

## HƯỚNG DẪN THỰC HÀNH

### DANH SÁCH LIÊN KẾT - NGĂN XẾP - HÀNG ĐỢI

#### I. Mục tiêu

Sinh viên cài đặt các cấu trúc dữ liệu cơ bản như danh sách liên kết, ngăn xếp, hàng đợi và ứng dụng nó để giải quyết các bài toán đặt ra.

#### II. Qui định nộp

- Sinh viên nộp một tập tin nén, có tên là **<MSSV>.zip** hoặc **<MSSV>.rar** chứa source code và báo cáo của chương trình.
- Sinh viên nộp kèm một file báo cáo ghi mức độ hoàn thành công việc của mình, các bộ dữ liệu mà mình test ở mỗi bài.
- Tất cả các bài tập sẽ lập trình theo command line (tham số dòng lệnh).
- Kiến trúc thư mục nộp:
  - o MSSV
    - MSSV\_BT1:
      - Source: chứa các file .cpp
      - Header: chứa các file .h
      - Test case: chứa các file input để test thử.
      - ...
    - MSSV\_BT2: ....
- Môi trường làm việc: Visual Studio 2015 hoặc các môi trường tương đương. Không sử dụng các hàm bị lỗi hồng bảo mật như gets, ...
- Hạn nộp: xem link trên Moodle.
- **Bài giống nhau hay nộp file rác sẽ 0 điểm MÔN HỌC.**

#### III. Nội dung

**BT1.** Sử dụng cấu trúc dữ liệu ngăn xếp Stack (cài đặt bằng danh sách liên kết Linked List), viết chương trình cho phép đọc nội dung từ tập tin và kiểm tra tính hợp lệ trong việc sử dụng các toán tử của một biểu thức toán học gồm các toán hạng là các số nguyên và các toán tử: +, -, \*, /, (, ), [, ], {, }.

**Lưu ý:** cần cài đặt đầy đủ các hàm cơ bản của ngăn xếp Stack: *push*, *pop*, *isEmpty*, *top*.

- **Ví dụ 3.1.1:** biểu thức  $1 + [2 / (3 - 4) + 5] * 6$  là hợp lệ.

- **Ví dụ 3.1.2:** biểu thức  $1 + (2 / [3 + 4) - 5] * 6$  là không hợp lệ (sai dấu đóng mở ngoặc tròn và ngoặc vuông).

- **Ví dụ 3.1.3:** biểu thức  $(1 + ) / 3 + (4 5) 6$  là không hợp lệ (vị trí thứ 3: sau dấu cộng đầu tiên không có toán hạng, vị trí thứ 9: sau toán hạng 4 không có toán tử phù hợp).

**Gợi ý kiểm tra tính hợp lệ trong việc sử dụng dấu đóng mở ngoặc:**

- **Ví dụ 3.1.1:**  $1 + [2 / (3 - 4) + 5] * 6$

Ta lần lượt duyệt các kí tự của biểu thức, nếu gặp dấu mở ngoặc thì **push** vào trong ngăn xếp, ngược lại nếu gặp dấu đóng ngoặc **phù hợp** thì **pop** ra khỏi ngăn xếp, nếu gặp dấu đóng ngoặc không phù hợp thì biểu thức không hợp lệ.

Duyệt đến kí tự thứ 3: gặp dấu mở ngoặc vuông  $\rightarrow push('[') \rightarrow$  nội dung ngăn xếp là  $["$ .

Duyệt đến kí tự thứ 5: gặp dấu mở ngoặc tròn  $\rightarrow push('(') \rightarrow$  nội dung ngăn xếp là  $["($ .

Duyệt đến kí tự thứ 9: gặp dấu đóng ngoặc tròn  $\rightarrow cần pop('(') \rightarrow$  hợp lệ  $\rightarrow$  nội dung ngăn xếp là  $["$ .

Duyệt đến kí tự thứ 12: gặp dấu đóng ngoặc vuông  $\rightarrow cần pop('[') \rightarrow$  hợp lệ  $\rightarrow$  nội dung ngăn xếp là  $""$  (**rỗng**).

Duyệt được hết toàn bộ biểu thức thì ngăn xếp rỗng  $\rightarrow$  biểu thức sử dụng dấu ngoặc hợp lệ.

- **Ví dụ 3.1.2:**  $1 + (2 / [3 + 4) - 5] * 6$

Duyệt đến kí tự thứ 3: gặp dấu mở ngoặc tròn  $\rightarrow push('(') \rightarrow$  nội dung ngăn xếp là  $["($ .

Duyệt đến kí tự thứ 5: gặp dấu mở ngoặc vuông  $\rightarrow push('[') \rightarrow$  nội dung ngăn xếp là  $["([$ .

Duyệt đến kí tự thứ 9: gặp dấu đóng ngoặc tròn  $\rightarrow cần pop('(') \rightarrow$  không hợp lệ do đỉnh ngăn xếp hiện giờ là  $[']$   $\rightarrow$  biểu thức sử dụng dấu ngoặc không hợp lệ.

**Command line: MSSV\_BT1.exe kt input.txt output.txt**

Ví dụ: 1512345\_BT1.exe kt in1.txt out1.txt

Nghĩa là kiểm tra biểu thức chứa trong tập tin in1.txt và xuất kết quả ra out1.txt

**Định dạng input:**

- Dòng thứ tự lẻ chứa số thành phần của biểu thức
- Dòng thứ tự chẵn nội dung của biểu thức tương ứng, mỗi thành phần của biểu thức cách nhau 1 khoảng trắng.

- Ví dụ:

15

1 + [2 / (3 - 4) + 5] \* 6

11

(1 + ) / 3 + (4 5)

**Định dạng output:**

- Biểu thức hợp lệ hay không, nếu không hợp lệ thì tại vị trí nào.

- Ví dụ:  
 $1 + [2 / (3 - 4) + 5] * 6$  là biểu thức hợp lệ  
 $(1 + ) / 3 + (4 5)$  không hợp lệ tại vị trí: 3, 9

**BT2.** Cài đặt một hàng đợi Queue cho dữ liệu sinh viên (mã sinh viên, họ và tên, ngày tháng năm sinh) với các hàm cơ bản *enqueue*, *dequeue*, *isEmpty* và *front* sử dụng danh sách liên kết Linked List.

**Command line: MSSV\_BT2.exe Queue Data.txt Input.txt Output.txt**

Ví dụ: 1512345\_BT2.exe Queue data1.txt in1.txt out1.txt

Nghĩa là đọc danh sách dữ liệu sinh viên chứa trong tập tin data1.txt, các lời gọi enqueue, dequeue trong in1.txt và xuất danh sách dữ liệu kết quả ra out1.txt

**Định dạng data:**

- Mỗi dòng chứa các thông tin dữ liệu sinh viên, cách nhau bởi dấu “,”
- Ví dụ:  
1512001, Nguyen Van A, 27/02/1997  
1512002, Nguyen Thi B, 06/11/1997  
1512003, Nguyen Van B, 20/09/1997

**Định dạng input:**

- Các lần gọi enqueue và dequeue cần thực hiện.
- Ví dụ:  
enqueue 1512004, Nguyen Thi C, 15/09/1997  
dequeue  
dequeue

**Định dạng output:**

- Giống định dạng tập tin data.

**BT3.** (không bắt buộc) Mở rộng bài tập 1, viết chức năng tính giá trị biểu thức nếu biểu thức hợp lệ.

**Command line: MSSV\_BT1.exe gt input.txt output.txt**

Ví dụ: 1512345\_BT3.exe gt in1.txt out1.txt

Nghĩa là kiểm tra và tính giá trị biểu thức chứa trong tập tin in1.txt và xuất kết quả ra out1.txt

**Định dạng input:**

- Như bài tập 1

**Định dạng output:**

- Giá trị của biểu thức nếu biểu thức hợp lệ, nếu không hợp lệ thì tại vị trí nào.
- Ví dụ:  
 $1 + [2 / (3 - 4) + 5] * 6 = 19$   
 $(1 + 2) / 3 + (4 - 5) * 6 = -5$   
 $(1 + ) / 3 + (4 5)$  không hợp lệ tại vị trí: 3, 9