

TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP
KHOA ĐIỆN TỬ - BỘ MÔN CÔNG NGHỆ THÔNG TIN



BÀI TẬP KẾT THÚC MÔN HỌC
MÔN: LẬP TRÌNH PYTHON

NGÀNH : KỸ THUẬT MÁY TÍNH

HỆ : ĐẠI HỌC CHÍNH QUY

THÁI NGUYÊN – 2025

TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP
KHOA ĐIỆN TỬ - BỘ MÔN CÔNG NGHỆ THÔNG TIN



BÀI TẬP KẾT THÚC MÔN HỌC
MÔN: LẬP TRÌNH PYTHON

GIÁO VIÊN HƯỚNG DẪN	:	TS.Nguyễn Văn Huy
HỌ TÊN SINH VIÊN	:	Lý Văn Kiên
MSSV	:	K225480106101
LỚP	:	K58KTP

THÁI NGUYÊN - 2025

BÀI TẬP KẾT THÚC MÔN HỌC
MÔN HỌC: LẬP TRÌNH PYTHON
BỘ MÔN : CÔNG NGHỆ THÔNG TIN

Sinh viên: Lý Văn Kiên

MSSV: K225480106101

Lớp: K58KTP

Ngành: Kỹ Thuật Máy Tính

Giáo viên hướng dẫn: TS. Nguyễn Văn Huy

Ngày giao nhiệm vụ: 19/05/2025

Ngày hoàn thành nhiệm vụ: 08/06/2025

Tên đề tài: *Game Hangman GUI*

Yêu cầu: Xây dựng game Hangman với giao diện tkinter, người chơi đoán chữ cái để tìm từ được lưu trong file words.txt.

.

GIÁO VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

TRƯỜNG ĐHKTCN
KHOA ĐIỆN TỬ

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

PHIẾU GHI ĐIỂM

Sinh viên: Lý Văn Kiên

Lớp: K58KTP

GVHD: TS. Nguyễn Văn Huy

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Xếp loại : Điểm :

Thái Nguyên, ngày tháng năm 2025.

GIÁO VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

LỜI CAM ĐOAN

Em xin cam đoan rằng bài tập kết thúc môn Lập trình Python là do chính em thực hiện dưới sự hướng dẫn của thầy Nguyễn Văn Huy. Toàn bộ nội dung, số liệu và kết quả trình bày trong bài báo cáo đều là trung thực, không sao chép từ bất kỳ nguồn nào nếu không được trích dẫn rõ ràng. Em khẳng định rằng mọi thông tin và kết quả trong bài là do em tự thu thập và xử lý, không vi phạm quy định về đạo đức học thuật.

Nếu có bất kỳ sai sót hay vi phạm nào trong quá trình thực hiện, em xin hoàn toàn chịu trách nhiệm trước giảng viên bộ môn và nhà trường.

Em xin chân thành cảm ơn!

Họ và tên

Lý Văn Kiên

MỤC LỤC

LỜI CAM ĐOAN	3
DANH MỤC BẢNG VẼ VÀ ĐỒ THỊ	5
LỜI NÓI ĐẦU	6
CHƯƠNG 1. GIỚI THIỆU ĐẦU BÀI	7
1.1. Mục tiêu đề tài	7
1.2. Tính năng chính và thách thức	7
1.2.1. Tính năng	7
1.2.2. Thách thức	7
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT	8
2.1. Giao diện đồ họa với tkinter	8
2.2. Xử lý logic trò chơi bằng Python	8
2.3. Xử lý file dữ liệu	8
CHƯƠNG 3. THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH	10
3.1. Sơ đồ khối hệ thống	10
3.1.1. Các module chính trong chương trình	10
3.1.2. Sơ đồ phân cấp chức năng	11
3.2. Sơ đồ khối các thuật toán chính	11
3.2.1. Bắt đầu trò chơi	12
3.2.2. Xử lý đoán chữ	13
3.3. Cấu trúc dữ liệu	13
3.4. Cấu trúc chương trình	14
CHƯƠNG 4. THỰC NGHIỆM VÀ KẾT LUẬN	15
4.1. Kết quả thực nghiệm	15
4.2. Kết luận và định hướng phát triển	17

DANH MỤC BẢNG VẼ VÀ ĐỒ THỊ

Hình 3.1: Sơ đồ phân cấp hệ thống game Hangman

Hình 3.2.1: Sơ đồ khối thuật toán bắt đầu trò chơi

Hình 3.2.1: Sơ đồ khối thuật toán xử lý đoán chữ

Hình 4.1: Giao diện GUI game Hangman

Hình 4.2: Kết quả thắng game hangman

Hình 4.3: Kết quả thua game hangman

LỜI NÓI ĐẦU

Trong quá trình học tập ngôn ngữ lập trình Python, việc vận dụng kiến thức vào các dự án thực tế là cách hiệu quả để củng cố tư duy lập trình và nâng cao kỹ năng giải quyết vấn đề. Python không chỉ mạnh mẽ trong xử lý dữ liệu mà còn hỗ trợ xây dựng các ứng dụng có giao diện người dùng thông qua các thư viện như Tkinter.

Với mục tiêu rèn luyện tư duy logic và làm quen với thiết kế giao diện đồ họa, em đã lựa chọn đề tài “Xây dựng game Hangman bằng GUI” làm bài tập kết thúc môn Lập trình Python. Game Hangman (còn gọi là "Treo cổ") là một trò chơi đơn giản nhưng thú vị, đòi hỏi người chơi phải đoán từ theo từng chữ cái, qua đó giúp luyện khả năng suy đoán và ghi nhớ. Việc hiện thực hóa trò chơi này bằng Python giúp em củng cố kiến thức về xử lý chuỗi, điều kiện, vòng lặp, cấu trúc chương trình và giao diện người dùng.

Báo cáo này trình bày quá trình thiết kế, lập trình và thử nghiệm game Hangman, đồng thời phản ánh những kinh nghiệm mà em đã tích lũy được trong suốt quá trình thực hiện.

Em xin gửi lời cảm ơn chân thành đến thầy Nguyễn Văn Huy đã tận tình giảng dạy và hướng dẫn em trong quá trình học tập và thực hiện đề tài. Em kính mong nhận được sự góp ý để hoàn thiện tốt hơn trong các dự án sau này.

CHƯƠNG 1. GIỚI THIỆU ĐẦU BÀI

1.1. Mục tiêu đề tài

Đề tài yêu cầu xây dựng một chương trình trò chơi Hangman (Treo cổ) đơn giản sử dụng ngôn ngữ lập trình Python kết hợp với thư viện giao diện tkinter để xây dựng đồ họa GUI. Trò chơi có nhiệm vụ chọn một từ bí mật ngẫu nhiên, và người chơi phải lần lượt đoán từng chữ cái trong từ đó. Nếu đoán sai quá số lần cho phép, hình vẽ người treo cổ sẽ được hoàn thiện dần, và người chơi sẽ thua khi toàn bộ hình được vẽ xong. Ngược lại, nếu người chơi đoán đúng toàn bộ các chữ cái trước khi vượt quá giới hạn sai, họ sẽ chiến thắng.

1.2. Tính năng chính và thách thức

1.2.1. Tính năng

Chương trình có các tính năng chính bao gồm: đọc từ ngẫu nhiên từ file words.txt, hiển thị từ bí mật dưới dạng các dấu gạch dưới, cho phép người chơi nhập ký tự để đoán, theo dõi số lần đoán sai, hiển thị hình vẽ người treo cổ tương ứng, thông báo kết quả thắng hoặc thua, và cung cấp nút "Chơi lại" để khởi động trò chơi mới.

1.2.2. Thách thức

Thách thức trong quá trình thực hiện bao gồm việc kết hợp giữa xử lý logic trò chơi với giao diện người dùng, đảm bảo các thành phần hoạt động đồng bộ và chính xác, cũng như việc tổ chức mã nguồn sao cho dễ đọc, dễ mở rộng, và sử dụng lập trình hướng đối tượng hiệu quả.

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

2.1. Giao diện đồ họa với tkinter

Tkinter là một thư viện GUI tích hợp sẵn trong Python, cho phép xây dựng giao diện người dùng đơn giản mà không cần cài thêm thư viện bên ngoài. Trong chương trình Hangman, tkinter được sử dụng để tạo cửa sổ chính, khung vẽ hình ảnh (Canvas), nhãn hiển thị (Label), ô nhập liệu (Entry), và nút điều khiển (Button). Các sự kiện như nhập ký tự được xử lý bằng phương thức bind, giúp chương trình phản hồi theo thời gian thực. Việc xây dựng GUI bằng tkinter giúp trò chơi trở nên trực quan, dễ sử dụng, phù hợp với người học mới làm quen với lập trình giao diện.

2.2. Xử lý logic trò chơi bằng Python

Logic của trò chơi được tổ chức theo mô hình lập trình hướng đối tượng thông qua hai lớp chính: HangmanGame và HangmanGUI. Trong đó, HangmanGame phụ trách toàn bộ phần xử lý nội dung, bao gồm việc chọn từ ngẫu nhiên, kiểm tra ký tự đoán, đếm số lần đoán sai và kiểm tra trạng thái thắng/thua. Các kiểu dữ liệu cơ bản như str, list, set được sử dụng để biểu diễn từ bí mật, chữ cái đã đoán và định dạng hiển thị từ. Lớp HangmanGUI tập trung vào việc xử lý giao diện, cập nhật kết quả và hiển thị phản hồi người dùng theo dữ liệu từ lớp logic.

2.3. Xử lý file dữ liệu

Dữ liệu từ bí mật được lưu trong file words.txt dưới dạng văn bản, mỗi từ trên một dòng. Chương trình mở file này bằng lệnh with open(...) và đọc toàn bộ nội dung vào một danh sách. Sau khi lọc bỏ dòng trống và chuẩn hóa định dạng chữ in hoa, một từ được chọn ngẫu nhiên để sử dụng trong vòng chơi hiện tại. Thiết kế này

giúp chương trình dễ dàng mở rộng hoặc cập nhật danh sách từ bằng cách sửa trực tiếp nội dung file mà không cần thay đổi mã nguồn.

CHƯƠNG 3. THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH

3.1. Sơ đồ khối hệ thống

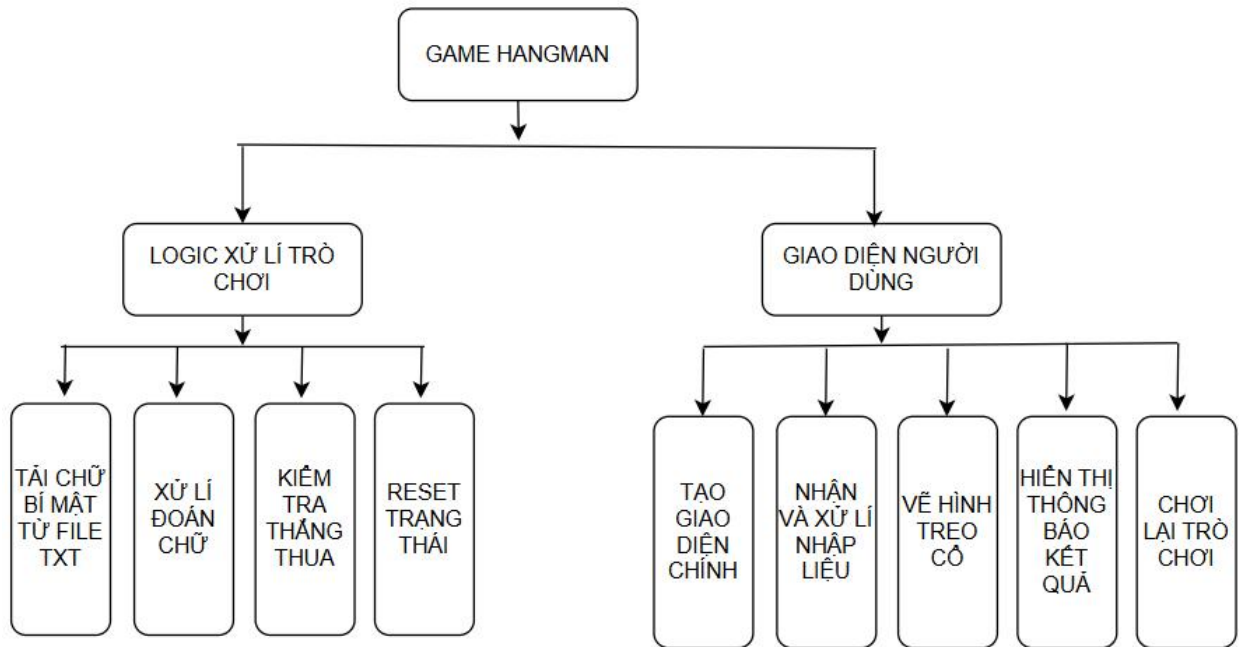
3.1.1. Các module chính trong chương trình

Chương trình Hangman được xây dựng với kiến trúc đơn giản nhưng có tổ chức rõ ràng, chia thành hai module chính: module xử lý logic trò chơi và module giao diện người dùng. Mỗi module đóng vai trò riêng biệt nhưng liên kết chặt chẽ để tạo nên một hệ thống hoàn chỉnh, dễ sử dụng và dễ bảo trì.

- Module thứ nhất là module logic xử lý trò chơi, được hiện thực thông qua lớp HangmanGame. Đây là thành phần trung tâm chịu trách nhiệm quản lý toàn bộ trạng thái và quy tắc của trò chơi. Nó bao gồm các chức năng như đọc từ ngẫu nhiên từ file words.txt, theo dõi các chữ cái mà người chơi đã đoán, cập nhật số lần đoán sai và kiểm tra điều kiện thắng hoặc thua. Khi khởi động, module này sẽ chọn một từ mới từ danh sách, đặt lại toàn bộ trạng thái liên quan đến trò chơi, và chuẩn bị sẵn sàng để xử lý các lần đoán tiếp theo từ người chơi.
- Module thứ hai là module giao diện người dùng, được xây dựng bằng lớp HangmanGUI sử dụng thư viện tkinter. Giao diện này bao gồm các thành phần trực quan như nhãn tiêu đề, ô nhập chữ cái, nút chơi lại, vùng hiển thị từ cần đoán, nhãn thông báo số lần đoán sai và vùng vẽ hình người treo cổ bằng Canvas. Giao diện được tổ chức sao cho người chơi có thể dễ dàng nhập dữ liệu, xem kết quả và nhận được phản hồi ngay lập tức từ hệ thống. Toàn bộ giao diện được cập nhật tự động theo trạng thái hiện tại của trò chơi do module logic cung cấp.

3.1.2. Sơ đồ phân cấp chức năng

Xét về mặt phân cấp chức năng, hệ thống chương trình có thể được chia thành ba tầng.



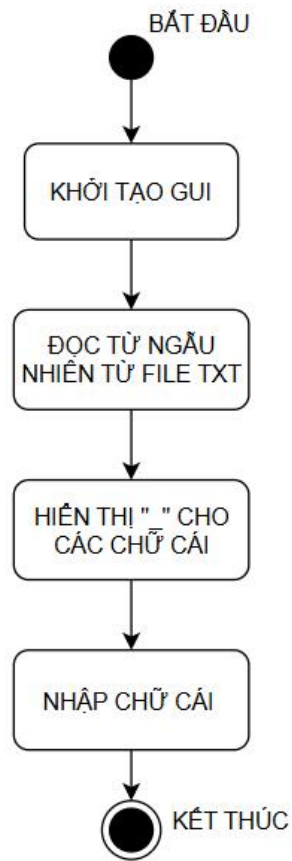
Hình 3.1: Sơ đồ phân cấp hệ thống game Hangman

3.2. Sơ đồ khối các thuật toán chính

Thuật toán chính của chương trình bắt đầu bằng việc khởi tạo trò chơi và chọn ngẫu nhiên một từ bí mật. Khi người dùng nhập một chữ cái, chương trình kiểm tra xem chữ đó đã được đoán trước đó chưa. Nếu chưa, chữ cái được thêm vào danh sách chữ đã đoán, sau đó kiểm tra xem nó có nằm trong từ bí mật không. Nếu đúng, chương trình cập nhật hiển thị từ đang đoán bằng cách thay thế các dấu gạch dưới tương ứng. Nếu sai, số lần đoán sai sẽ tăng và chương trình cập nhật hình ảnh treo

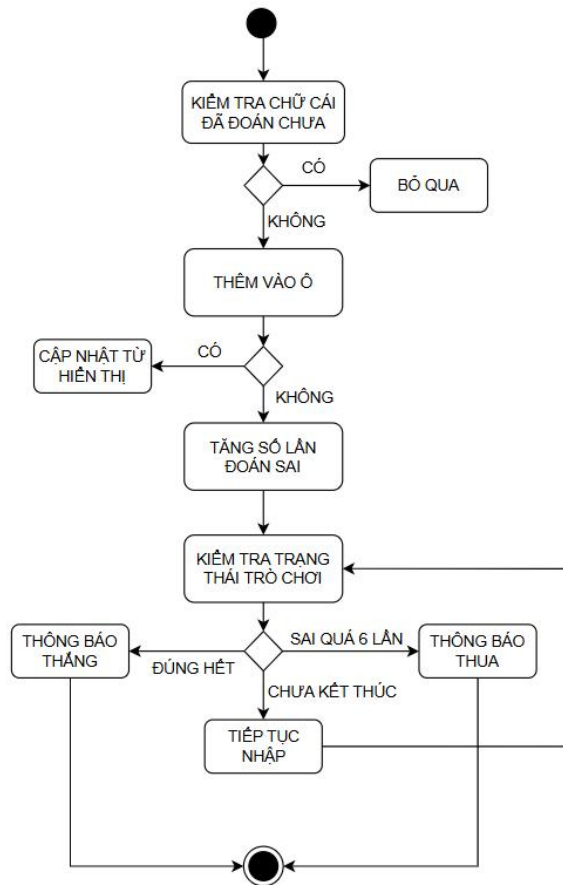
cổ tương ứng. Sau mỗi lần đoán, chương trình kiểm tra điều kiện thắng hoặc thua để kết thúc trò chơi hoặc tiếp tục nhận dữ liệu.

3.2.1. Bắt đầu trò chơi



Hình 3.2.1: Sơ đồ khối thuật toán bắt đầu trò chơi

3.2.2. Xử lý đoán chữ



Hình 3.2.1: Sơ đồ khối thuật toán xử lý đoán chữ

3.3. Cấu trúc dữ liệu

Chương trình sử dụng một số biến quan trọng để lưu trữ trạng thái. Biến `secret_word` chứa từ bí mật đã chọn, `guessed_letters` là tập hợp các chữ cái người chơi đã đoán, và `wrong_guesses` là số lần đoán sai hiện tại. Các biến này được khởi tạo trong phương thức `reset_game` và cập nhật liên tục trong quá trình chơi. Việc sử dụng tập hợp (set) cho các chữ cái đã đoán giúp chương trình kiểm tra nhanh và loại bỏ trùng lặp.

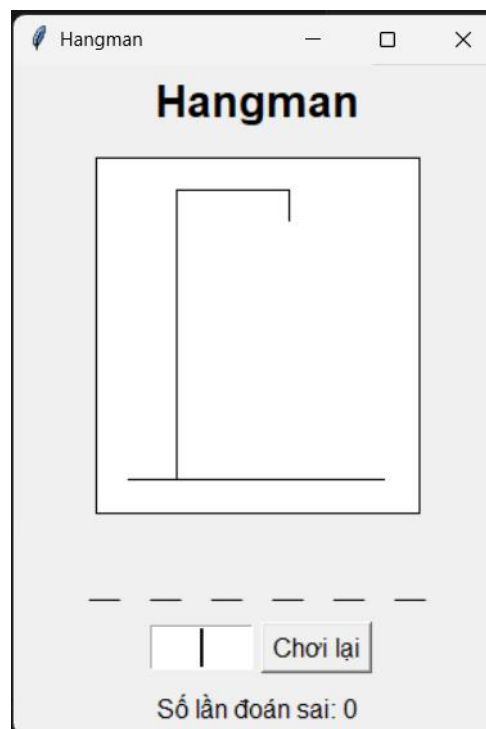
3.4. Cấu trúc chương trình

Toàn bộ chương trình nằm trong file `main.py`, chia thành hai lớp chính. Lớp `HangmanGame` chứa các phương thức xử lý logic như `load_word`, `guess_letter`, `get_display_word`, `is_won`, `is_lost` và `reset_game`. Lớp `HangmanGUI` quản lý giao diện với các phương thức như `create_widgets`, `handle_guess_with_letter`, `draw_hangman`, `update_display` và `reset_game`. Mỗi phần đều được thiết kế riêng biệt để đảm bảo tính mô-đun và dễ bảo trì.

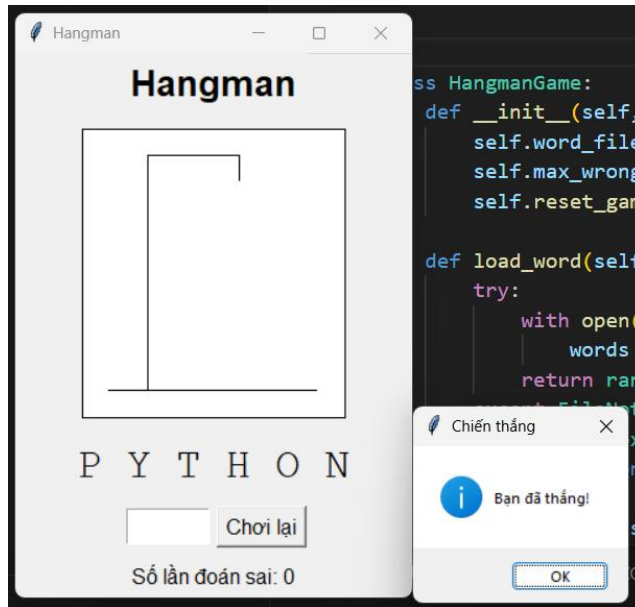
CHƯƠNG 4. THỰC NGHIỆM VÀ KẾT LUẬN

4.1. Kết quả thực nghiệm

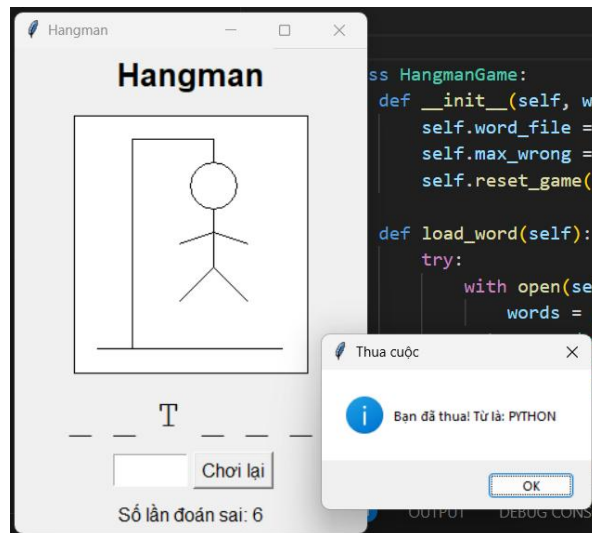
Chương trình đã được chạy thử nghiệm trên môi trường Python 3.9 sử dụng Visual Studio Code với hệ điều hành Windows. Khi khởi động, trò chơi hiển thị đầy đủ giao diện với tiêu đề, vùng nhập chữ cái, hình vẽ, từ đang đoán và nút "Chơi lại". Khi người chơi nhập đúng chữ cái, các ký tự tương ứng trong từ được hiện ra. Khi nhập sai, hình người treo cổ được vẽ dần từ đầu, thân, tay cho đến chân. Sau khi đoán hết từ hoặc vượt quá số lần sai cho phép, hộp thoại thông báo kết quả xuất hiện và ô nhập liệu bị khóa. Nút "Chơi lại" cho phép bắt đầu lại ván mới với từ khác. Toàn bộ chức năng hoạt động ổn định, không xuất hiện lỗi trong quá trình thử nghiệm.



Hình 4.1: Giao diện GUI game Hangman



Hình 4.2: Kết quả thắng game hangman



Hình 4.3: Kết quả thua game hangman

4.2. Kết luận và định hướng phát triển

Chương trình đã đáp ứng đầy đủ các yêu cầu đề bài, bao gồm cả phần giao diện, logic xử lý, và khả năng tương tác trực quan. Người thực hiện đã học được cách kết hợp lập trình hướng đối tượng với giao diện tkinter, xử lý tệp dữ liệu và tổ chức mã nguồn theo hướng mô-đun. Trong tương lai, chương trình có thể được cải tiến thêm như cho phép lựa chọn độ khó, hiển thị gợi ý cho từ cần đoán, thêm chế độ nhiều người chơi hoặc cập nhật kho từ trực tiếp từ giao diện thay vì chỉnh sửa file thủ công.

