

**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 LINH**

**LỚP : K58KTP**

**GIÁO VIÊN GIẢNG DẠY : TS.NGUYỄN VĂN HUY**

**Link GitHub :**

TRƯỜNG ĐHKTCN

CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM

KHOA ĐIỆN TỬ

Độc lập - Tự do - Hạnh phúc

## BÀI TẬP KẾT THÚC MÔN HỌC

**MÔN HỌC: LẬP TRÌNH PYTHON**

**BỘ MÔN: CÔNG NGHỆ THÔNG TIN**

**Sinh viên:** Nguyễn Thị Linh

**MSSV:** K225480106040

**Lớp:** K58KTP.K01

**Ngành:** Kỹ thuật máy tính

**Giáo viên hướng dẫn:** TS. Nguyễn Văn Huy

**Ngày giao đề:** 20/05/2025

**Ngày hoàn thành:** 10/06/2025

**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).

**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

**GIÁO VIÊN HƯỚNG DẪN**

*(Ký và ghi rõ họ tên)*

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

.....

.....

.....

.....

Xếp loại : ..... Điểm : .....

Thái Nguyên, ngày 23 tháng 4 năm 2025.

**GIÁO VIÊN HƯỚNG DẪN**

*(Ký và ghi rõ họ tên)*

## **LỜI CAM ĐOAN**

Em xin cam đoan bài báo cáo với đề tài “Máy tính đơn giản (Simple Calculator GUI)” là kết quả của quá trình học tập, nghiên cứu và thực hiện của cá nhân em dưới sự hướng dẫn của Thầy Nguyễn Văn Huy trong học phần Lập trình Python.

Toàn bộ nội dung, mã nguồn và kết quả trong báo cáo đều do em tự tìm hiểu và xây dựng, không sao chép từ bất kỳ nguồn nào mà không trích dẫn rõ ràng. Nếu có bất kỳ sai sót hoặc vi phạm nào liên quan đến bản quyền hoặc nội dung trong báo cáo, em xin hoàn toàn chịu trách nhiệm trước nhà trường và giáo viên hướng dẫn.

**Sinh viên thực hiện**

*( Ký và ghi rõ họ tên)*

## MỤC LỤC

LỜI CAM ĐOAN .....	3
MỤC LỤC.....	4
LỜI MỞ ĐẦU .....	6
CHƯƠNG 1.GIỚI THIỆU ĐẦU BÀI.....	7
1.1.Đề tài .....	7
1.2.Tính năng chính của chương trình .....	7
1.3.Thách thức khi thực hiện.....	7
1.4.Kiến thức được vận dụng .....	7
CHƯƠNG 2.CƠ SỞ LÝ THUYẾT .....	9
2.1. Ngôn ngữ Python.....	9
2.1.1. Giới thiệu.....	9
2.2. Thư viện Tkinter.....	9
2.2.1.Tkinter là gì? .....	9
2.3. Xử lý dữ liệu và ngoại lệ trong Python .....	10
2.4. Cơ chế xử lý sự kiện trong GUI.....	10
2.5.Quy trình hoạt động tổng quát của chương trình .....	11
CHƯƠNG 3. THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH .....	12
3.1. Sơ đồ khối hệ thống.....	12
3.1.1.Biểu đồ phân cấp chức năng .....	12
3.2.Sơ đồ các thuật toán chính .....	13
3.3. Cấu trúc dữ liệu .....	15

3.4. Chương trình .....	15
CHƯƠNG 4: THỰC NGHIỆM VÀ KẾT LUẬN .....	17
4.1.Thực nghiệm.....	17
4.2.Kết luận .....	19

## LỜI MỞ ĐẦU

Trong bối cảnh công nghệ thông tin không ngừng bùng nổ, lập trình đã trở thành kỹ năng cốt lõi đối với sinh viên ngành Kỹ thuật Máy tính. Ngôn ngữ Python – với cú pháp súc tích, dễ tiếp cận và hệ sinh thái thư viện phong phú – tạo điều kiện lý tưởng cho việc hiện thực hóa các ứng dụng tương tác. Trong số đó, tkinter là thư viện giao diện đồ họa tích hợp sẵn, cho phép lập trình viên nhanh chóng xây dựng những phần mềm trực quan và thân thiện với người dùng.

Được sự hướng dẫn tận tình của thầy Nguyễn Văn Huy trong học phần Lập trình Python, em lựa chọn thực hiện đề tài “Máy tính đơn giản (Simple Calculator GUI)” nhằm áp dụng kiến thức đã học vào một sản phẩm cụ thể. Thông qua đề tài này, em có cơ hội củng cố lại các khái niệm lập trình hướng đối tượng, xử lý sự kiện, thiết kế giao diện đồ họa cũng như cách tổ chức mã nguồn rõ ràng, logic.

Chương trình máy tính đơn giản cho phép người dùng nhập hai số thực, lựa chọn một trong bốn phép toán cơ bản (+, -, ×, ÷) và nhận kết quả ngay trên giao diện. Ứng dụng đồng thời xử lý các ngoại lệ thường gặp như nhập sai định dạng hay phép chia cho 0, đảm bảo tính ổn định và thân thiện trong quá trình sử dụng.

Đề tài không chỉ giúp em hiểu sâu hơn về Python và thư viện tkinter, mà còn rèn luyện kỹ năng phân tích, thiết kế và giải quyết vấn đề – những năng lực quan trọng cho hành trang nghề nghiệp tương lai. Mặc dù sản phẩm vẫn còn một số hạn chế, kết quả đạt được đã phản ánh nỗ lực học tập và tinh thần sáng tạo của bản thân.

Em xin chân thành cảm ơn thầy Nguyễn Văn Huy đã tận tâm hướng dẫn, định hướng và tạo điều kiện thuận lợi để em hoàn thành đề tài này. Em rất mong nhận được những góp ý quý báu để tiếp tục hoàn thiện và nâng cao chất lượng sản phẩm trong thời gian tới.



## **CHƯƠNG 1. GIỚI THIỆU ĐẦU BÀI**

### **1.1. Đề tài**

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.

Trong bài tập này, sinh viên được yêu cầu xây dựng một chương trình máy tính đơn giản (Simple Calculator) có giao diện đồ họa (GUI) bằng ngôn ngữ Python. Chương trình cho phép người dùng nhập hai số thực thông qua các ô nhập liệu (Entry), sau đó chọn phép toán ( $+$ ,  $-$ ,  $\times$ ,  $\div$ ) bằng cách sử dụng nút chọn (radio button hoặc menu), và cuối cùng hiển thị kết quả ngay trên giao diện sau khi nhấn nút “Tính”.

### **1.2. Tính năng chính của chương trình**

Theo đề bài, chương trình yêu cầu các chức năng sau:

- 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.

### **1.3. Thách thức khi thực hiện**

- Xử lý dữ liệu đầu vào: cần đảm bảo người dùng nhập đúng kiểu số thực, nếu sai phải thông báo lỗi rõ ràng.
- Chia cho 0 là một lỗi logic thường gặp cần được xử lý riêng.
- Thiết kế giao diện sao cho trực quan, bố trí hợp lý, dễ thao tác.
- Liên kết giữa giao diện và logic xử lý: yêu cầu hiểu rõ cách hoạt động của các widget trong tkinter.

### **1.4. Kiến thức được vận dụng**

- Lập trình Python cơ bản: biến, hàm, điều kiện, ép kiểu dữ liệu.

- Lập trình giao diện với thư viện tkinter: dùng các thành phần như Tk, Label, Entry, Button, Radiobutton, messagebox.
- Xử lý ngoại lệ (try-except) để bắt các lỗi như nhập sai kiểu (ValueError) hoặc chia cho 0 (ZeroDivisionError).
- Tư duy thiết kế hệ thống đơn giản với giao diện người dùng và xử lý sự kiện.

## **CHƯƠNG 2. CƠ SỞ LÝ THUYẾT**

### **2.1. Ngôn ngữ Python**

#### *2.1.1. Giới thiệu*

Python là một ngôn ngữ lập trình thông dịch (interpreted), đa mục đích, hướng đối tượng (object-oriented) và bậc cao (high-level) với ngữ nghĩa động (dynamic semantics). Python hỗ trợ khái niệm module và package, khuyến khích tái sử dụng mã và phát triển theo hướng mô-đun. Bộ thông dịch cùng thư viện chuẩn được phát hành miễn phí dưới dạng mã nguồn lẫn nhị phân trên hầu hết các nền tảng phổ biến và có thể phân phối tự do.

Một số đặc điểm nổi bật:

- Kết nối dễ dàng với môi trường khác: tích hợp COM, .NET (IronPython), JVM (Jython), chia sẻ thư viện C/C++ (CPython) hoặc giao tiếp qua ICE, CORBA...
- Đa nền tảng: chạy nhất quán trên Windows, Linux/Unix, macOS, Android, iOS; cũng tồn tại các bản chạy trên .NET, JVM, thiết bị di động.
- Cú pháp đơn giản, dễ học; cộng đồng lớn và hệ sinh thái thư viện phong phú.
- Mã nguồn mở: tuân theo giấy phép của Python Software Foundation, cho phép sử dụng trong cả môi trường thương mại.

### **2.2. Thư viện Tkinter**

#### *2.2.1. Tkinter là gì?*

Tkinter là thư viện tiêu chuẩn đi kèm với Python, hỗ trợ xây dựng các ứng dụng có giao diện đồ họa. Nó giúp tạo ra các thành phần như ô nhập liệu (Entry), nút bấm (Button), nhãn hiển thị (Label), lựa chọn (Radiobutton), hộp thoại thông báo (messagebox)...

Các thành phần thường dùng:

- Tk(): Tạo cửa sổ chính cho ứng dụng.

- Label: Hiển thị văn bản trên giao diện.
- Entry: Nhập dữ liệu từ người dùng.
- Button: Thực hiện hành động khi được nhấn.
- Radiobutton: Cho phép chọn một trong nhiều phép toán.
- `messagebox.showerror()`: Hiển thị thông báo lỗi.

Hệ thống bố cục `grid()` của Tkinter cho phép chia layout thành lưới (row, column), giúp dễ bố trí giao diện.

### 2.3. Xử lý dữ liệu và ngoại lệ trong Python

Khi làm việc với nhập liệu từ người dùng, dữ liệu từ Entry là chuỗi (str). Cần chuyển thành kiểu float để thực hiện phép toán:

```
num1 = float(entry1.get())  
num2 = float(entry2.get())
```

Tuy nhiên, nếu người dùng nhập ký tự không phải số (ví dụ: "abc"), chương trình sẽ báo lỗi. Để tránh điều này, Python cung cấp cơ chế **xử lý ngoại lệ** (try-except) để "bắt lỗi" và xử lý thân thiện hơn.

Các lỗi thường gặp:

- `ValueError`: Khi nhập không phải số.
- `ZeroDivisionError`: Khi chia cho 0.

```
except ValueError:  
    messagebox.showerror("Lỗi đầu vào", "Vui lòng nhập số hợp lệ.")  
    result_var.set("")
```

### 2.4. Cơ chế xử lý sự kiện trong GUI

Trong ứng dụng GUI, người dùng tương tác qua nút, ô nhập liệu, radio,... Các hành động này gọi là “sự kiện” và được xử lý bằng các hàm sự kiện (callback function).

```
button = Button(root, text="Tính", command=calculate)
```

Khi người dùng nhấn nút “Tính”, hàm `calculate()` được gọi để thực hiện các bước:

- Lấy dữ liệu từ các ô nhập (Entry)
- Kiểm tra phép toán được chọn
- Tính toán và hiển thị kết quả
- Thông báo lỗi nếu có

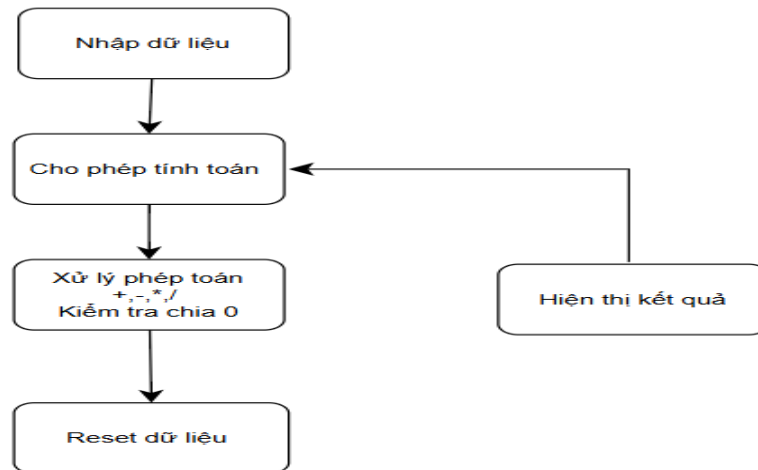
## **2.5. Quy trình hoạt động tổng quát của chương trình**

- Người dùng nhập số vào hai ô Entry
- Chọn phép toán (+, −, ×, ÷) bằng Radiobutton
- Nhấn nút “Tính” → gọi hàm calculate()
- Hàm xử lý:
  - Chuyển dữ liệu sang số
  - Thực hiện phép tính
  - Hiển thị kết quả lên Label
  - Nếu lỗi: hiển thị thông báo bằng messagebox
- Nút “Reset” sẽ xóa dữ liệu hai ô và kết quả.

## CHƯƠNG 3. THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH

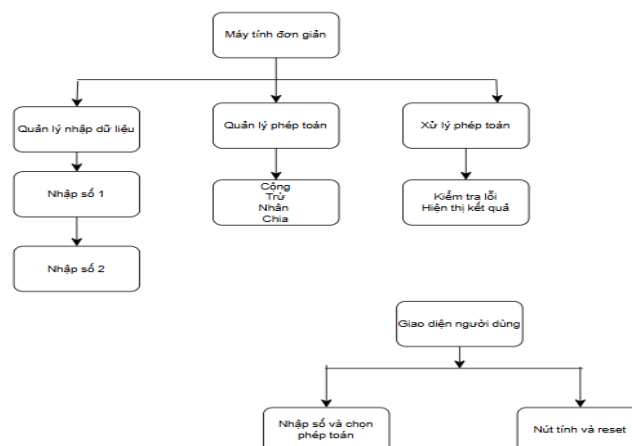
### 3.1. Sơ đồ khối hệ thống

Dưới đây là sơ đồ khối mô tả các module chính trong chương trình:



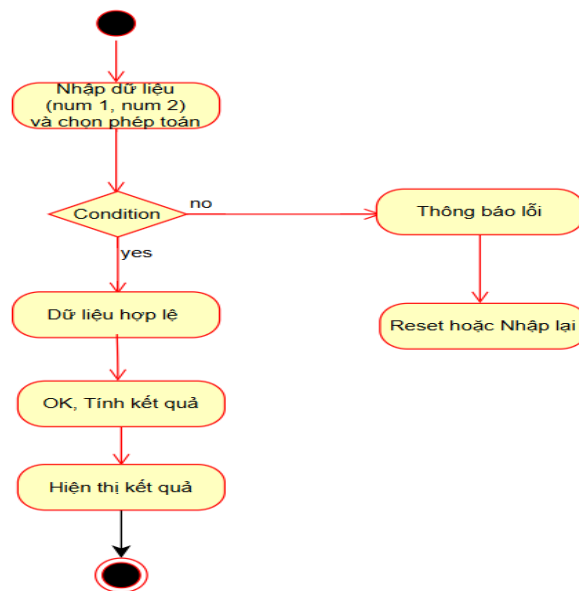
- Nhập dữ liệu: Nhận số từ người dùng qua các ô nhập.
- Cho phép tính toán: Lựa chọn phép tính từ các nút radio
- Xử lý tính toán: Tính toán kết quả và kiểm tra lỗi (như chia cho 0).
- Hiện thị kết quả: Hiện thị kết quả ra màn hình.
- Reset dữ liệu: Xóa toàn bộ dữ liệu đã nhập để thực hiện phép tính mới.

#### 3.1.1. Biểu đồ phân cấp chức năng



- Cấp cao nhất là toàn bộ chương trình máy tính.
- Các nhánh con chia thành nhóm chức năng chính như quản lý nhập dữ liệu, quản lý phép toán, xử lý phép toán và giao diện người dùng.
- Mỗi chức năng lại chia nhỏ ra thành các chức năng con như xử lý lỗi, hiển thị kết quả, nhập số và chọn phép toán...v.v.

### 3.2.Sơ đồ các thuật toán chính



#### ❖ Các thuật toán chính có trong Simple Calculator GUI

- Nhập dữ liệu
  - Chức năng: Cho phép người dùng nhập vào hai số để thực hiện phép tính.
  - Đầu vào:
    - + Số 1 (entry\_num1)
    - + Số 2 (entry\_num2)
  - Xử lý:
    - + Đọc giá trị từ hai ô nhập liệu.
    - + Kiểm tra kiểu dữ liệu có hợp lệ không ( phải là số).
  - Đầu ra: Trả về hai số đã nhập nếu hợp lệ, hoặc thông báo lỗi nếu không.
- Cho phép toán

- Chức năng: Cho phép người dùng chọn một trong bốn phép toán cơ bản: cộng, trừ, nhân, chia.
- Đầu vào: Phép toán được chọn qua các nút radio ( radio buttons).
- Xử lý:
  - + Gán giá trị +, -, \*, / vào biến operation.
  - + Kiểm tra phép toán có hợp lệ không ( đặc biệt là chia cho 0).
- Đầu ra: Phép toán đã chọn hoặc thông báo lỗi nếu sai.
- **Tính toán kết quả**
  - Chức năng: Thực hiện phép tính dựa trên hai số nhập vào và phép toán được chọn.
  - Đầu vào: num1, num2, operation
  - Xử lý:
    - + Nếu phép toán là:
      - + “+”: cộng hai số.
      - + “-”: trừ hai số.
      - + “\*”: nhân hai số.
      - + “/”: kiểm tra mẫu số khác 0 rồi thực hiện chia.
    - + Bắt lỗi chia cho 0 hoặc dữ liệu không hợp lệ.
  - Đầu ra: Kết quả phép tính hoặc thông báo lỗi.
- **Hiện thị kết quả**
  - Chức năng: Hiện thị kết quả tính toán trên giao diện.
  - Đầu vào : Kết quả tính toán.
  - Xử lý: Gán kết quả vào biến result\_var để hiện thị lên Label.
  - Đầu ra: Chuỗi “ Kết quả:...” hiện thị trên giao diện.
- **Reset chương trình**
  - Chức năng: Xóa dữ liệu nhập và kết quả để thực hiện lại phép tính mới.
  - Đầu vào: Nhấn nút “ Reset”.
  - Xử lý:



- + Xóa nội dung trong hai ô nhập liệu.
- + Xóa kết quả.
- + Đặt lại phép toán mặc định là “+”
- Đầu ra: Giao diện trở về trạng thái ban đầu.

### 3.3. Cấu trúc dữ liệu

Trong chương trình **Máy tính đơn giản**, dữ liệu được lưu trữ thông qua các thành phần giao diện (tkinter) và biến điều khiển như sau:

- **entry\_num1** và **entry\_num2** là hai ô nhập liệu (Entry widget), nơi người dùng nhập hai số cần tính toán.
- **operation** là một biến kiểu StringVar dùng để lưu phép toán người dùng chọn, bao gồm: cộng (+), trừ (-), nhân ( $\times$ ), chia ( $\div$ ).
- **result\_var** là một biến StringVar chứa chuỗi kết quả sẽ được hiển thị trên giao diện sau khi thực hiện tính toán.
- Danh sách ops chứa các phép toán hợp lệ và được dùng để tạo các nút RadioButton cho người dùng lựa chọn phép tính.
- Ngoài ra còn có các đối tượng như root, mainframe, btn\_calculate, btn\_reset, ... là các thành phần của giao diện chính, được tổ chức và hiển thị qua tkinter.

Toàn bộ dữ liệu được xử lý theo phiên chạy, không lưu trữ vào file hay cơ sở dữ liệu. Thông tin chỉ tồn tại trong thời gian sử dụng ứng dụng.

### 3.4. Chương trình

Chương trình chính gồm hai hàm chính và phần giao diện người dùng được khởi tạo ở cuối chương trình:

- Hàm calculate(): Hàm này chịu trách nhiệm đọc dữ liệu từ hai ô nhập, kiểm tra phép toán được chọn, thực hiện tính toán và hiển thị kết quả. Nếu dữ liệu đầu vào không hợp lệ hoặc xảy ra lỗi chia cho 0 thì sẽ hiển thị thông

báo lỗi cho người dùng bằng hộp thoại cảnh báo (`messagebox.showerror`).

Kết quả hợp lệ sẽ được hiển thị thông qua `result_var`.

- Hàm `reset()`: Hàm này dùng để đưa chương trình về trạng thái ban đầu. Nó xóa dữ liệu đã nhập trong hai ô nhập số, xóa kết quả và đưa phép toán về mặc định là phép cộng. Đồng thời, con trỏ chuột sẽ được đưa về ô nhập đầu tiên để tiện thao tác lại.

- Giao diện người dùng (GUI):

Giao diện được xây dựng bằng thư viện `tkinter`. Gồm các phần:

- Hai ô nhập liệu cho số thứ nhất và số thứ hai.
- Bốn nút chọn phép toán: cộng, trừ, nhân, chia.
- Hai nút chức năng: Tính và Reset.
- Hiện thị kết quả.
- Khi người dùng nhấn phím `Enter`, chương trình sẽ tự động gọi hàm `calculate()`.

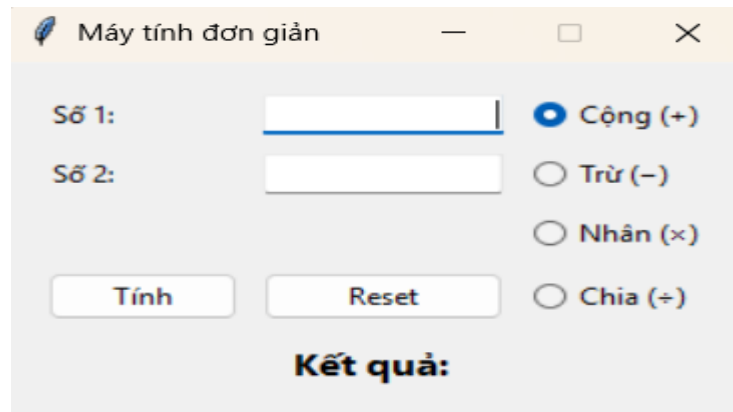
Tổ chức chương trình theo hướng chia nhỏ chức năng giúp mã nguồn dễ hiểu, dễ bảo trì và mở rộng. Các thao tác đều được phản hồi trực tiếp thông qua giao diện người dùng thân thiện.

## CHƯƠNG 4: THỰC NGHIỆM VÀ KẾT LUẬN

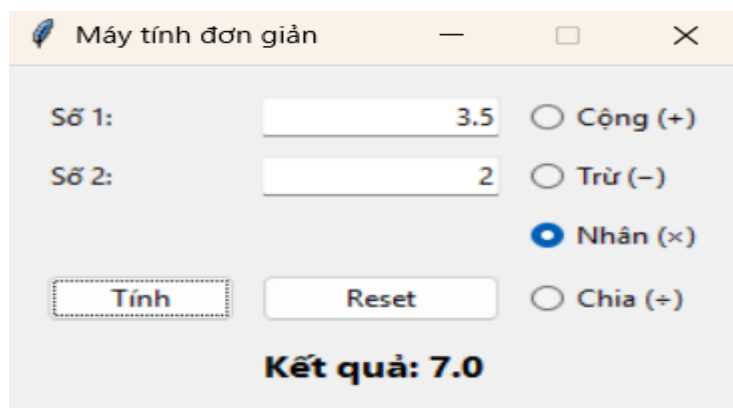
### 4.1. Thực nghiệm

Chương trình đã được chạy thử và kiểm tra các tính năng chính:

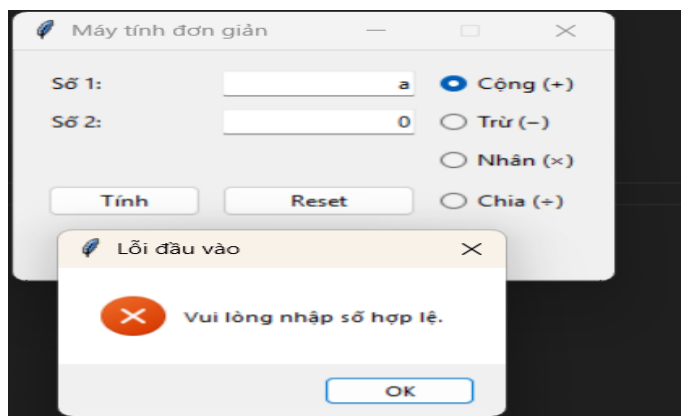
- Giao diện:



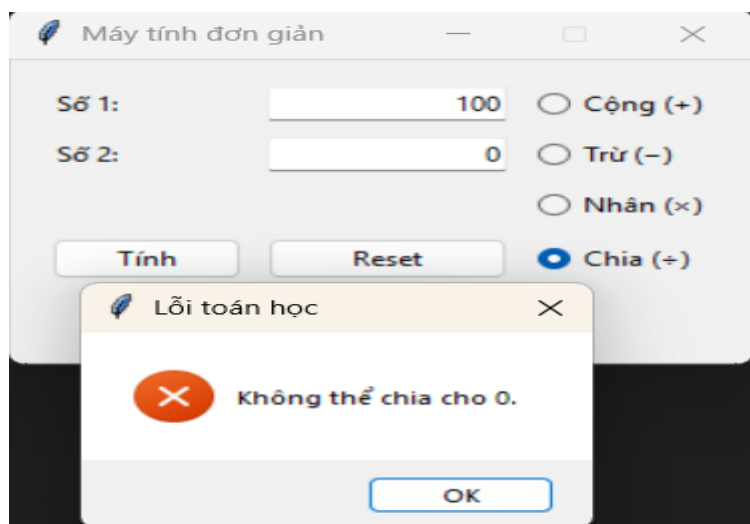
- Nhập số hợp lệ và chọn phép tính: Nhập 3.5 và 2, chọn phép tính (x), kết quả hiển thị là Kết quả: 7.0.



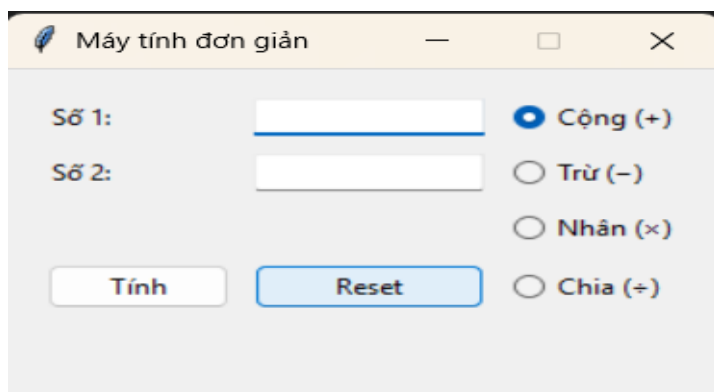
- Thử với dữ liệu sai định dạng (nhập chữ): Nhập a và 0, chương trình hiển thị hợp thoại lỗi “Vui lòng nhập số hợp lệ”



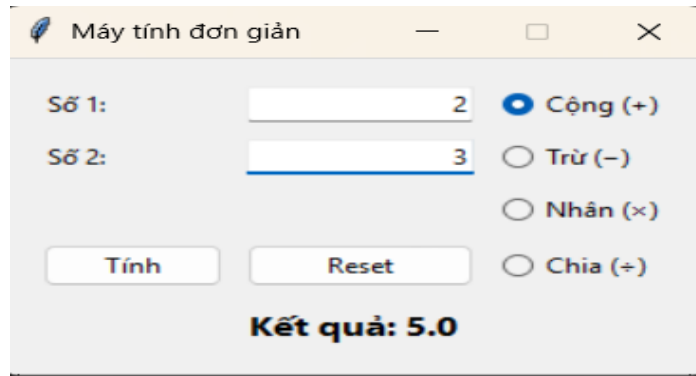
- Thử chia cho 0: Nhập 100 và 0, chọn phép chia > Hộp thoại “ Không thể chia cho 0”.



- Chức năng Reset: Sau khi tính toán, nhấn nút Reset, hai ô nhập được xóa , phép toán về mặc định và kết quả bị xóa.



- Tương tác bằng phím Enter: Sau khi nhập dữ liệu, nhấn phím Enter để thực hiện tính toán nhanh, không cần nhấn nút Tính.



## 4.2. Kết luận

Qua quá trình thiết kế và xây dựng, chương trình Máy tính đơn giản bằng Python đã hoàn thành với đầy đủ các chức năng cơ bản như: cộng, trừ, nhân, chia, xử lý lỗi đầu vào, hiển thị kết quả và hỗ trợ người dùng thao tác dễ dàng bằng giao diện trực quan.

### ❖ Những điểm đạt được:

- Giao diện rõ ràng, trực quan bằng tkinter.
- Xử lý lỗi hợp lý: nhập sai định dạng, chia cho 0.
- Tổ chức mã nguồn rõ ràng, dễ bảo trì và mở rộng.
- Phản hồi tức thời khi người dùng thao tác.

### ❖ Hạn chế:

- Chỉ thực hiện được bốn phép tính cơ bản, chưa hỗ trợ biểu thức phức tạp.
- Giao diện chưa có nhiều tùy chọn về màu sắc hay chủ đề.

### ❖ Hướng phát triển:

- Bổ sung các chức năng nâng cao như phép lũy thừa, căn bậc hai, phần trăm,...
- Hỗ trợ nhập toàn biểu thức (dạng  $3+5*2$ ) và tính toán theo thứ tự ưu tiên.
- Nâng cấp giao diện đồ họa thân thiện hơn với người dùng.