngoặc mới này.  
Gợi ý phiên bản đã được cải tiến của chương trình kiểm tra biểu thức với ba loại dấu đóng mở ngoặc:  
class Stack:  
 def \_\_init\_\_(self):  
 self.items = []  
 def is\_empty(self):  
 return self.items == []  
 def push(self, item):  
 self.items.append(item)  
 def pop(self):  
 if not self.is\_empty():  
 return self.items.pop()  
 else:  
 return None  
 def peek(self):  
 if not self.is\_empty():  
 return self.items[-1]  
 else:  
 return None  
def kiemtrabt(bieuthuc):  
 hople = True  
 ngoacmo = Stack()  
 for i in range(len(bieuthuc)):  
 if bieuthuc[i] in "([{":  
 ngoacmo.push(bieuthuc[i])  
 elif bieuthuc[i] in ")]}":  
 if ngoacmo.is\_empty():  
 hople = False  
 break  
 else:  
 tmp = ngoacmo.pop()  
 if (bieuthuc[i] == ")" and tmp != "(") or \  
 (bieuthuc[i] == "]" and tmp != "[") or \  
 (bieuthuc[i] == "}" and tmp != "{"):  
 hople = False  
 break  
 if not ngoacmo.is\_empty():  
 hople = False  
 return hople  
bieuthuc = input("Hãy nhập vào một biểu thức: \n")  
hople = kiemtrabt(bieuthuc)  
if hople:  
 print("Biểu thức hợp lệ")  
else:  
 print("Biểu thức không hợp lệ")  
Trong đoạn mã trên:  
- Chúng ta định nghĩa một lớp Stack để mô phỏng ngăn xếp.  
- Hàm kiemtrabt kiểm tra tính hợp lệ của biểu thức với các cặp ngoặc "()", "[]", và "{}".  
- Chúng ta duyệt qua từng ký tự trong biểu thức và xử lý các dấu ngoặc mở bằng cách đẩy chúng vào ngăn xếp, và xử lý các dấu ngoặc đóng bằng cách kiểm tra xem chúng có khớp với dấu ngoặc mở tương ứng ở đỉnh ngăn xếp hay không.