

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA TOÁN – TIN



BÁO CÁO BÀI TẬP LỚN
Môn: Lập Trình Hướng Đối Tượng

Sinh Viên Thực Hiện :Trần Thị Kỳ Phương
Lớp :20TTH
MSSV : 20110287
Giảng Viên Hướng Dẫn :ThS.Nguyễn Ngọc Long



Thành Phố Hồ Chí Minh, ngày 10 tháng 6 năm 2022.

Lời cảm ơn

Lời đầu tiên, em xin gửi lời cảm ơn chân thành và sự tri ân sâu sắc đối với thầy Ths. Nguyễn Ngọc Long phụ trách bộ môn Lập Trình Hướng Đối Tượng đã nhiệt tình hướng dẫn, giảng dạy môn học để em hoàn thiện Bài Tập Lớn.

Trong quá trình nghiên cứu, cũng như là trong quá trình làm dự án, khó tránh khỏi sai sót, rất mong các thầy, cô bỏ qua. Đồng thời do trình độ lý luận cũng như kinh nghiệm thực tiễn còn hạn chế nên bài báo cáo không thể tránh khỏi những thiếu sót, em rất mong nhận được ý kiến đóng góp thầy, cô để em học thêm được nhiều kinh nghiệm và sẽ hoàn thành tốt hơn dự án của mình.

Em xin chân thành cảm ơn!

Mục lục

LỜI CẢM ƠN	1
I.BÀI TOÁN CẦN GIẢI QUYẾT,MÔ TẢ QUY TRÌNH THỰC HIỆN CÁC CÔNG VIỆC CHÍNH.....	3
a) Bài toán cần giải quyết.....	3
b) Quy trình thực hiện	3
c) Thảo luận và kết quả của sản phẩm	3
II. XÂY DỰNG LỚP STACK	4
a) STACK	4
b) Mô tả quá trình xây dựng lớp STACK.....	5
III. XÂY DỰNG CẤU TRÚC DỮ LIỆU	6
a) Danh sách liên kết đôi	6
b) Mô tả quá trình xây dựng cấu trúc dữ liệu	6
IV. XÂY DỰNG CALCULATOR.....	7
a) Thuật toán trong CALCULATOR	7
b) Kết quả.....	8
TÀI LIỆU THAM KHẢO	9

I. GIỚI THIỆU BÀI TOÁN CẦN GIẢI QUYẾT, MÔ TẢ QUY TRÌNH THỰC HIỆN CÁC CÔNG VIỆC CHÍNH.

a) Bài Toán Cần Giải Quyết

Xây dựng lớp biểu diễn khái niệm Stack. Sử dụng Stack kể trên, viết một ứng dụng cho phép giải quyết bài tính giá trị một biểu thức trung tố (không dùng đệ qui). Biểu thức trung tố có thể bao gồm cả phép gọi hàm các hàm thông dụng như sqrt, exp, log, sin, cos...

Ví dụ: Nhập: $(5+2*2) - \text{sqrt}(4)$

Xuất: 7

Ngôn ngữ lập trình: c++

Công cụ phát triển: Visual Studio 2019

b) Quy Trình Thực Hiện

Bước 1: Dựa các tính năng có sẵn trên máy tính Casio FX-570 ES PLUS hoặc Casio FX-470 VN PLUS, để nghiên cứu các chức năng mà cần lập trình ở các đối tượng trong bài toán.

Bước 2: Đưa ra các bài toán và hướng giải quyết tổng quát.

Bước 3: Viết thuật toán và vẽ lưu đồ để khai triển.

Bước 4: Dựa trên thuật toán và lưu đồ đã có để viết chương trình trên Visual Studio C++

Bước 5: Kiểm tra các chức năng qua nhiều dạng biểu thức khác nhau, sửa các lỗi khi biểu thức chưa đưa ra kết quả chính xác.

Bước 6: Hoàn thiện sản phẩm.

c) Thảo Luận Và Kết Quả Của Sản Phẩm

- Kết quả đạt được so với lý thuyết nói chung thành công nhưng còn nhiều thiếu sót nhỏ trong quá trình tính toán các biểu thức.
- Các lỗi còn thiếu sót người dùng có thể phản hồi:
 - + Các lỗi trên máy chính
 - + Thiếu sót một vài chức năng nào đó

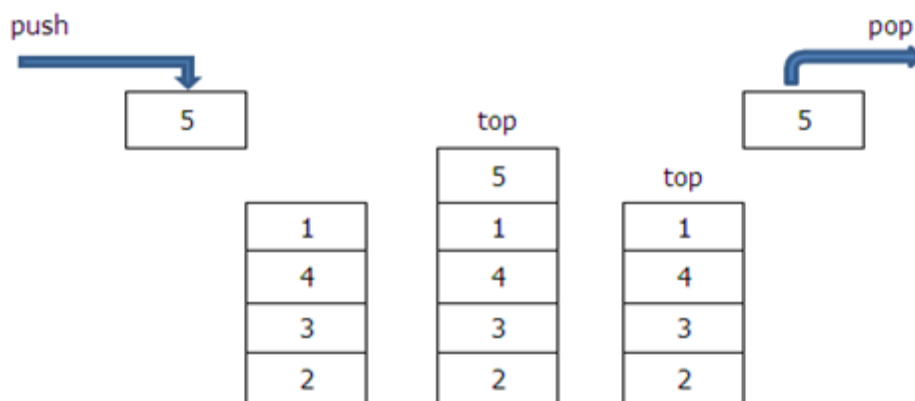
Bảng Thống Kê Các Chức Năng

Chức Năng	Lý Thuyết	Kết Quả Đạt Được
Cộng (+) , trừ (-), nhân(*), chia(/), mũ(^)	Có	Có
Sqrt()	Có	Có
Sin()	Có	Có
Cos()	Có	Có
Exp()	Có	Có
Ln()	Có	Có
Abs()	Có	Có

II. XÂY DỰNG LỚP STACK

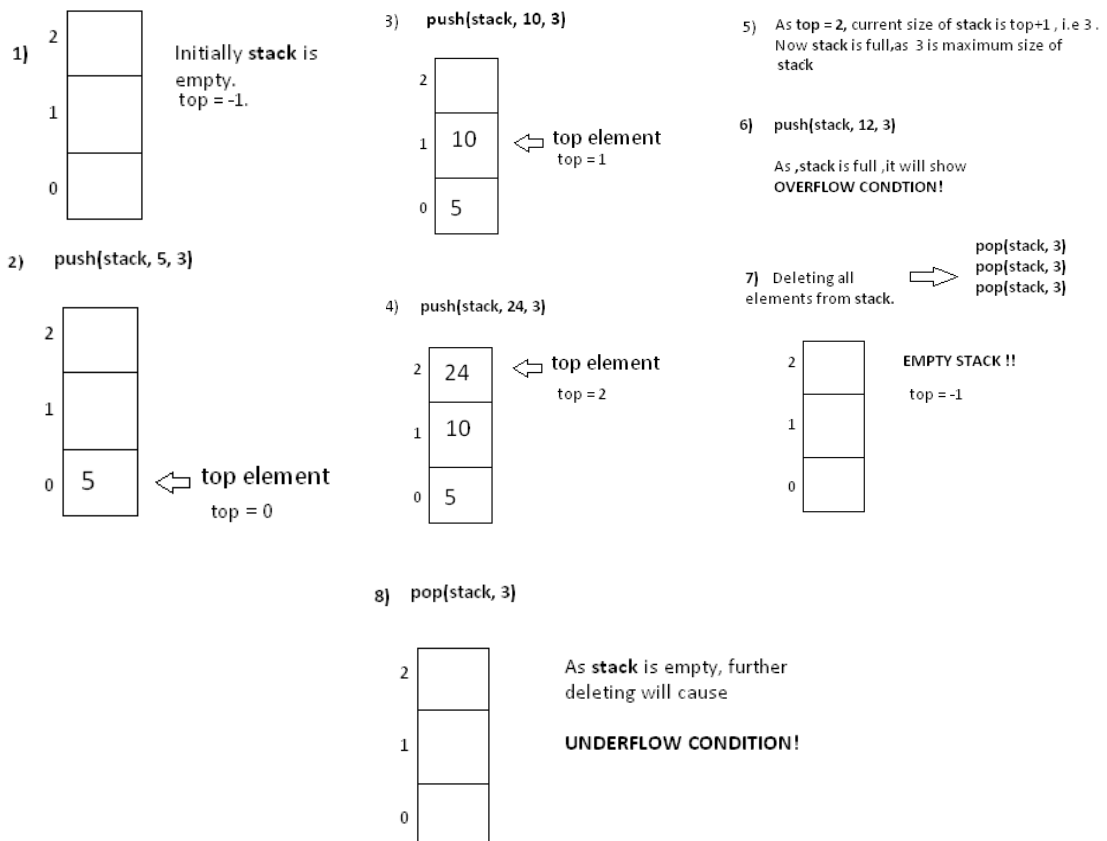
a) Stack

- Stack là một kiểu danh sách tuyến tính đặc biệt mà phép bổ sung và phép loại bỏ luôn luôn được thực hiện ở một đầu (gọi là đỉnh). Hay còn có một định nghĩa khác: Ngăn xếp Stack là một cấu trúc dữ liệu trừu tượng làm việc theo nguyên lý vào sau ra trước **LIFO** (last in first out).



b) Mô tả quá trình xây dựng lớp STACK

STT	Hàm	Diễn Giải
1	IsEmpty	- Kiểm tra stack có rỗng hay không. Khi stack rỗng là stack không chứa phần tử nào, tức là gán $top = -1$ để đánh dấu. Vậy ta so sánh giá trị top để kiểm tra.
2	Push	- <i>Push</i> có nghĩa là thêm phần tử vào đỉnh stack khi rỗng. Ngược lại, nếu đã đầy ta sẽ thông báo stack đã không thể thêm phần tử. Tăng top để gán phần tử cần thêm vào ($stack[++top] = value$).
3	Pop	- <i>Pop</i> có nghĩa là xóa phần tử khỏi đỉnh của stack khi đầy. Ngược lại, nếu đã rỗng ta sẽ thông báo stack không thể xóa phần tử vì chúng đang rỗng không chứa bất kì phần tử nào. Giảm top để xóa phần tử khỏi đỉnh ($--top$).
4	Top	- <i>Top</i> có nghĩa là lấy phần tử ở đỉnh của stack. Trả về phần tử tại vị trí top ($stack[top]$).
5	Size	- <i>Size</i> có nghĩa là trả về số lượng trong stack. Lúc đầu, $top = -1$ vậy để trả về <i>Size</i> ta cần tăng top lên một ($top + 1$).

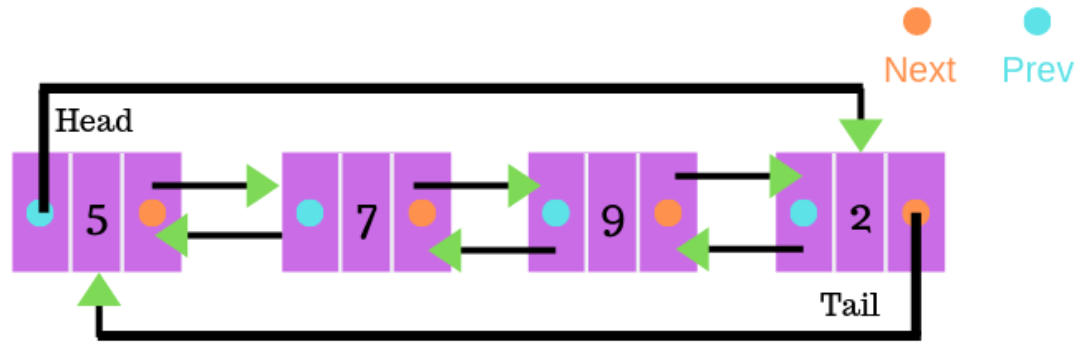


Hình 1.b.ii: Quá trình hoạt động STACK

III. XÂY DỰNG CẤU TRÚC DỮ LIỆU

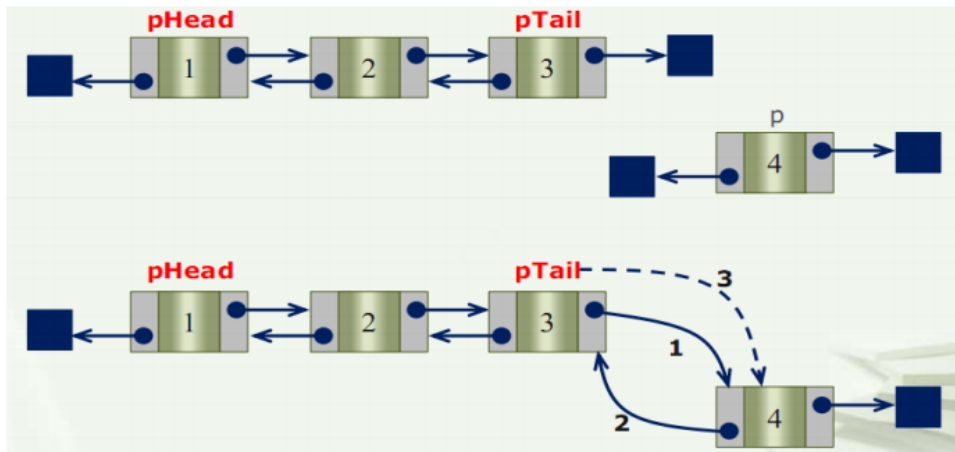
a) Danh Sách Liên Kết Đôi

- Danh sách liên kết đôi (**Doubly linked list**) là danh sách liên kết mà mỗi phần tử có hai liên kết đến phần tử liên trước và liên sau nó. Khi duyệt các nút sẽ thực hiện theo hai chiều về trước và về sau thay vì thực hiện duyệt một chiều như danh sách liên kết đơn.

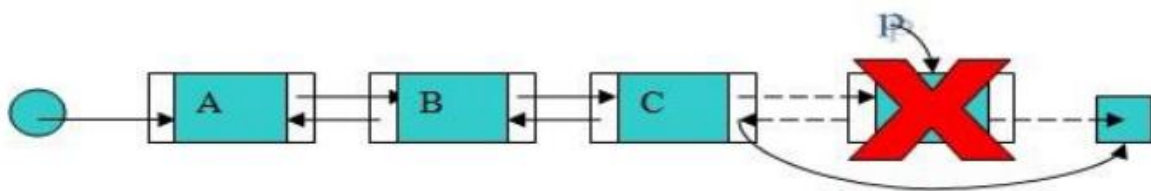


b) Mô Tả Quá Trình Xây Dựng Cấu Trúc Dữ Liệu

STT	Hàm	Diễn Giải
1	Node	- Khởi tạo <i>Node</i> không chỉ có con trỏ <i>next</i> mà còn có con trỏ <i>prev</i> để liên kết <i>Node</i> trước nó. Và khi tạo một <i>Node</i> mới thì không những chỉ <i>next</i> = <i>NULL</i> mà cả <i>prev</i> = <i>NULL</i> và gán <i>value</i> vào.
2	IsEmpty	- <i>IsEmpty</i> kiểm tra <i>tail</i> (điểm cuối) có trả về giá trị <i>NULL</i> .
3	Add	- <i>Add</i> có nghĩa là thêm <i>Node</i> vào cuối. Kiểm tra <i>IsEmpty</i> nếu trả về <i>NULL</i> thì trả <i>head, tail</i> = <i>node</i> (con trỏ <i>node</i> được tạo ra có kiểu dữ liệu là <i>Node</i>), nếu không trả về <i>NULL</i> thì ta tạo liên kết <i>tail</i> hiện tại với <i>tail</i> mới.
4	RemoveTail	- <i>RemoveTail</i> có nghĩa là xóa dữ liệu ở cuối (<i>tail</i>). Kiểm tra <i>IsEmpty</i> nếu trả về <i>NULL</i> thì không chứa dữ liệu vào cần xóa, nếu không trả về <i>NULL</i> ta cho lùi <i>tail</i> và gán <i>next</i> của <i>tail</i> = <i>NULL</i> .
5	GetTail, GetHead	- <i>GetHead</i> là trả giá trị đầu và <i>GetTail</i> trả giá trị cuối.



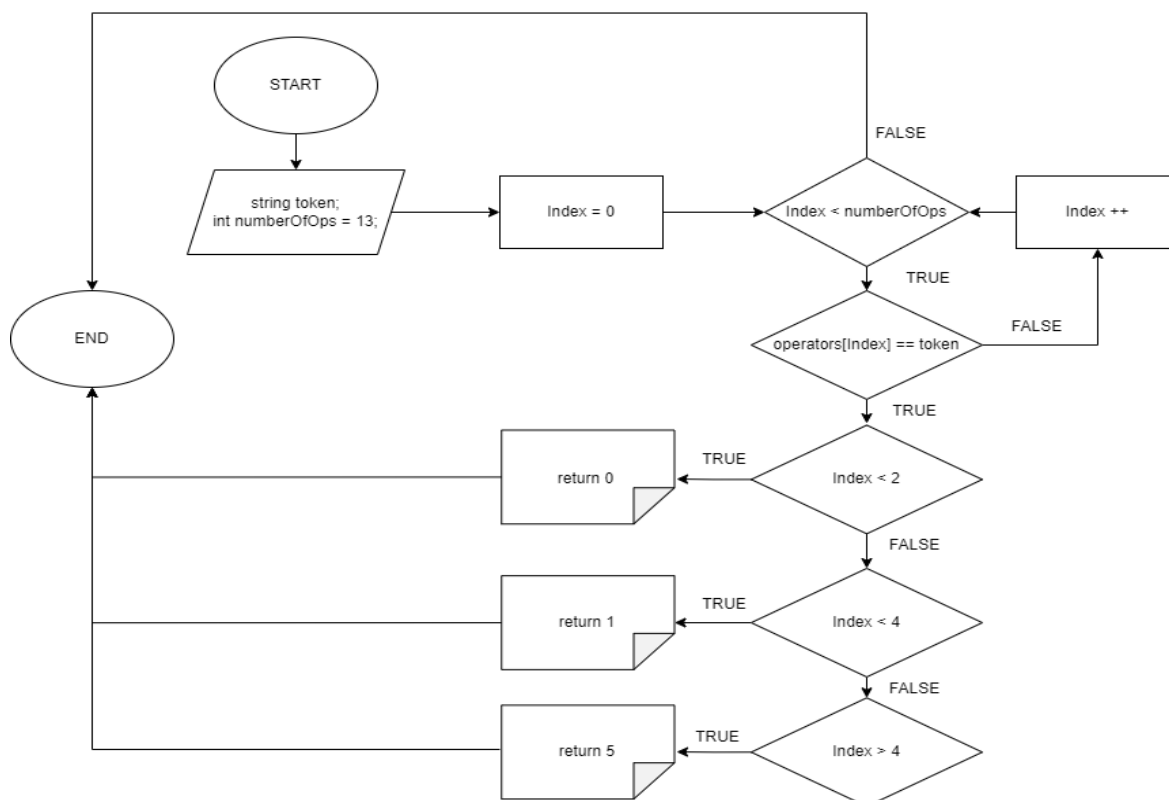
Hình 1.b.iii: Quá trình hoạt động thêm vào tail



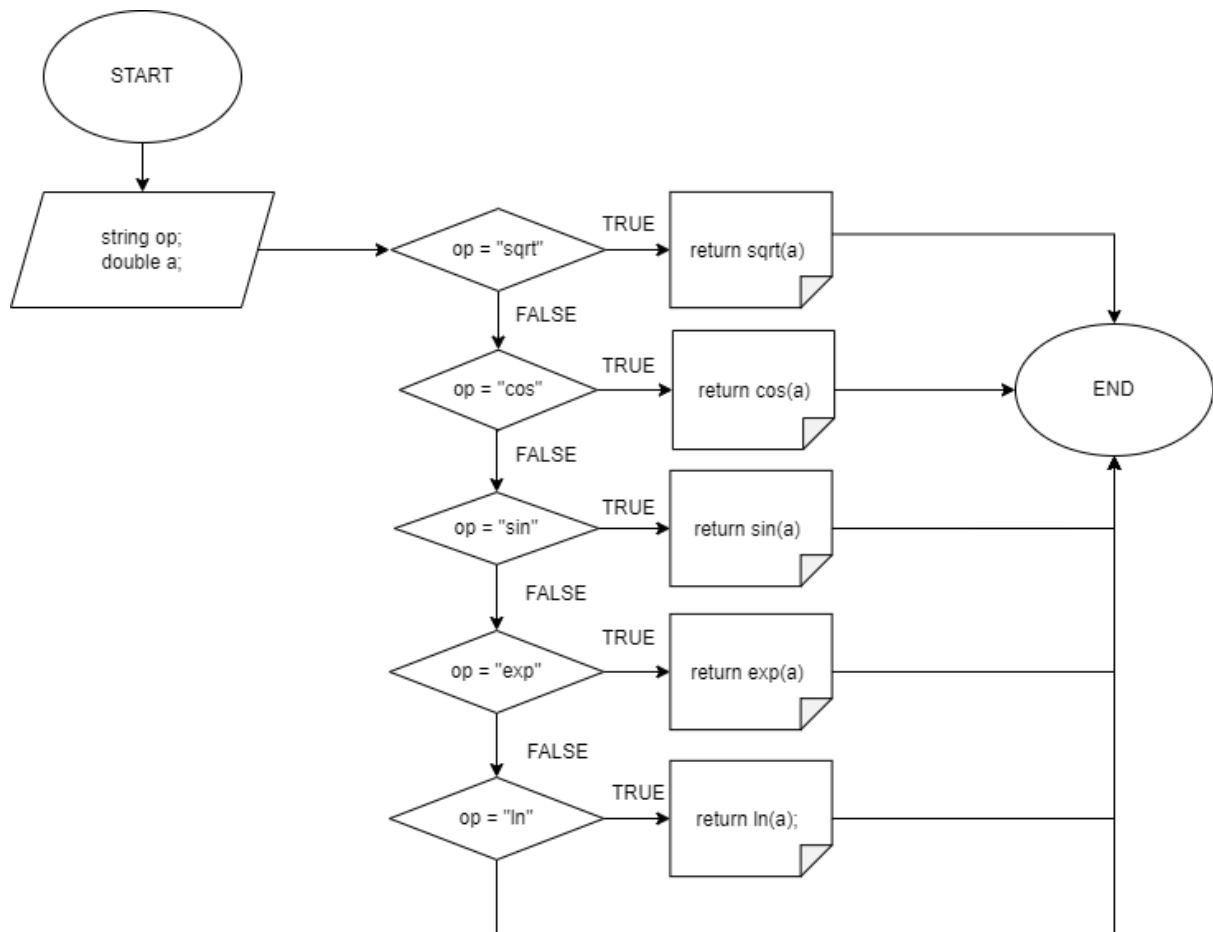
Hình 2.b.iii: Quá trình hoạt động xóa ở tail

IV. XÂY DỰNG CALCULATOR

a) Thuật Toán Trong CALCULATOR

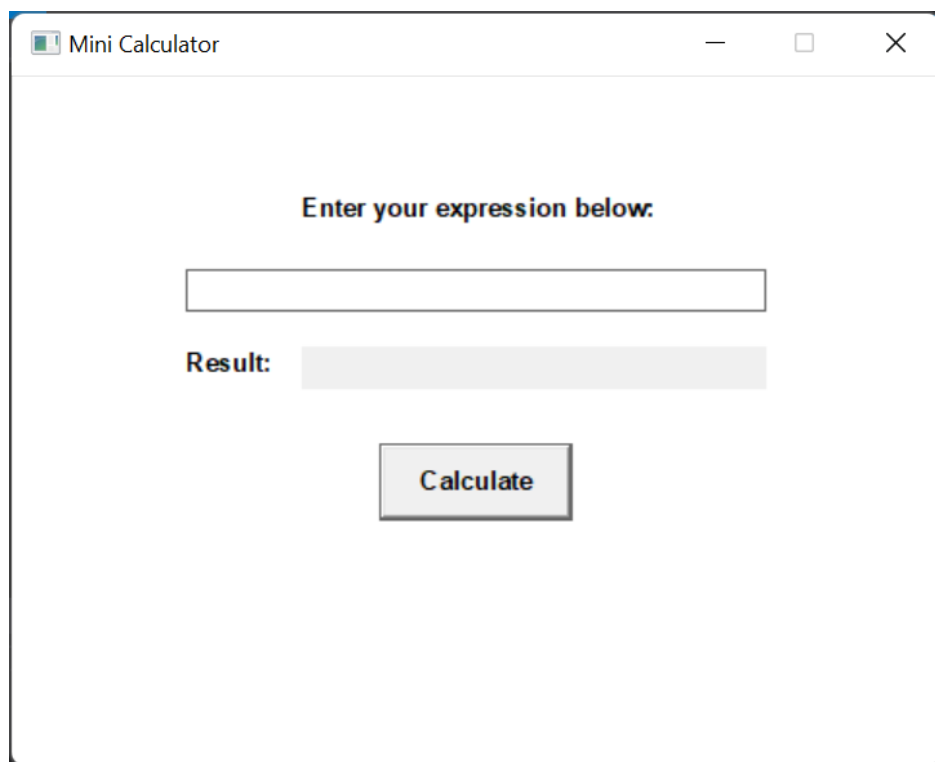


Hình 1.a.IV: Lưu đồ xếp ưu tiên trong các toán tử toán học

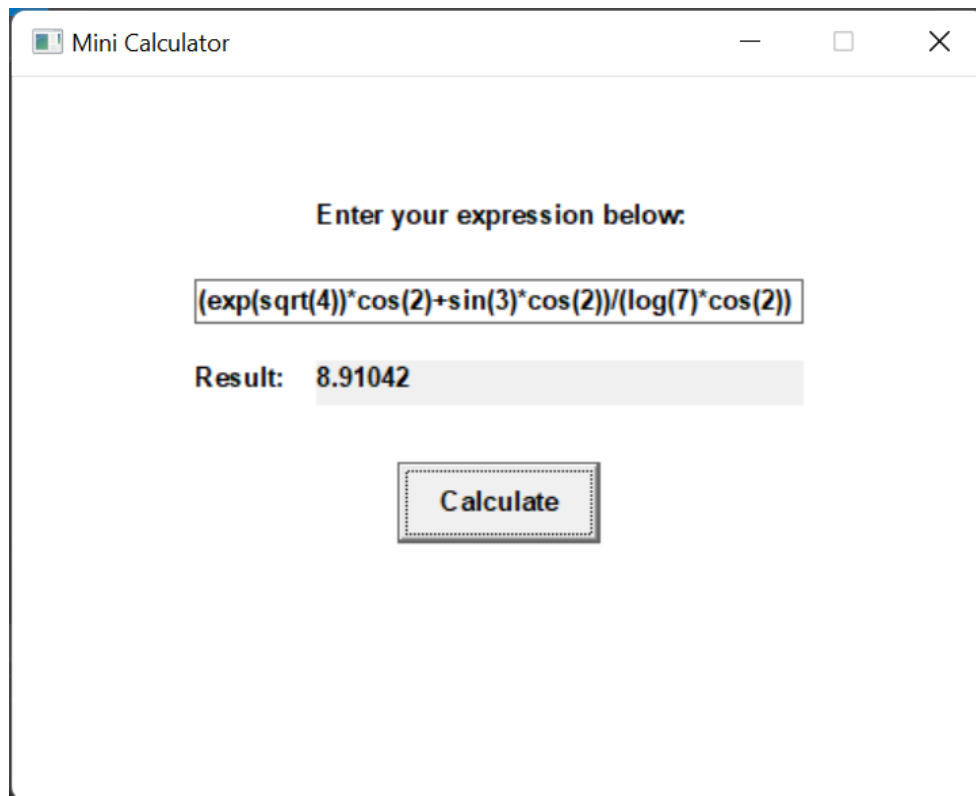


Hình 2.a.IV: Lưu đồ thuật toán gán vai trò cho các toán tử toán học

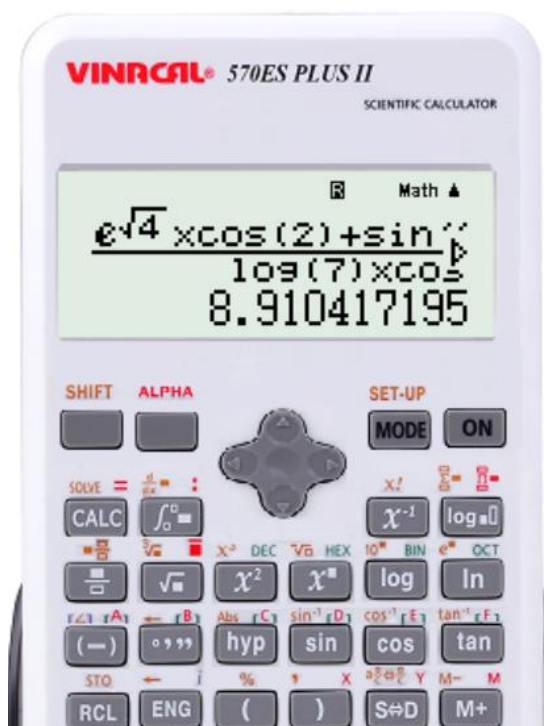
b. Kết Quả



Hình 1.b.IV: Màn hình hiển thị nhập biểu thức



Hình 2.b.IV: Nhập ví dụ biểu thức và ra kết quả.



Hình 2.b.IV: Kiểm tra lại trên máy tính giả lập.

TÀI LIỆU THAM KHẢO

- [1] Giáo Trình Lập Trình Hướng Đối Tượng – NXB Khoa Học Và Kỹ Thuật
- [2] <https://stackoverflow.com/>
- [3] <https://www.geeksforgeeks.org/stack-set-2-infix-to-postfix/>
- [4] <https://www.geeksforgeeks.org/data-structures/linked-list/>