**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA KỸ THUẬT MÁY TÍNH**

**HỌ VÀ TÊN SINH VIÊN**

**Trần Anh Tuấn  
Trần Văn Cường**

**BÁO CÁO ĐỒ ÁN MÔN HỌC CUỐI KỲ**

**RÚT GỌN BẢNG TRẠNG THÁI   
VÀ HIỂN THỊ LÊN WEB BROWSER**

****

**GIẢNG VIÊN HƯỚNG DẪN**

**ThS. Trương Văn Cương**

**THÀNH PHỐ HỒ CHÍ MINH, 2023**

|  |  |
| --- | --- |
| **ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**  **TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **ĐỘC LẬP – TỰ DO – HẠNH PHÚC** |

**CHI TIẾT**

|  |  |
| --- | --- |
| **TÊN ĐỒ ÁN TIẾNG VIỆT: RÚT GỌN BẢNG TRẠNG THÁI VÀ HIỂN THỊ LÊN WEB BROWSER** | |
| **TÊN ĐỒ ÁN TIẾNG ANH: MINIMIZATION STATE TABLE AND DISPLAYING ON WEB BROWSER** | |
| **Giảng viên hướng dẫn:** ThS. Trương Văn Cương, Khoa Kỹ thuật máy tính | |
| **Thời gian thực hiện:** Từ ngày: 15/05/2023 Đến ngày: 31/05/2023 | |
| **Sinh viên thực hiện:** Trần Anh Tuấn, Trần Văn Cường | |
| **Ngôn ngữ lập trình:** Python  **IDE (Integrated Development Environment):** Visual Studio Code | |
| **Chứng nhận của giảng viên**  (chữ ký và họ tên đầy đủ) | **Tp. Hồ Chí Minh, 31/05/2023**  **Sinh viên**  (chữ ký và họ tên đầy đủ) |

**MỤC LỤC**

[**CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI** 1](#_Toc136341786)

[**1.1** **Giới thiệu** 1](#_Toc136341787)

[**1.1.1 Giới thiệu chung** 1](#_Toc136341788)

[**1.1.2 Mô tả hoạt động của chương trình** 1](#_Toc136341789)

[**1.2 Ngôn ngữ, thư viện, framework sử dụng** 1](#_Toc136341790)

[**1.2.1 Ngôn ngữ lập trình Python** 1](#_Toc136341791)

[**1.2.2 Ngôn ngữ web HTML, CSS** 1](#_Toc136341792)

[**1.2.3 Thư viện Pandas** 2](#_Toc136341793)

[**1.2.4 Framework Flask** 2](#_Toc136341794)

[**CHƯƠNG 2: LƯU ĐỒ THUẬT TOÁN VÀ CÁC CHƯƠNG TRÌNH** 3](#_Toc136341795)

[**2.1 Lưu đồ thuật toán chương trình chính** 3](#_Toc136341796)

[**2.2 Lưu đồ thuật toán của chương trình phụ** 4](#_Toc136341797)

[**2.2.1 Lưu đồ thuật toán chương trình chia nhóm theo từng Output** 4](#_Toc136341798)

[**2.2.2 Lưu đồ thuật toán chương trình xác định nhóm TTKT** 5](#_Toc136341799)

[**2.2.3 Lưu đồ thuật toán chương trình kiểm tra điều kiện và tách nhóm** 6](#_Toc136341800)

[**2.2.4 Lưu đồ thuật toán chương trình rút gọn trạng thái trong nhóm tương đương** 7](#_Toc136341801)

[**2.3 Source code chương trình** 8](#_Toc136341802)

[**2.3.1 Hướng tiếp cận 1** 8](#_Toc136341803)

[**2.3.2 Hướng tiếp cận 2** 8](#_Toc136341804)

[**CHƯƠNG 3: THỰC NGHIỆM** 9](#_Toc136341805)

[**3.1 Điều kiện thực nghiệm** 9](#_Toc136341806)

[**3.2 Kết quả thu được** 9](#_Toc136341807)

[**CHƯƠNG 4: ĐÁNH GIÁ** 14](#_Toc136341808)

[**4.1 Điểm mạnh của chương trình** 14](#_Toc136341809)

[**4.2 Một số hạn chế còn tồn tại** 14](#_Toc136341810)

[**4.3 Kết luận** 14](#_Toc136341811)

[**TÀI LIỆU THAM KHẢO** 15](#_Toc136341812)

**DANH MỤC HÌNH ẢNH**

[**Hình ảnh 2. 1 Lưu đồ thuật toán chương trình chính** 3](#_Toc136300847)

[**Hình ảnh 2.2 1 Lưu đồ thuật toán chương trình chia nhóm theo từng Output** 4](#_Toc136301979)

[**Hình ảnh 2.2 2 Lưu đồ thuật toán chương trình xác định nhóm TTKT** 5](#_Toc136301980)

[**Hình ảnh 2.2 3 Lưu đồ thuật toán chương trình kiểm tra điều kiện và tách nhóm** 6](#_Toc136301981)

[**Hình ảnh 2.2 4 Lưu đồ thuật toán chương trình rút gọn trạng thái trong nhóm tương đương** 7](#_Toc136301982)

[**Hình ảnh 3. 1 Giao diện khi up lên browser** 9](#_Toc136302243)

# **CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI**

## **Giới thiệu**

### **1.1.1 Giới thiệu chung**

Chỉ trong vỏn vẹn vài thập kỉ vừa qua, nhân loại chúng ta mặc dù đã trải qua biết bao là bước tiến vĩ đại và mạnh mẽ trong cả lĩnh vực phần cứng và phần mềm, tuy nhiên việc sử dụng FSM vẫn là một phương pháp quan trọng để mô hình hóa chương trình. Gắn liền với FSM, việc quản lý số lượng trạng thái lại là một thách thức không nhỏ đối với các nhà phát triển bởi có thể có nhiều FSM cùng chung chức năng nhưng khác nhau về số lượng trạng thái.

→ Việc tìm ra FSM có chức năng tương tự với số lượng trạng thái là nhỏ nhất (rút gọn trạng thái) cực kỳ quan trọng khi nó mang lại hiệu suất cao khi thiết kế phần cứng cũng như thời gian xử lý của chương trình sẽ được tối ưu.

### **1.1.2 Mô tả hoạt động của chương trình**

- Ban đầu, chúng ta sẽ sử dụng ngôn ngữ lập trình Python và thư viện Pandas để thu thập data của bảng trạng thái cần rút gọn từ file excel.

- Sau đó, thực hiện rút gọn trạng thái dựa trên data đã thu thập.

- Cuối cùng, lấy data sau khi rút gọn hiển thị lên web browser bằng cách sử dụng HTML, CSS và framework Flask.

## **1.2 Ngôn ngữ, thư viện, framework sử dụng**

### **1.2.1 Ngôn ngữ lập trình Python**

- Python là ngôn ngữ lập trình cấp cao mạnh mẽ, dễ học với cú pháp đơn giản, ngắn gọn và có thể ứng dụng ở nhiều lĩnh vực khác nhau như: khoa học dữ liệu, AI, phát triển phần mềm nhúng, …

- Ngoài ra, Python còn được hỗ trợ bởi nhiều thư viện mạnh mẽ như Pandas, Openpyxl, Matplotlib, …

- Các lý do nhóm lựa chọn Python thay vì C++ :

+ Hỗ trợ đọc file excel nhanh gọn.

+ Cú pháp ngắn gọn, dễ đọc hiểu, bảo trì code.

+ Là một cách để học thêm ngôn ngữ mới thông qua dự án thực tế.

### **1.2.2 Ngôn ngữ web HTML, CSS**

- HTML (HyperText Markup Language) là ngôn ngữ đánh dấu siêu văn bản.

- Không phải là ngôn ngữ lập trình.

- HTML có tác dụng tạo bố cục và định dạng trang web.

- CSS (Cascading Style Sheets) định nghĩa về cách hiển thị của một tài liệu HTML.

- CSS đặc biệt hữu ích trong việc thiết kế Web. Nó giúp cho người thiết kế dễ dàng đặt các style đã được thiết kế lên bất kì page nào của website một cách nhanh chóng và đồng bộ.

### **1.2.3 Thư viện Pandas**

- Pandas là một thư viện mã nguồn mở được phát triển dựa trên ngôn ngữ Python bởi Wes Mckinney năm 2008.

- Pandas thực hiện phân tích và thao tác dữ liệu; bất kỳ loại xử lý, phân tích, lọc và tổng hợp dữ liệu.

- Những phần sử dụng trong đồ án:

+ Kiểu dữ liệu DataFrame: là 1 kiểu dữ liệu như mảng 2 chiều hoặc giống như bảng gồm nhiều hàng và cột.

* Hàm dropna() : giúp loại bỏ các dòng chứa giá trị NULL.
* Hàm rename() : giúp đổi tên index hàng hoặc tên cột.

+ Hàm read\_excel() : đọc data từ file excel và trả về dưới dạng một dataframe.

### **1.2.4 Framework Flask**

- Flask là một framework web nhẹ, linh hoạt và dễ sử dụng được viết bằng Python, cung cấp cho người dùng một cách tiếp cận tối giản hóa việc phát triển ứng dụng web.

- Nó được sử dụng để xây dựng các ứng dụng web, API và các dịch vụ web khác.

- Mục đích sử dụng trong đồ án: dùng để truyền data từ Python sang cho HTML để hiển thị lên web browser.

# **CHƯƠNG 2: LƯU ĐỒ THUẬT TOÁN VÀ CÁC CHƯƠNG TRÌNH**

## **2.1 Lưu đồ thuật toán chương trình chính**

A diagram of a flowchart

Description automatically generated with low confidence

**Hình ảnh 2. 1 Lưu đồ thuật toán chương trình chính**

## **2.2 Lưu đồ thuật toán của chương trình phụ**

### **2.2.1 Lưu đồ thuật toán chương trình chia nhóm theo từng Output**

A picture containing text, diagram, font, line

Description automatically generated

**Hình ảnh 2.2 1 Lưu đồ thuật toán chương trình chia nhóm theo từng Output**

### **2.2.2 Lưu đồ thuật toán chương trình xác định nhóm TTKT**

A picture containing text, diagram, screenshot, line

Description automatically generated

**Hình ảnh 2.2 2 Lưu đồ thuật toán chương trình xác định nhóm TTKT**

### **2.2.3 Lưu đồ thuật toán chương trình kiểm tra điều kiện và tách nhóm**

A diagram of a group list

Description automatically generated with low confidence

**Hình ảnh 2.2 3 Lưu đồ thuật toán chương trình kiểm tra điều kiện và tách nhóm**

### **2.2.4 Lưu đồ thuật toán chương trình rút gọn trạng thái trong nhóm tương đương**

A picture containing text, diagram, screenshot, line

Description automatically generated

**Hình ảnh 2.2 4 Lưu đồ thuật toán chương trình rút gọn trạng thái trong nhóm tương đương**

## **2.3 Source code chương trình**

### **2.3.1 Hướng tiếp cận 1**

**-** Code Python:

<https://github.com/CuongTranMCU/MinimizeStateTable/blob/master/Version1/minimizeState.py>

* Code HTML, CSS: tham khảo thêm ở link

<https://github.com/CuongTranMCU/MinimizeStateTable/blob/master/Version1/templates/index.html>

<https://github.com/CuongTranMCU/MinimizeStateTable/tree/master/Version1/static/css>

### **2.3.2 Hướng tiếp cận 2**

- Nhận thấy hướng tiếp cận 1 vẫn còn một số khuyết điểm về mặt thời gian khi phải rút gọn với số lượng trạng thái lớn nên nhóm đã quyết định đưa ra thêm hướng tiếp cận mới đối với bài toán.

- Code Python: tham khảo thêm ở link <https://github.com/CuongTranMCU/MinimizeStateTable/blob/master/Version2/minimizeState.py>

- Code HTML, CSS: tham khảo thêm ở link

<https://github.com/CuongTranMCU/MinimizeStateTable/blob/master/Version2/templates/index.html>

<https://github.com/CuongTranMCU/MinimizeStateTable/blob/master/Version2/static/css/styles.css>

# **CHƯƠNG 3: THỰC NGHIỆM**

## **3.1 Điều kiện thực nghiệm**

- Thực nghiệm chương trình bằng tệp 50 testcase với số lượng trạng thái cần rút gọn dao động từ 3 đến 10 000 trạng thái với điều kiện input và output đều chỉ có 1 bit giá trị.

## **3.2 Kết quả thu được**

A screenshot of a table

Description automatically generated with low confidence

**Hình ảnh 3. 1 Giao diện khi up lên browser**

**- Đối với tệp 25 testcase, số lượng trạng thái ≤ 30:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Số trạng thái trước khi rút gọn | Số trạng thái sau khi rút gọn | Thời gian xử lý cách 1 | Thời gian xử lý cách 2 | Kết quả đúng hay sai |
| 1 | 16 | 12 | 0.38837 | 0.46304 | Đúng |
| 2 | 8 | 4 | 0.40710 | 0.46198 | Đúng |
| 3 | 10 | 8 | 0.37952 | 0.47490 | Đúng |
| 4 | 26 | 21 | 0.47959 | 0.48900 | Đúng |
| 5 | 3 | 2 | 0.40832 | 0.53415 | Đúng |
| 6 | 7 | 4 | 0.49442 | 0.50992 | Đúng |
| 7 | 7 | 4 | 0.39485 | 0.46802 | Đúng |
| 8 | 7 | 3 | 0.41644 | 0.49207 | Đúng |
| 9 | 7 | 2 | 0.48264 | 0.49196 | Đúng |
| 10 | 10 | 4 | 0.42833 | 0.52699 | Đúng |
| 11 | 26 | 25 | 0.42755 | 0.52207 | Đúng |
| 12 | 4 | 4 | 0.55138 | 0.52800 | Đúng |
| 13 | 8 | 5 | 0.39337 | 0.56090 | Đúng |
| 14 | 10 | 5 | 0.33893 | 0.56992 | Đúng |
| 15 | 3 | 3 | 0.32787 | 0.49900 | Đúng |
| 16 | 7 | 5 | 0.44603 | 0.57392 | Đúng |
| 17 | 15 | 7 | 0.57216 | 0.61794 | Đúng |
| 18 | 6 | 4 | 0.40871 | 0.50498 | Đúng |
| 19 | 8 | 4 | 0.39881 | 0.55294 | Đúng |
| 20 | 15 | 11 | 0.38995 | 0.58300 | Đúng |
| 21 | 17 | 13 | 0.41808 | 0.55399 | Đúng |
| 22 | 26 | 21 | 0.45046 | 0.58301 | Đúng |
| 23 | 30 | 26 | 0.50265 | 0.51306 | Đúng |
| 24 | 30 | 25 | 0.43936 | 0.50798 | Đúng |
| 25 | 30 | 6 | 0.48817 | 0.54217 | Đúng |
| Thời gian xử lý trung bình | | | 0.43332 | 0.52499 |  |

**- Đối với tệp 5 testcase, số lượng trạng thái 100 -> 200:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Số trạng thái trước khi rút gọn | Số trạng thái sau khi rút gọn | Thời gian xử lý cách 1 | Thời gian xử lý cách 2 | Kết quả đúng hay sai |
| 1 | 196 | 163 | 2.62531 | 0.55497 | Chưa xác định |
| 2 | 119 | 107 | 1.45683 | 0.59999 | Chưa xác định |
| 3 | 144 | 128 | 1.66553 | 0.61208 | Chưa xác định |
| 4 | 152 | 127 | 2.93928 | 0.53400 | Chưa xác định |
| 5 | 172 | 146 | 3.57717 | 0.56800 | Chưa xác định |
| Thời gian xử lý trung bình | | | 2.45282 | 0.57381 |  |

**- Đối với tệp 5 testcase, số lượng trạng thái 500:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Số trạng thái trước khi rút gọn | Số trạng thái sau khi rút gọn | Thời gian xử lý cách 1 | Thời gian xử lý cách 2 | Kết quả đúng hay sai |
| 1 | 500 | 488 | 78.93497 | 0.78591 | Chưa xác định |
| 2 | 500 | 486 | 51.67404 | 0.74098 | Chưa xác định |
| 3 | 500 | 483 | 106.24608 | 0.73707 | Chưa xác định |
| 4 | 500 | 495 | 115.49494 | 0.77395 | Chưa xác định |
| 5 | 500 | 486 | 140.14605 | 0.76806 | Chưa xác định |
| Thời gian xử lý trung bình | | | 98.5 | 0.76119 |  |

**- Đối với tệp 5 testcase, số lượng trạng thái 1000:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Số trạng thái trước khi rút gọn | Số trạng thái sau khi rút gọn | Thời gian xử lý cách 1 | Thời gian xử lý cách 2 | Kết quả đúng hay sai |
| 1 | 1000 | 990 | X | 1.57859 | Chưa xác định |
| 2 | 1000 | 987 | X | 1.83449 | Chưa xác định |
| 3 | 1000 | 997 | X | 1.91267 | Chưa xác định |
| 4 | 1000 | 996 | X | 1.63239 | Chưa xác định |
| 5 | 1000 | 744 | 197.44328 | 1.01034 | Chưa xác định |
| Thời gian xử lý trung bình | | |  | 1.59369 |  |

**- Đối với tệp 5 testcase, số lượng trạng thái 5000:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Số trạng thái trước khi rút gọn | Số trạng thái sau khi rút gọn | Thời gian xử lý cách 1 | Thời gian xử lý cách 2 | Kết quả đúng hay sai |
| 1 | 5000 | 4947 | X | 31.01727 | Chưa xác định |
| 2 | 5000 | 4940 | X | 27.94767 | Chưa xác định |
| 3 | 5000 | 4816 | X | 24.77711 | Chưa xác định |
| 4 | 5000 | 4807 | X | 24.42100 | Chưa xác định |
| 5 | 5000 | 4676 | X | 23.91540 | Chưa xác định |
| Thời gian xử lý trung bình | | |  | 26.41569 |  |

**- Đối với tệp 5 testcase, số lượng trạng thái 10 000:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Số trạng thái trước khi rút gọn | Số trạng thái sau khi rút gọn | Thời gian xử lý cách 1 | Thời gian xử lý cách 2 | Kết quả đúng hay sai |
| 1 | 10 000 | 9839 | X | 100.47385 | Chưa xác định |
| 2 | 10 000 | 9798 | X | 102.78088 | Chưa xác định |
| 3 | 10 000 | 9225 | X | 134.52881 | Chưa xác định |
| 4 | 10 000 | 8895 | X | 88.69334 | Chưa xác định |
| 5 | 10 000 | 8863 | X | 88.65438 | Chưa xác định |
| Thời gian xử lý trung bình | | |  | 103.026252 |  |

* **Lưu ý:** Đối với 25 testcase có số lượng trạng thái từ 100 -> 10 000 trạng thái, nhóm không tìm ra được phương án hiệu quả để kiểm tra độ chính xác của kết quả rút gọn trạng thái nên nhóm đã so sánh kết quả của hai hướng tiếp cận và các bạn khác chung đề tài để tham khảo và do đó chỉ có thể kết luận kết quả rút gọn chính xác tương đối.

# **CHƯƠNG 4: ĐÁNH GIÁ**

## **4.1 Điểm mạnh của chương trình**

- Hướng tiếp cận 1:

* + Đối với bộ 25 testcase với số lượng trạng thái ≤ 30 thì thời gian rút gọn nhanh hơn khoảng 18% so với cách tiếp cận 2.
  + Độ chính xác của bộ 25 testcase với số lượng trạng thái ≤ 30 là 25/25 = 100%.
  + Có thể chạy được nếu thứ tự các trạng thái không được sắp xếp.

- Hướng tiếp cận 2:

* + Ngoại trừ bộ 25 testcase với số lượng trạng thái ≤ 30 thì 25 testcase còn lại hướng tiếp cận 2 đều có thời gian rút gọn nhanh hơn đáng kể so với hướng tiếp cận 1.
  + Có thể chạy được testcase với 10 000 trạng thái.
  + Độ chính xác của bộ 25 testcase với số lượng trạng thái ≤ 30 là 25/25 = 100%.

## **4.2 Một số hạn chế còn tồn tại**

- Hướng tiếp cận 1:

* + Code chưa thực sự tối ưu khi gặp số lượng testcase lớn (≥ 500).

- Hướng tiếp cận 2:

* + Không thể chạy được nếu thứ tự các trạng thái không được sắp xếp.

## **4.3 Kết luận**

- Vì số lượng testcase chưa quá lớn để kiểm tra nên khi deploy sản phẩm lên website cho users sử dụng thì sẽ gặp một vài lỗi rút gọn.

- Hướng phát triển trong tương lai:

+ Tối ưu hóa code để giảm thời gian rút gọn với số lượng trạng thái lớn và số lượng input, output nhiều hơn 1.

+ Hiển thị được sơ đồ trạng thái sau khi đã rút gọn.

+ Thực hiện lại bài toán với hướng tiếp cận bảng kéo theo.

+ Deploy được sản phẩm lên website.

# **TÀI LIỆU THAM KHẢO**

[1] Lâm Đức Khải (chủ biên), Hồ Ngọc Diễm, Trần Đại Dương, Giáo trình *Thiết kế luận lý số 2*, Nhà xuất bản Đại học Quốc gia Thành phố Hồ Chí Minh, 2019.