*BỘ GIÁO DỤC VÀ ĐÀO TẠO*

*TRƯỜNG ĐẠI HỌC CÔNG NGHỆ SÀI GÒN*

***KHOA CÔNG NGHỆ THÔNG TIN***

***ĐỒ ÁN TIN HỌC***

*Tên đề tài:*

*Quản lý công việc*

*GV hướng dẫn: Trần Văn Hùng*

*SV thực hiện:*

*Hàng Gia Thuận MSSV: DH52100524*

*Nguyễn Cảnh Thịnh MSSV: DH52105095*

***TP HỒ CHÍ MINH – NĂM 2023***

***MỤC LỤC***

[*CHƯƠNG I : GIỚI THIỆU 3*](#_Toc154175565)

[*1. Mục tiêu của báo cáo. 3*](#_Toc154175566)

[*2. Những thách thức cần giải quyết. 3*](#_Toc154175567)

[*3. Nội dung, phạm vi thực hiện. 3*](#_Toc154175568)

[*4. Kết quả cần đạt. 3*](#_Toc154175569)

[*5. Bảng phân công nhiệm vụ 4*](#_Toc154175570)

[*CHƯƠNG II: MÔ TẢ CẤU TRÚC 5*](#_Toc154175571)

[*2.1. Danh sách liên kết đơn 5*](#_Toc154175572)

[*CHƯƠNG III: THIẾT KẾ 9*](#_Toc154175573)

[*3. Hàm cấu trúc thông tin cơ bản. 9*](#_Toc154175574)

[*3.1. Chức năng quản lý trà sữa: 10*](#_Toc154175575)

[*3.2. Chức năng quản lý loại trà sữa. 14*](#_Toc154175576)

[*3.3. Chức năng quản lý bán hàng 19*](#_Toc154175577)

[*CHƯƠNG IV: GIAO DIỆN 23*](#_Toc154175578)

[*4. Giao diện thiết kế 23*](#_Toc154175579)

[*4.1. Chức năng quản lý trà sữa: 23*](#_Toc154175580)

[*4.2. Chức năng quản lý loại trà sữa 24*](#_Toc154175581)

[*4.3. Chức năng quản lý hóa đơn 25*](#_Toc154175582)

[*CHƯƠNG V: KẾT LUẬN 26*](#_Toc154175583)

[*5.1. Ưu điểm và nhược điểm của phần mềm 26*](#_Toc154175584)

[*5.3. Hướng dẫn sử dụng 26*](#_Toc154175585)

# *CHƯƠNG I: GIỚI THIỆU*

## *Mục tiêu của báo cáo.*

*Mục tiêu của ứng dụng quản lý công việc này là giúp người dùng hiệu quả hóa quá trình quản lý và theo dõi các công việc của họ. Cung cấp một công cụ mạnh mẽ và linh hoạt để quản lý công việc và lịch làm, giúp người dùng tổ chức công việc của họ một cách hiệu quả và dễ dàng. Ngoài ra, tôi hướng đến việc tăng cường trải nghiệm người du thông qua giao diện thân thiện và tính năng đa dạng. Bằng cách này, chúng tôi hi vọng đóng góp giúp đỡ người dùng quản lý công việc và cuộc sống một cách hiệu quả.*

## *Những thách thức cần giải quyết.*

*Giao diện còn đơn giản và tính thẩm mỹ còn chưa cao. Một số chức năng còn chưa được phát triển một cách tối ưu nhất.*

## *Nội dung, phạm vi thực hiện.*

*Xây dựng thành công phần mềm quản lý lịch làm, tối ưu được thao tác khi sử dụng, cập nhật, thêm, xóa, sửa công việc một cách nhanh chóng và tiện lợi. Người dùng có thể quản lý công việc, ngày kết thúc và quản lý tình trạng công việc của người dùng theo ngày hoặc tháng. Với tính năng còn nhiều thiếu sót, phần mềm cung cấp đầy đủ tính năng cũng như thể hiện kiến thức lập trình của tôi thông qua quá trình học tập và tìm hiểu. Phạm vi thực hiện có thể áp dụng làm đồ án tham khảo cho sinh viên và phù hợp với những người dùng cá nhân cần quản lý một số lượng công việc vừa phải.*

## *Kết quả cần đạt.*

*• Xây dựng chức năng quản lý lịch làm như thêm, xóa, sửa, tìm kiếm, cập nhật công việc và trạng thái công việc*

*• Xây dựng một hệ thống quản lý công việc bao gồm: sắp xếp và cập nhật công việc, hiển thị công việc theo tên, ngày/tháng/năm và trạng thái*

*• Xây dựng một giao diện tối ưu, ưa nhìn, dễ dàng sử dụng.*

## *Bảng phân công nhiệm vụ*

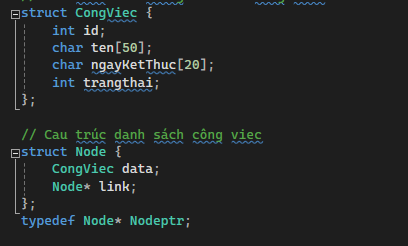
|  |  |  |  |
| --- | --- | --- | --- |
| ***STT*** | ***Họ tên*** | ***Nội dung*** | ***Ghi chú*** |
| *1* | *Hàng Gia Thuận* | * *Tạo cấu trúc cơ bản của CongViec* * *Tạo node “Node” chức năng lưu trữ, liên kết với node tiếp theo, tạo danh sách liên kết, dễ dàng thêm, xoá và duyệt qua các thông tin.* * *Xử lý chức năng quản lý, viết hàm quản lý công việc bao gồm: Thêm, xóa, sửa, tìm kiếm, sắp xếp và cập nhật danh sách công việc, bao gồm các hàm:* * *void them\_CongViecVaoDS(Nodeptr& list);* * *void sua\_CongViec(Nodeptr& list, char\* ten);* * *void sua\_Ten\_CongViec(Nodeptr& list, char\* ten);* * *void sua\_NgayKetThuc\_CongViec(Nodeptr& list);* * *void capNhatTrangThai(Nodeptr& list, char\* ten);* * *void xoa\_CongViec(Nodeptr& list, char\* ten);* * *void xoaToanBoDanhSach(Nodeptr& list);* * *void sapxep\_TheoTen(Nodeptr& list);* * *void sapxep\_TheoTrangThai(Nodeptr& list);* * *void sapxep\_TheoId(Nodeptr& list);* * *void sapxep\_TheoNgayKetThuc(Nodeptr& list);* * *void tim\_CongViec\_TheoTen(Nodeptr list, char\* ten);* * *void tim\_CongViec\_TheoId(Nodeptr list, int id);* * *void tim\_CongViec\_TheoNgayKetThuc(Nodeptr list, char\* ngayKetThuc);* * *void tim\_CongViec\_TheoTrangThai(Nodeptr list, int trangthai);* * *void xuat\_DanhSachCongViec(Nodeptr list);* * *void capNhat\_TrangThaiTuDong(Nodeptr& list);* |  |
| *2* | *Nguyễn Cảnh Thịnh* | * *Viết khai báo về các chức năng chính của chương trình và tạo menu thu gọn một vài chức năng chính bao gồm các hàm:* * *hàm main()* * *void sua\_CV\_menu(Nodeptr list, int luaChon);* * *void sapXep\_menu(Nodeptr& list, int luaChon);* * *void timKiem\_menu(Nodeptr list, int luaChon);* * *void pressAnyKey();* |  |

# *CHƯƠNG II: MÔ TẢ CẤU TRÚC*

## *2.1. Danh sách liên kết đơn*

*- Danh sách liên kết đơn (Single Linked List) là một cấu trúc dữ liệu động, nó là một danh sách mà mỗi phần tử đều liên kết với phần tử đúng sau nó trong danh sách. Mỗi phần tử (được gọi là một node hay nút) trong danh sách liên kết đơn là một cấu trúc có hai thành phần:*

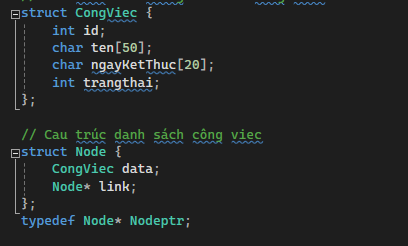
* *Thành phần dữ liệu: lưu thông tin về bản thân phần tử đó.*
* *Thành phần liên kết: lưu địa chỉ phần tử đứng sau trong danh sách, nếu phần tử đó là phần tử cuối cùng thì thành phần này bằng NULL.*
* *Đặc điểm của danh sách liên kết đơn do danh sách liên kết đơn là một cấu trúc dữ liệu động, được tạo nên nhờ việc cấp phát động nên nó có một số đặc điểm sau đây:*
* *Được cấp phát bộ nhớ khi chạy chương trình*
* *Có thể thay đổi kích thước qua việc thêm, xóa phần tử*
* *Kích thước tối đa phụ thuộc vào bộ nhớ khả dụng của RAM*
* *Các phần tử được lưu trữ ngẫu nhiên (không liên tiếp) trong RAM*
* *Và do tính liên kết của phần tử đầu và phần tử đứng sau nó trong danh sách liên kết đơn, nó có các đặc điểm sau:*
* *Chỉ cần nắm được phần tử đầu và cuối là có thể quản lý được danh sách*
* *Truy cập tới phần tử ngẫu nhiên phải duyệt từ đầu đến vị trí đó*
* *Chỉ có thể tìm kiếm tuyến tính một phần tử*
* *Cài đặt danh sách liên kết đơn.*
* *Tạo node: Danh sách liên kết đơn được tạo thành từ nhiều node, do đó, chúng ta sẽ cùng đi từ node trước. Một node gồm một thành phần là thành phần dữ liệu và thành phần liên kết. Thành phần dữ liệu có thể là kiểu dữ liệu có sẵn hoặc bạn tự định nghĩa (struct hay class). Thành phần liên kết là địa chỉ đương nhiên sẽ là con trỏ, con trỏ này trỏ đến node tiếp theo, do đó, con trỏ này là con trỏ trỏ vào một node.*

******

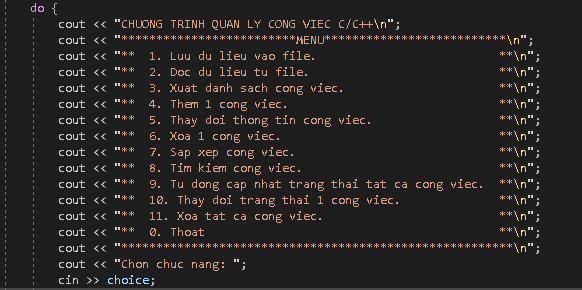
# *CHƯƠNG III: THIẾT KẾ*

## *Hàm cấu trúc thông tin cơ bản.*

* *Tạo cấu trúc cơ bản của “Công Việc”. Với cách sử dụng con trỏ trong “Node” giúp quản lí các phần tử trong danh sách một cách linh hoạt để thêm, xóa, sửa và sắp xếp thông tin công việc.*

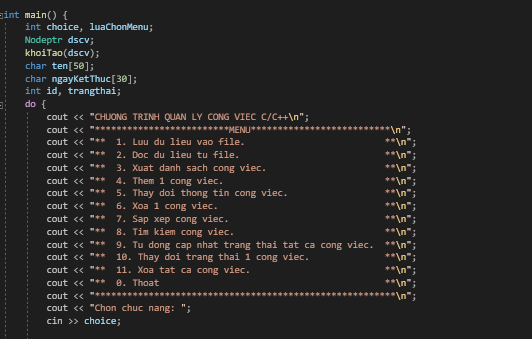
******

* *Chương trình bao gồm 8 chức năng chính: thêm, sửa, xoá, sắp xếp, tìm kiếm, cập nhật trạng thái, đọc/ghi file và thoát chương trình.Trong mỗi chức năng sẽ xử dụng hàm riêng để quản lý và xử lý.*
* ***Giao diện của menu phần mềm:***

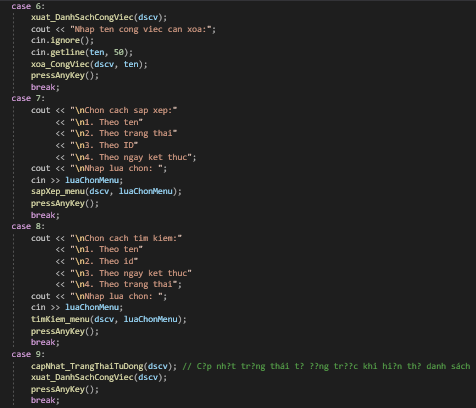
**

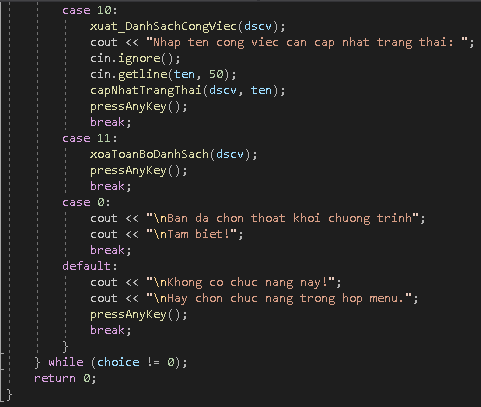
## *Chức năng quản lý công việc:*

* *Hàm main() khởi tạo các danh sách dữ liệu từ file CongViec. Sử dụng vòng lặp Do while để lặp các chức năng của chương trình cho đến khi người dùng thoát ra. Sử dụng lệnh switch để nhập chức năng menu, nhận lựa chọn từ người dùng.*

**

**

**

**

* *Chức năng thêm công việc:*

*Hàm them\_CongViecVaoDS và các hàm được sử dụng trong hàm đó:*

*Nodeptr tao\_Node\_CongViec(CongViec cv): Hàm này tạo một nút mới (Node) chứa thông tin về một công việc (CongViec cv). Nút được cấp phát động và được trả về.*

*Nodeptr them\_CongViec(Nodeptr& list, CongViec cv): Hàm này thêm một công việc mới vào đầu danh sách liên kết (list). Đầu tiên, nó tạo một nút mới với thông tin công việc (cv) bằng cách sử dụng hàm tao\_Node\_CongViec, sau đó nút mới này được thêm vào đầu danh sách và trở thành đầu danh sách mới.*

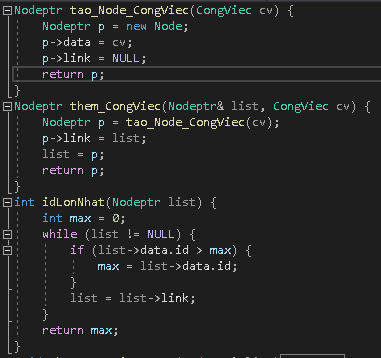
*int idLonNhat(Nodeptr list): Hàm này tìm và trả về ID lớn nhất trong danh sách công việc. Nó duyệt qua danh sách và so sánh ID của mỗi công việc để xác định ID lớn nhất.*

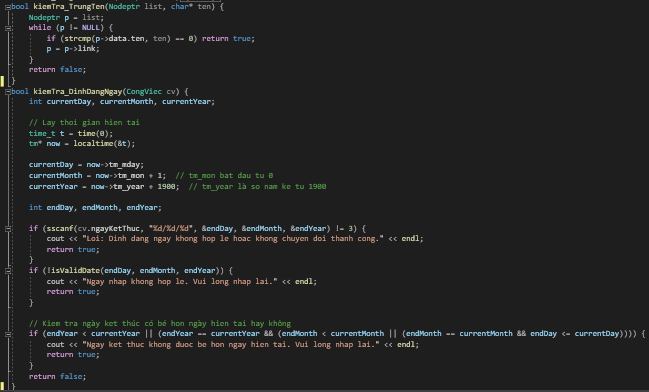
*void them\_CongViecVaoDS(Nodeptr& list): Hàm này thực hiện quy trình để người dùng nhập thông tin về một công việc mới và sau đó thêm công việc đó vào danh sách (list). Nó sử dụng các hàm khác như idLonNhat, kiemTra\_TrungTen, kiemTra\_DinhDangNgay để đảm bảo tính hợp lệ của thông tin nhập vào.*

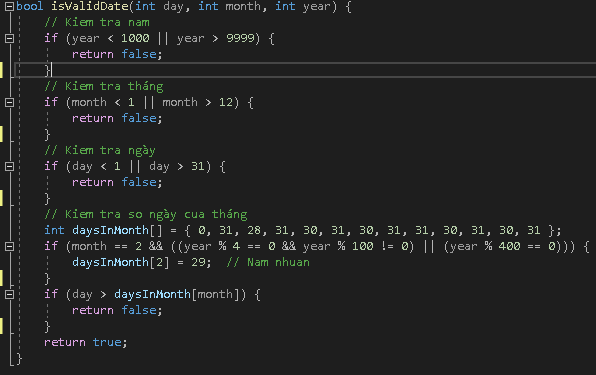
*bool kiemTra\_TrungTen(Nodeptr list, char\* ten): Hàm này kiểm tra xem tên công việc (ten) có trùng lặp trong danh sách công việc không.*

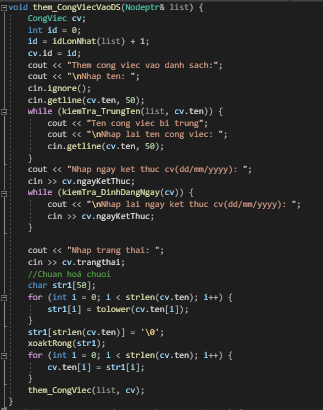
*bool kiemTra\_DinhDangNgay(CongViec cv): Hàm này kiểm tra xem định dạng ngày kết thúc của công việc có hợp lệ không. Nó sử dụng isValidDate để kiểm tra định dạng và tính hợp lệ của ngày.*

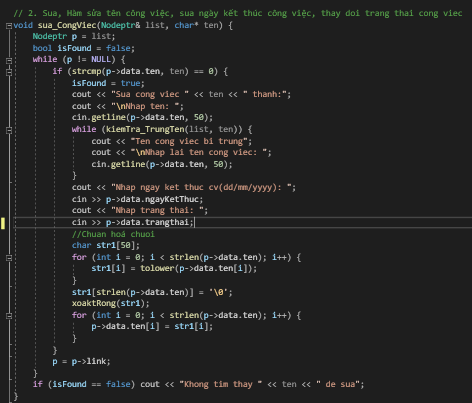
*bool isValidDate(int day, int month, int year): Hàm này kiểm tra xem một ngày cụ thể có hợp lệ không dựa trên ngày, tháng, và năm. Điều này bao gồm kiểm tra các ràng buộc như tháng không vượt quá 12, ngày không vượt quá số ngày trong tháng, năm phải nằm trong khoảng hợp lệ và năm nhuận.*

**

**

**

**

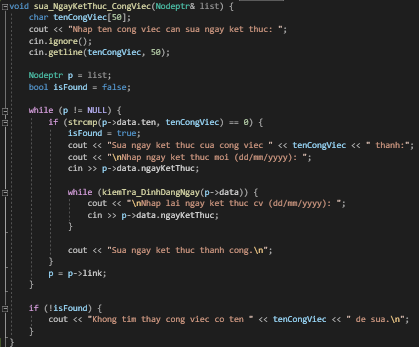
* *Chức năng sửa thông tin công việc:*

*Hàm sua\_CongViec thay đổi toàn bộ thông tin công việc gồm: tên, ngày kết thúc, trạng thái công việc.*

*Hàm sua\_Ten\_CongViec là hàm sử dụng cho chức năng sửa công việc: thay đổi tên 1 công việc bằng cách nhập tên công việc mà người dùng cần sửa.*

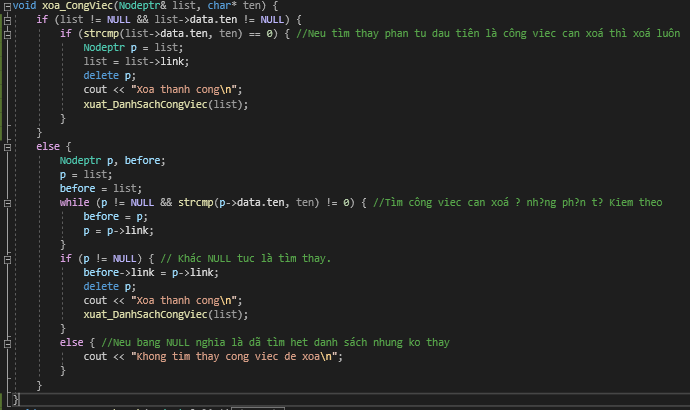
**

*Hàm sua\_NgayKetThuc\_CongViec là hàm sử dụng cho chức năng sửa công việc: thay đổi ngày kết thúc 1 công việc bằng cách nhập tên công việc cần sửa đổi ngày kết thúc.*

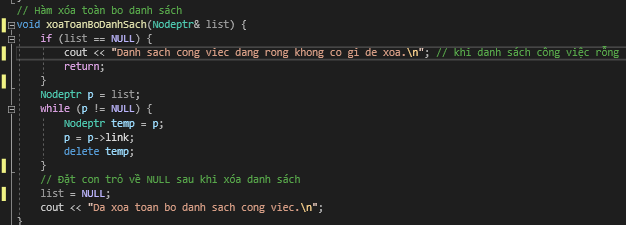
**

* *Chức năng xoá thông tin công việc*

*Hàm xoa\_CongViec là hàm sử dụng cho chức năng xoá: xoá 1 công việc khỏi danh sách công việc bằng cách nhập tên công việc đó.*

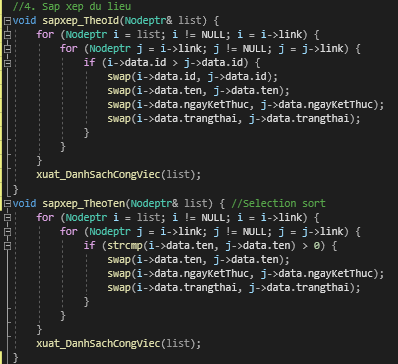
**

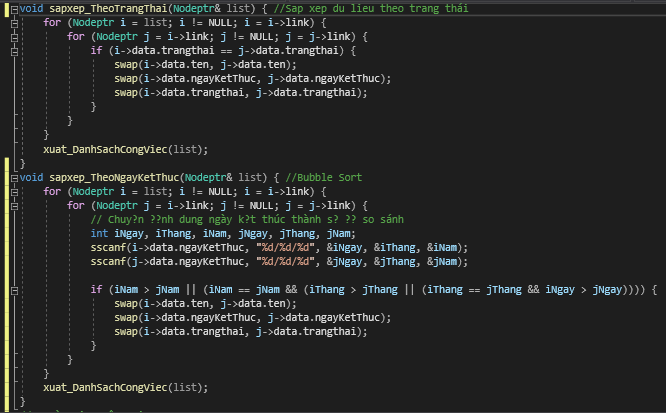
*Hàm xoaToanBoDanhSach là hàm sử dụng cho chức năng xoá: xoá toàn bộ danh sách công việc khi người dùng muốn giải phóng bộ nhớ hoặc muốn lập lại một lịch mới*

**

* *Chức năng sắp xếp danh sách công việc:*

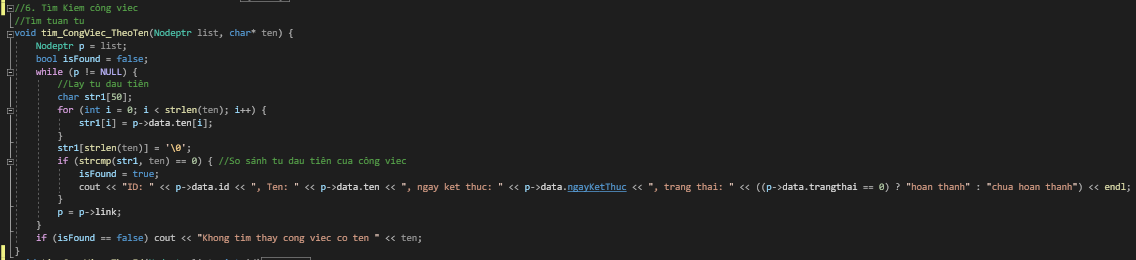
*Có 4 chức năng sắp xếp danh sách công việc là: sắp xếp theo ID, tên, ngày kết thúc và trạng thái công việc.*

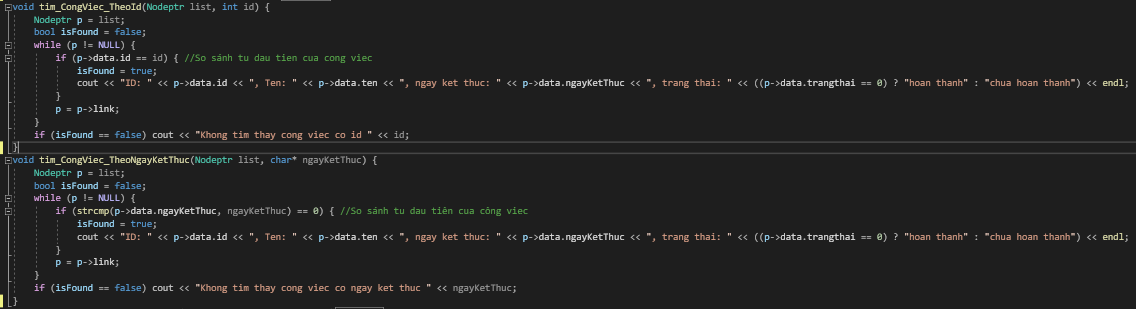
**

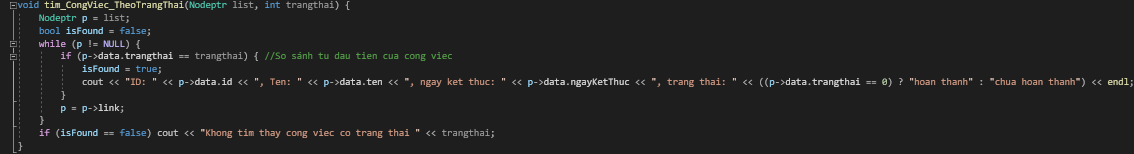
**

* *Chức năng tìm kiếm công việc trong danh sách:*

*Tương tự chức năng sắp xếp thì chức năng tìm kiếm cũng có 4 tiêu chí tìm kiếm là: ID, tên công việc, ngày kết thúc và trạng thái công việc.*

**

**

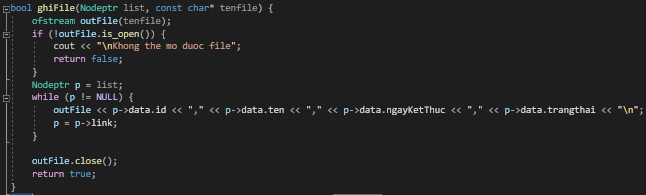
**

* *Các chức năng khác của chương trình:*

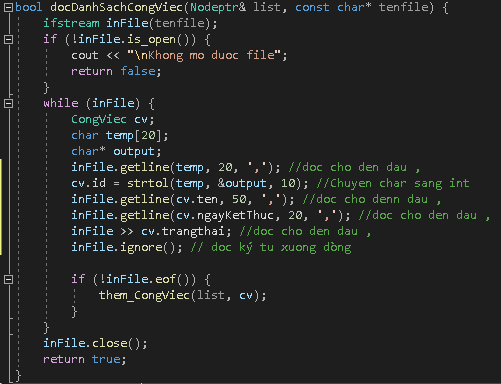
*Hàm xuat\_DanhSachCongViec: hiển thị ra màn hình danh sách toàn bộ công việc đã được thêm vào.*

**

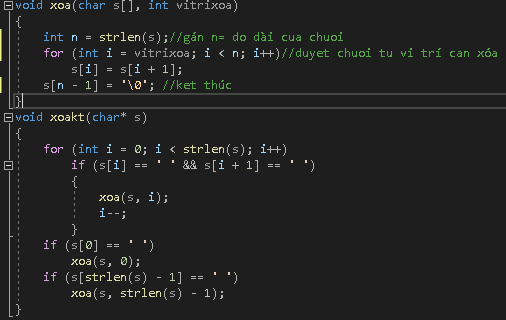
*Hàm ghiFile được dùng cho chức năng lưu trữ danh sách công việc đã được thêm vào.*

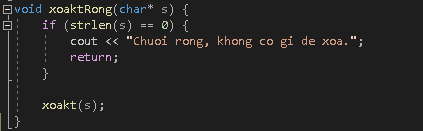
**

*Hàm docDanhSachCongViec dùng cho chức năng load danh sách công việc được lưu trong bộ nhớ.*

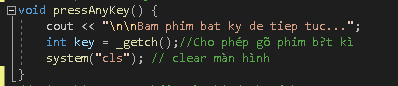
**

*Các hàm xoa, xoakt, xoaktRong được dùng trong các hàm thêm và sửa công việc với chức năng chuẩn hoá chuỗi khi nhập thông tin công việc.*

**

**

*Hàm pressAnyKey dùng để clear màn hình sau khi nhập lựa chọn chức năng trong menu.*

**

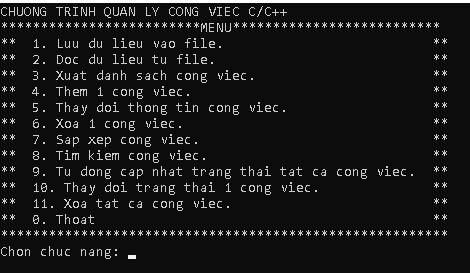
*Hàm capNhat\_TrangThaiTuDong tự động cập nhật trạng thái thành hoàn thành khi ngày hiện tại vượt quá ngày kết thúc trong danh sách công việc.*

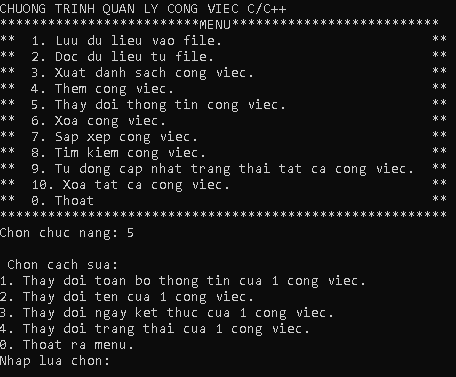
**

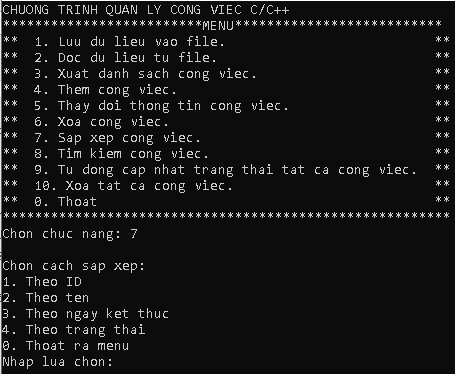
# *CHƯƠNG IV: GIAO DIỆN*

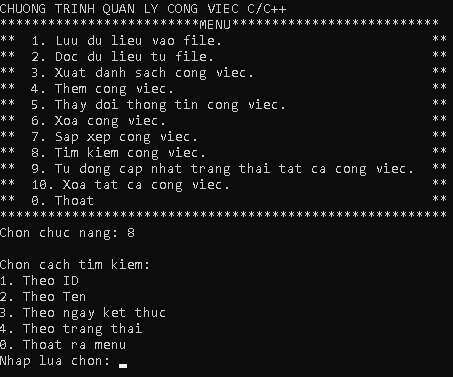
## *4. Giao diện thiết kế*

### *4.1. Chức năng quản lý công việc*

**

**

**

**

# *CHƯƠNG V: KẾT LUẬN*

## *5.1. Ưu điểm và nhược điểm của phần mềm*

* ***Ưu điểm:***

***Quản lý công việc hiệu quả:*** *Ứng dụng có khả năng quản lý công việc, giúp người dùng dễ dàng thêm, sửa, xoá công việc và theo dõi trạng thái của chúng.*

***Dễ sử dụng:*** *Giao diện menu và các chức năng được thiết kế đơn giản, dễ hiểu, giúp người dùng sử dụng mà không gặp khó khăn.*

***Tính linh hoạt:*** *Cung cấp các chức năng sắp xếp, tìm kiếm, và thay đổi trạng thái công việc, tạo sự linh hoạt trong việc quản lý công việc.*

***Lưu trữ dữ liệu:*** *Dữ liệu về công việc được lưu trữ và đọc ghi từ file, giúp người dùng giữ được thông tin công việc ngay cả sau khi tắt ứng dụng.*

***Phát triển dễ dàng:*** *Mã nguồn có tổ chức tốt, dễ bảo trì và phát triển thêm tính năng mới.*

* ***Nhược điểm:***

***Giao diện đồ họa giản dị:*** *Giao diện text-based có thể làm giảm trải nghiệm người dùng so với các ứng dụng có giao diện đồ họa hấp dẫn hơn.*

***Thiếu tính năng mở rộng:*** *Ứng dụng tập trung chủ yếu vào các chức năng cơ bản, có thể làm cho người dùng thiếu đi một số tính năng phức tạp hơn mà họ có thể cần.*

***Không có bảo mật cao:*** *Hiện vẫn chưa có cơ chế bảo mật mạnh mẽ nào được tích hợp, điều này có thể là một vấn đề nếu thông tin quan trọng được lưu trữ trong ứng dụng.*

***Chưa hỗ trợ đa người dùng:*** *Hiện tại, ứng dụng không hỗ trợ đa người dùng, điều này có thể làm giảm khả năng sử dụng cho các dự án cộng tác lớn.*

***Hạn chế về giao diện người dùng:*** *Mặc dù đơn giản và dễ sử dụng, nhưng giao diện dựa trên văn bản có thể làm cho trải nghiệm người dùng ít sinh động và thu hút hơn so với các ứng dụng có giao diện đồ họa đẹp mắt.*

***5.2. Đề xuất hướng phát triển***

* *Thời gian tới sẽ phát triển thêm một số chức năng khác, cũng như cải thiện giao diện người dùng tích hợp giao diện màu sắc, tính năng mở rộng, bảo mật và quyển riêng tư, hỗ trợ ngôn ngữ đa nền tảng, hỗ trợ đa người dùng và cộng tác.*

## *5.3. Hướng dẫn sử dụng*

* *Yêu cầu cài đặt môi trường: Cài đặt visual studio, ngôn ngữ lập trình C++, trình biên dịch C++ trên các hệ điều hành khác nhau.*
* *Chạy chương trình và thao tác theo chức năng tương ứng trên menu.*

***TÀI LIỆU THAM KHẢO***

*Dưới đây là một số nguồn tài liệu và hướng dẫn mà em đã tham khảo trong quá trình xây dựng phần mềm:*

* *Tài liệu hướng dẫn lý thuyết và thực hành Nhập môn cấu trúc dữ liệu STU (2022).*
* *Tài liệu hướng dẫn lý thuyết và thực hành Cấu trúc dữ liệu và giải thuật STU (2022).*
* *Một số trang web tham khảo tài liệu:*
* [*https://vi.wikipedia.org/wiki/C%2B%2B*](https://vi.wikipedia.org/wiki/C%2B%2B)