

**TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP**

**KHOA ĐIỆN TỬ**

**Bộ môn: Công nghệ Thông tin.**

**BÀI TẬP KẾT THÚC MÔN HỌC**

**MÔN HỌC**

**LẬP TRÌNH PYTHON**

**Sinh viên** : Nguyễn Thị Hằng Nga

**Lớp** : 58KTP

**Giáo viên GIẢNG DẠY:** TS Nguyễn Văn Huy

**Link GitHub** :



**Thái Nguyên – 2025**

## BÀI TẬP KẾT THÚC MÔN HỌC

### MÔN HỌC: LẬP TRÌNH PYTHON

#### BỘ MÔN : CÔNG NGHỆ THÔNG TIN

*Sinh viên* : Nguyễn Thị Hằng Nga

*Lớp* :58KTP

*Ngành* : Kỹ thuật máy tính

*Giáo viên hướng dẫn*: TS Nguyễn Văn Huy

*Ngày giao đề* :20/05/2025 *Ngày hoàn thành* : 07/06/2025

*Tên đề tài* : Máy tính đơn giản (Simple Calculator GUI)

*Yêu cầu* :

#### **Đầu vào – đầu ra:**

Đầu vào: Hai số thực (qua ô Entry), và lựa chọn phép toán (qua nút hoặc menu Radio).

Đầu ra: Kết quả tính trên GUI (Label).

#### **Tính năng yêu cầu:**

Nhập số, kiểm tra lỗi (không phải số, chia 0).

Cập nhật kết quả ngay khi nhấn nút “Tính”.

Cho phép reset (xóa cả 2 ô nhập).

Bất ngoại lệ với hộp thoại thông báo khi lỗi .

**GIÁO VIÊN HƯỚNG DẪN**

*(Ký và ghi rõ họ tên)*

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

.....

.....

.....

*Thái Nguyên, ngày....tháng.....năm 20....*

**GIÁO VIÊN HƯỚNG DẪN**

*(Ký ghi rõ họ tên)*

## MỤC LỤC

MỤC LỤC.....	2
CHƯƠNG 1 GIỚI THIỆU ĐỀ TÀI.....	4
CHƯƠNG 2 CƠ SỞ LÝ THUYẾT.....	5
2.1. Kiến thức lập trình áp dụng.....	5
2.2. Cấu trúc dữ liệu sử dụng.....	5
CHƯƠNG 3 THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH.....	6
3.1. Sơ đồ khối hệ thống.....	6
3.2. Sơ đồ khối các thuật toán chính.....	7
3.3. Cấu trúc dữ liệu.....	7
3.4. Chương trình.....	7
CHƯƠNG 4 THỰC NGHIỆM VÀ KẾT LUẬN.....	8
4.1. Thực nghiệm.....	8
4.2. Kết luận.....	10
TÀI LIỆU THAM KHẢO.....	12

## LỜI NÓI ĐẦU

Trong thời đại công nghệ phát triển mạnh mẽ như hiện nay, việc ứng dụng các công cụ tính toán vào đời sống hằng ngày đã trở nên phổ biến và không thể thiếu. Một trong những công cụ cơ bản nhưng thiết yếu đó chính là máy tính – thiết bị giúp con người thực hiện các phép toán một cách nhanh chóng, chính xác và hiệu quả.

Đề tài “**Máy tính đơn giản (Simple Calculator GUI)**” được thực hiện với mục tiêu xây dựng một ứng dụng có giao diện trực quan, dễ sử dụng, hỗ trợ người dùng thực hiện các phép toán cơ bản như cộng, trừ, nhân, chia. Thông qua đề tài này, người thực hiện không chỉ áp dụng kiến thức đã học về lập trình giao diện đồ họa (GUI), mà còn rèn luyện kỹ năng tư duy logic, thiết kế phần mềm và giải quyết vấn đề trong thực tiễn.

Đây là một đề tài tuy nhỏ, nhưng mang tính ứng dụng cao và phù hợp với người mới bắt đầu tiếp cận lập trình giao diện. Hy vọng sản phẩm hoàn thiện có thể trở thành nền tảng để phát triển các ứng dụng tính toán phức tạp hơn trong tương lai.

Xin chân thành cảm ơn quý thầy cô đã tận tình hướng dẫn và tạo điều kiện thuận lợi để em hoàn thành đề tài này.

## CHƯƠNG 1 GIỚI THIỆU ĐỀ TÀI

### 1.1 Tên đề tài : Máy tính đơn giản (Simple Calculator GUI)

#### Đầu bài:

Viết chương trình máy tính có giao diện GUI cho phép người dùng nhập hai số và chọn phép toán  $+$ ,  $-$ ,  $\times$ ,  $\div$  để tính toán.

#### Đầu vào – đầu ra:

- Đầu vào: Hai số thực (qua ô Entry), và lựa chọn phép toán (qua nút hoặc menu Radio).
- Đầu ra: Kết quả tính trên GUI (Label).

#### Tính năng yêu cầu:

- Nhập số, kiểm tra lỗi (không phải số, chia 0).
- Cập nhật kết quả ngay khi nhấn nút “Tính”.
- Cho phép reset (xóa cả 2 ô nhập).
- Bắt ngoại lệ với hộp thoại thông báo khi lỗi .

#### Kiểm tra & kết quả mẫu:

- Nhập 3.5 và 2, chọn “ $\times$ ”  $\rightarrow$  Kết quả: 7.0
- Nhập “a” và 1  $\rightarrow$  Hộp thoại: “Vui lòng nhập số hợp lệ.”

### 1.2 Tính năng và thách thức , kiến thức vận dụng

#### Tính năng

Thực hiện phép cộng, trừ, nhân, chia.

Giao diện đồ họa đơn giản, dễ dùng.

Hỗ trợ nhập liệu, xóa, hiển thị kết quả.

#### Thách thức

Thiết kế giao diện trực quan.

Xử lý logic tính toán và lỗi (chia 0, sai cú pháp).

Quản lý trạng thái và sự kiện người dùng.

### **Kiến thức vận dụng**

Lập trình GUI (Tkinter, JavaFX, hoặc HTML/CSS/JS).

Xử lý sự kiện và logic biểu thức.

Tư duy lập trình và kiểm tra lỗi cơ bản.

## **CHƯƠNG 2 CƠ SỞ LÝ THUYẾT**

### **2.1. Kiến thức lập trình áp dụng**

#### **Biến và kiểu dữ liệu:**

Dùng để lưu giá trị số, phép toán, kết quả. Sử dụng các kiểu như int, float, str.

#### **Câu lệnh điều kiện (if, else)**

Dùng để kiểm tra loại nút (số, toán tử), xử lý lỗi như chia cho 0.

#### **Hàm (Function):**

Phân tách các chức năng như tính toán, xử lý nút, xóa, hiển thị kết quả.

#### **Xử lý chuỗi (String):**

Dùng để tạo biểu thức, cập nhật màn hình hiển thị theo thao tác người dùng.

#### **Xử lý sự kiện (Event):**

Bắt sự kiện nhấn nút hoặc bàn phím và gọi hàm tương ứng.

### **2.2. Cấu trúc dữ liệu sử dụng**

#### **Danh sách (List):**

Lưu các phần tử của biểu thức toán học (số, toán tử).

Duyệt và xử lý biểu thức từng bước.

#### **Từ điển (Dict):**

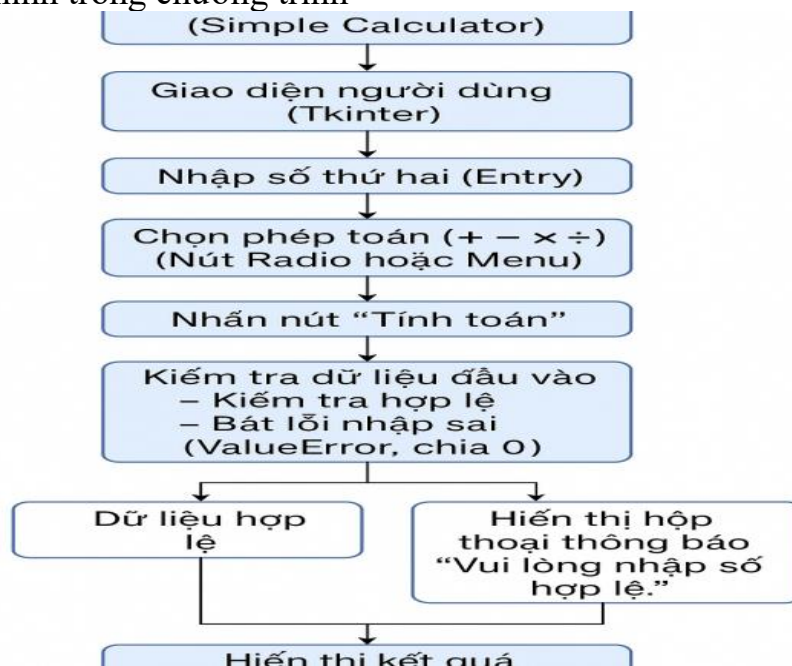
Ánh xạ toán tử với hàm tính toán tương ứng (ví dụ: '+' → add).

Hỗ trợ xử lý nhanh và linh hoạt hơn thay vì dùng nhiều if.

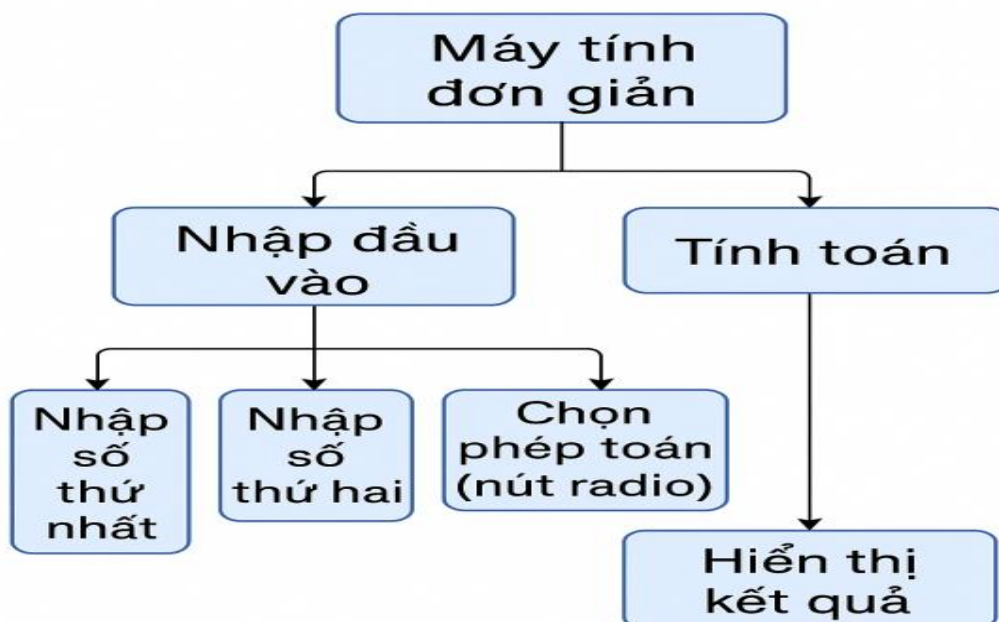
## CHƯƠNG 3 THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH

### 3.1. Sơ đồ khối hệ thống

Các module chính trong chương trình



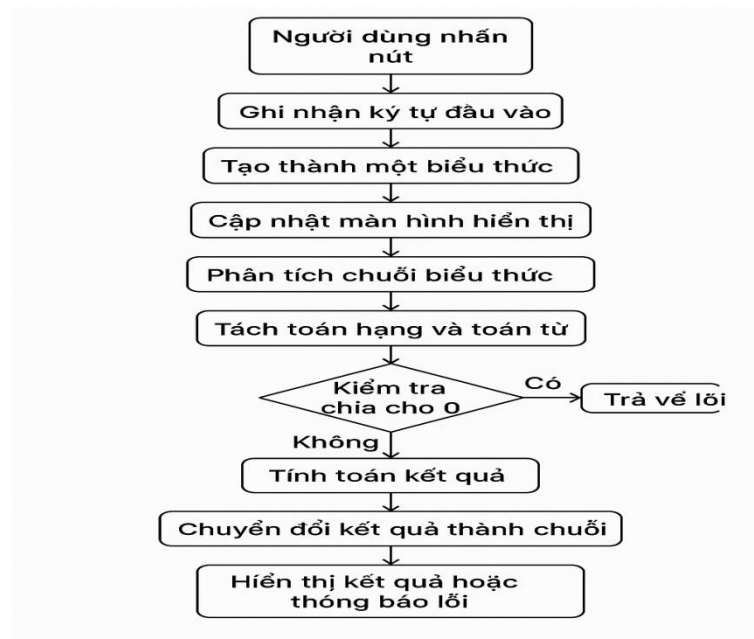
Hình 1 Các module chính



Hình 2 Biểu đồ chức năng



### 3.2. Sơ đồ khối các thuật toán chính



Hình 3 Sơ đồ các thuật toán chính

### 3.3. Cấu trúc dữ liệu

Do đề tài là một ứng dụng máy tính đơn giản chủ yếu xử lý dữ liệu đầu vào là hai số thực và phép toán, nên không sử dụng cơ sở dữ liệu hay bảng dữ liệu phức tạp. Các dữ liệu lưu trữ và xử lý trong chương trình gồm:

**Biến num1 (float):** lưu giá trị số thực nhập từ ô Entry đầu tiên.

**Biến num2 (float):** lưu giá trị số thực nhập từ ô Entry thứ hai.

**Biến operation (string):** lưu phép toán được chọn, có thể là "+", "-", "\*", hoặc "/".

**Biến result (float):** lưu kết quả phép tính giữa num1 và num2.

**Các biến hỗ trợ giao diện:** như biến liên kết với widget Entry, RadioButton hoặc Button để lấy dữ liệu đầu vào và điều khiển giao diện.

### 3.4. Chương trình

Chương trình được xây dựng bằng Python với Tkinter, gồm các hàm chính sau:

**\*) calculate()**

- Chức năng: Lấy dữ liệu từ các ô nhập và phép toán đã chọn, kiểm tra hợp lệ, thực hiện phép tính và cập nhật kết quả lên Label.

- Xử lý lỗi: Nếu dữ liệu không phải số thực hoặc phép chia cho 0, hiển thị hộp thoại cảnh báo lỗi.

#### **\*) reset()**

- Chức năng: Xóa dữ liệu trong hai ô Entry và xóa kết quả hiển thị

#### **\*) validate\_input(value) (nếu có)**

- Chức năng: Kiểm tra xem chuỗi nhập có phải là số thực hợp lệ hay không (có thể kết hợp với xử lý ngoại lệ).

**\*)Hàm khởi tạo giao diện:** Tạo các widget Entry cho số nhập, RadioButton hoặc nút để chọn phép toán, nút "Tính" để gọi calculate(), nút "Reset" để gọi reset(), và Label để hiển thị kết quả.

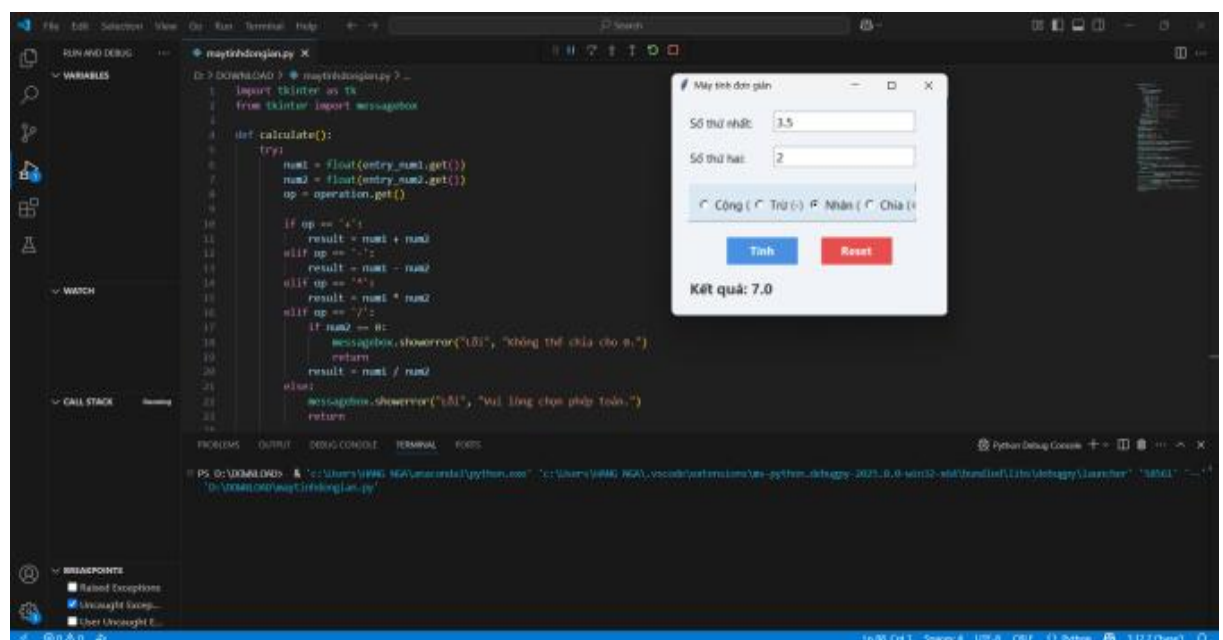
## CHƯƠNG 4 THỰC NGHIỆM VÀ KẾT LUẬN

### 4.1. Thực nghiệm

#### - Test 1: Phép nhân với số thực

Nhập 3.5 và 2 vào hai ô Entry, chọn phép nhân (×), nhấn nút “Tính”.

**Kết quả:** Label hiển thị kết quả 7.0, đúng như mong đợi.

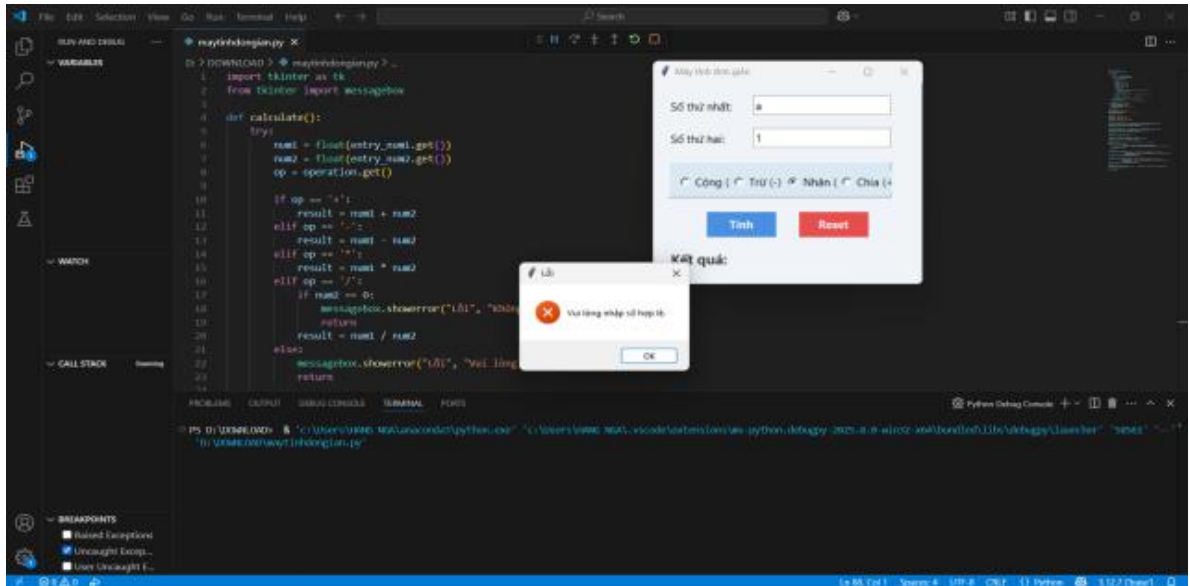


Hình 4 Kết quả test 1

### - Test 2: Nhập dữ liệu không hợp lệ

Nhập a và 1 vào hai ô Entry, chọn phép cộng (+), nhấn nút “Tính”.

**Kết quả:** Hiện thị hộp thoại cảnh báo: “Vui lòng nhập số hợp lệ.”

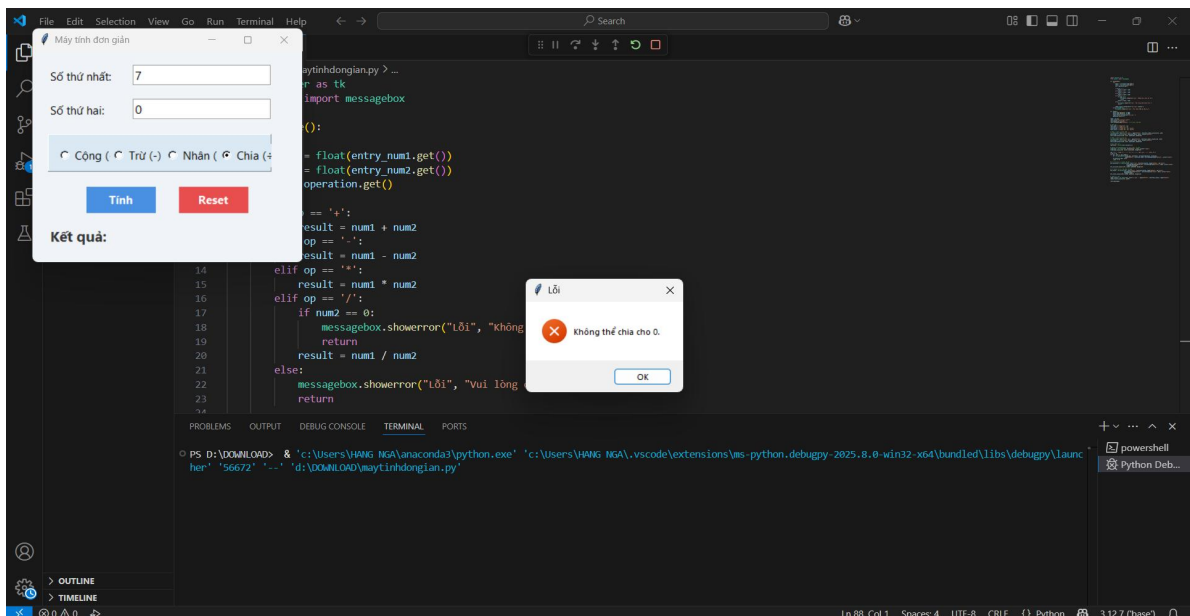


Hình 5 Kết quả test 2

### - Test 3: Thực hiện chia cho 0

Nhập 7 và 0 vào hai ô Entry, chọn phép chia (/), nhấn nút “Tính”.

**Kết quả:** Hiện thị hộp thoại cảnh báo: “không thể chia cho 0”



Hình 6 Kết quả test 3

## 4.2. Kết luận

- **Kết quả đạt được:**

Qua quá trình xây dựng và hoàn thiện, đề tài đã thực hiện thành công một ứng dụng máy tính đơn giản sử dụng ngôn ngữ lập trình Python kết hợp với thư viện giao diện Tkinter. Ứng dụng cho phép người dùng nhập hai số thực và lựa chọn thực hiện các phép toán số học cơ bản bao gồm: cộng, trừ, nhân và chia.

Giao diện người dùng được thiết kế trực quan, dễ sử dụng, thân thiện với người dùng kể cả những người không có kiến thức chuyên môn về lập trình. Một điểm nổi bật của ứng dụng là khả năng xử lý lỗi hiệu quả: các trường hợp nhập sai định dạng, để trống ô dữ liệu hoặc thực hiện phép chia cho 0 đều được hệ thống phát hiện và phản hồi bằng các hộp thoại cảnh báo rõ ràng, giúp người dùng dễ dàng nhận biết và sửa lỗi.

Sau khi nhấn nút “Tính”, kết quả được hiển thị một cách nhanh chóng, giúp người dùng có phản hồi tức thì. Ngoài ra, chức năng “Reset” cũng được tích hợp nhằm hỗ trợ người dùng xóa toàn bộ dữ liệu đầu vào và kết quả chỉ với một thao tác đơn giản, từ đó cải thiện trải nghiệm sử dụng và tăng tính tiện lợi cho ứng dụng.

- **Kiến thức và kỹ năng thu nhận được:**

Trong quá trình thực hiện đề tài, người làm đã có cơ hội tiếp cận và thực hành trực tiếp với lập trình giao diện đồ họa thông qua thư viện Tkinter – một thư viện GUI phổ biến và mạnh mẽ trong Python. Việc xây dựng ứng dụng từ khâu thiết kế giao diện đến xử lý logic đã giúp củng cố và nâng cao kỹ năng lập trình tổng hợp, đặc biệt là tư duy xử lý ngoại lệ và kiểm tra tính hợp lệ của dữ liệu đầu vào – một yếu tố rất quan trọng trong các phần mềm có tương tác người dùng.

Bên cạnh đó, người làm còn học được cách tổ chức mã nguồn một cách khoa học, phân chia chức năng rõ ràng và xây dựng luồng xử lý hợp lý, tạo nền tảng vững chắc cho việc phát triển các ứng dụng có quy mô lớn hơn trong tương lai.

- **Hướng phát triển trong tương lai:**

Mặc dù ứng dụng đã đáp ứng được các yêu cầu cơ bản đặt ra ban đầu, tuy nhiên vẫn còn nhiều tiềm năng để mở rộng và nâng cấp. Trong tương lai, ứng dụng có thể được cải tiến theo một số hướng sau:

- Bổ sung thêm các phép toán nâng cao như lũy thừa, căn bậc hai, tính phần trăm, hoặc các hàm lượng giác để phục vụ nhu cầu tính toán đa dạng hơn.

- Hỗ trợ tính toán với nhiều số thay vì giới hạn trong hai số như hiện tại, hoặc cho phép người dùng nhập biểu thức đầy đủ thay vì chọn phép toán thủ công.
- Tích hợp chức năng lưu lại lịch sử các phép tính để người dùng có thể theo dõi, kiểm tra hoặc sử dụng lại kết quả trước đó.
- Cải thiện giao diện người dùng bằng cách sử dụng thêm các thư viện nâng cao như ttk, customtkinter hoặc thậm chí là tích hợp với thư viện đồ họa như PyQt, từ đó tăng tính chuyên nghiệp và hiện đại cho sản phẩm.
- Hỗ trợ đa nền tảng, đóng gói ứng dụng dưới dạng tệp thực thi để người dùng có thể cài đặt và sử dụng trực tiếp mà không cần cài đặt môi trường lập trình.

Tóm lại, qua đề tài này, người làm không chỉ hoàn thiện được một sản phẩm cụ thể mà còn tích lũy được nhiều kinh nghiệm quý báu về phát triển ứng dụng thực tế, mở rộng hiểu biết về GUI trong Python, và đặt nền móng cho các dự án phần mềm sau này

## TÀI LIỆU THAM KHẢO

1. [Hướng dẫn lập trình giao diện GUI bằng Tkinter trong ngôn ngữ Python - VniTeach - Giáo viên 4.0](#)
2. [Giao diện GUI là gì? Tìm hiểu chi tiết các thành phần và tính năng nổi bật](#)

## LINK YOUTUBE

