Lời cảm ơn !

Để bài nghiên cứu khoa khọc này được kết quả tốt đẹp , em đã nhận được nhiều sự giúp đỡ của các thầy cô trong Khoa Công Nghệ Thông Tin – Trường cao đẳng hữu nghị Việt Hàn . Cho phép em gửi lời cảm ơn sâu sắc đến các thầy cô đã giúp đỡ em trong quá trình hoàn thành đề tài bà đặt biệt là thầy Nguyễn Thanh Cẩm – Giảng Viên bộ môn Lập Trình Cơ Bản của trường đã quan tâm và giúp đỡ , giải đáp thắc mắc cho em trong quá trình hoàn thành đề tài này .

Với điều kiện về thời gian cũng như kinh nghiệm còn hạn chế của một học viên , luận văn này không khỏi có những thiếu sót . Em rất mong nhận được sự chỉ bảo , đóng góp ý kiến của thầy cô để em có điều kiện được bổ sung , nâng cao ý thức của mình phục vụ công tác thực tế sau này !

Em xin chân thành cảm ơn !

Đà Nẵng , Thứ Tư ,Ngày 22 Tháng 5 Năm 2019

Sinh Viên

Nguyễn Thanh Phương .

Lời mở đầu !

Ngày nay , Đất nước Việt Nam đang trên đà phát triển . Nhà nước chú trọng đầu tư vào công nghệ phần mềm . Do sự phát triển ngày càng nhanh của xã hội thôi thúc mỗi người chúng ta phải làm được điều gì đó cho xã hội . Máy tính – Công cụ sẽ hỗ trợ đắc lực cho con người làm được điều đó thông qua ngôn ngữ lập trình . Hầu hết các chương trình ứng dụng được viết trên ngôn ngữ bậc cao nhiều. Với ngôn ngữ bậc thấp như C, Quản lý quán café bằng ngôn ngữ C tuy nó không được ứng dụng rộng rãi và cũng không có những ưu thế so với những ngôn ngữ bậc cao hơn nhưng nó sẽ giúp tạo tiền đề để hiểu những việc do máy tính tự thực hiện khi ta viết bằng ngôn ngữ bậc cao .

**MỤC LỤC**

**LỜI CẢM ƠN 1**

**MỞ ĐẦU 2**

**MỤC LỤC 3**

**CHƯƠNG I : THÔNG TIN CHUNG VỀ ĐỀ TÀI 5**

**CHƯƠNG II : THUYẾT MINH ĐỀ TÀI 6**

1. **Đặt vấn đề 6**
   1. **Tóm lược những nghiên cứu trongvà ngoài nước 6**
   2. **Tính cấp thiết của đề tài 6**
   3. **Mục tiêu nghiên cứu 6**
   4. **Mục đích nghiên cứu 6**
2. **Nội dung nghiên cứu 7**
3. **Đối tượng và phạm vi nghiên cứu 7**
   1. **Đối tượng nghiên cứu 7**
   2. **Phạm vi nghiên cứu 7**

**CHƯƠNG II : CƠ SỞ LÝ THUYẾT 8**

1. **Những kiến thức cơ sở về form 8**
   1. **Khái niệm Form 8**
   2. **Các đặc trưng của thuật toán trong form 8**
   3. **Tổng quang về mô phỏng thuật toán 8**
2. **Phương pháp nghiên cứu 9**
   1. **Phương pháp nghiên cứu tài liệu 9**
   2. **Phương pháp quan sát 9**
   3. **Phương pháp nghiên cứu các sản phẩm**

**hoạt động 9**

* 1. **Phương pháp thu thập số liệu từ những**

**thực nghiệm 9**

* 1. **Phương pháp chọn mẫu 10**

1. **Tổng quan về ngôn ngữ C 10**
   1. **Định nghĩa 10**
   2. **Lịch sử phát triển 10**
   3. **Đặc trưng của ngôn ngữ C 11**
   4. **Các thư viện trong C 11**
   5. **Trình biên dịch trong C 12**

**CHƯƠNG III : MÔ PHỎNG CÁC THUẬT TOÁN TẠO FORM 13**

1. **Thuật toán 13**
2. **Thuật toán tạo một danh sách liên kết 13**
3. **Thuật toán nhận biết khi di chuyển chuột vào**

**vị trí trên màn hình 18**

1. **Thuật toán tạo nên một textbox 19**
2. **Thuật toán tạo nên một List Box 20**
3. **Thuật toán viết chữ trong đồ họa 21**
4. **Thuật toán sử dụng list box 23**
5. **Thuật toán đọc dữ liệu từ file .txt 24**
6. **Thuật toán ghi file 26**
7. **Thuật toán xóa mảng chuỗi của những mảng cấp**

**phát động 27**

1. **Thuật toán lấy giờ thêm vào danh sách liên kết 28**
2. **Biểu mẫu ứng dụng thuật toán 28**
3. **Đoạn chương trình đọc file 29**
4. **Tạo form order thông qua các thuật toán trên 31**
5. **Xuất order ứng dụng danh sách liên kết 32**
6. **Một form thêm thông tin của một nhân viên 33**
7. **Ứng dụng các thuật toán làm mới , buton, … để tạo nên**

**một form xóa nhân viên 34**

1. **Ứng dụng các thuật toán trên chế tạo form đổi mật khẩu 35**
2. **Ứng dụng thuật toán làm nên form nhập kho trong chế độ**

**đồ họa 36**

CHƯƠNG 1 : THÔNG TIN CHUNG VỀ ĐỀ TÀI

1. Tên đề tài: SỬ DỤNG NGÔN NGỮ C XÂY DỰNG CHƯƠNG TRÌNH

QUẢN LÝ QUÁN CAFÉ

1. Mã số đề tài :
2. Chủ nhiệm đề tài (Nhóm trưởng): Nguyễn Thanh Phương

Khoa: Công Nghệ Thông Tin

Lớp: K1205A

1. Đơn vị chủ trì :
2. Danh sách sinh viên tham gia nghiên cứu đề tài :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TT** | **Tên sinh viên** | **Mã sinh viên** | **Khoa** | **Thực hiện số chương của báo cáo** |
| 1 | Nguyễn Thanh Phương | K12C05092 | Công Nghệ Thông Tin |  |

1. Thời gian thực hiện được phê duyệt :
2. Thời gian kết thúc thực tế :
3. Kinh phí thực hiện đề tài :

CHƯƠNG 2 : THUYẾT MINH ĐỀ TÀI

1. Đặt vấn đề :
2. Tóm lược những nghiên cứu trong và ngoài nước

Hiện nay , Ngành công nghệ thông tin đang là một ngành phổ biến đang được nhà nước rất quan tâm . là động lực quan trọng trong sự nghiệp công nghiệp hóa , hiện đại hóa đất nước . Bây giờ có rất nhiều ngôn ngữ lập trình khác nhau . Việc ứng dụng một demo chương trình cấu trúc dạng form trong vấn đề dạy học là một việt hết sức cần thiết cho người mới bắt đầu.

1. Tính cấp thiết của đề tài :

Một người có kiến thức cơ bản về ngôn ngữ lập trình .Nếu được hỏi nếu muốn làm một ứng dụng hay một chương trình phần mềm chạy cụ thể bạn sẽ chọn ngôn ngữ gì để viết ??? một danh sách các ngôn ngữ lập trình được liệt kê ra như java , C# , C++,… Trong đó chăc chắc không ai lập trình một ngôn ngữ C nào cả . Vì sao vậy ??? Dễ hiểu thôi bởi vì C là ngôn ngữ cấu trúc, nó khác với những ngôn ngữ kể trên và có phần tương tự với ngôn ngữ bậc thấp (Hợp ngữ ) , nó không có một công cụ tự động hóa nào làm những phần cơ bản một form trong một chương trình khả dụng như button , checkbox, listbox,text box, ..ta phải bắt tay tự làm mọi thứ từ việc cơ bản đến nâng cao hay trong C nếu muốn lưu dữ liệu ra tệp chủ yếu là ra file .txt để quản lý trong khi hiện nay đã có SQL quản lí nhanh và gọn hơn nhưng tôi vẫn muốn làm một ứng dụng minh họa quán café bằng ngôn ngữ C thì lí do của tôi là gì? Dễ hiểu nếu tôi cho một ví dụ là : một phương trình bậc hai có thể giải trên máy tính CASIO bằng cách bấm những con số đơn giản vào thì vì sao nhà trường vẫn phải bắt chúng ta giải trên giấy từng bước một , rườm rà , khó hiểu . nếu bạn hiểu được lí do của ví dụ trên thì đó cũng là tính cấp thiết của đề tài và cần cho những ai !

1. Mục tiêu nghiên cứu:

Xây dựng thành công ứng dụng minh họa quản lý quán café .

1. Mục đích nghiên cứu :

* Hiểu được quy trình hoạt động của các nút button, Listbox , Textbox, trong ngôn ngữ bậc cao và sự liên kết dữ liệu hay ràng buộc dữ liệu từ chúng .
* Có thêm kinh nghiệm để lập trình một phần mềm .
* Hiểu được quy trình hoạt động của các form trong ngôn ngữ bậc thấp , quy trình cơ bản của việc làm một chương trình ứng dụng .
* Hiểu biết cơ bản về một số việc tự động hóa mà máy tính tự làm trên ngôn ngữ bậc cao.

1. Nội dung nghiên cứu :

* Nghiên cứu xử lí đồ họa trong C.
* Đọc,In và xử lí file trong C bằng tệp đuôi .txt .
* ứng dụng những thuật toán cơ bản vào chương trình như sắp xếp , tìm kiếm , các thuật toán thêm , sửa , xóa , đổi chỗ ...
* Ứng dụng danh sách liên kết (Cấu trúc dữ liệu ) trong chương trình .
* Mảng động và xử lí thời gian .
* Liên kết dữ liệu giữa các kiểu cấu trúc .
* Những thuật toán khác .

1. Đối tượng và phạm vi nghiên cứu
   1. Đối tượng nghiên cứu :

Những nhân viên và chủ quán ở một số quán café lân cận trường và một số nơi khác

* 1. Phạm vi nghiên cứu :
* **-** Địa điểm nghiên cứu : Các quán café gần trường Cao Đẳng Việt - Hàn .
* **-** Hoạt động được nghiên cứu : cách thức quản lí thu chi và những nhu cầu của người quản lý trong một quán café .
* **-** Thời gian nghiên cứu : từ tháng 4🡪5/2019 .

CHƯƠNG II : CƠ SỞ LÝ THUYẾT

1. Những kiến thức cơ sở về form :
   1. Khái niệm Form :
2. Windows Forms (WinForms) là một [thư viện lớp](https://en.wikipedia.org/wiki/Library_(computing))[đồ họa](https://en.wikipedia.org/wiki/Graphical_user_interface) (GUI) được bao gồm như một phần của [Microsoft](https://en.wikipedia.org/wiki/Microsoft)[.NET Framework](https://en.wikipedia.org/wiki/.NET_Framework) hoặc [Mono Framework](https://en.wikipedia.org/wiki/Mono_Framework) , cung cấp một nền tảng để viết các ứng dụng khách phong phú cho máy tính để bàn, máy tính xách tay và máy tính bảng. Mặc dù được coi là sự thay thế cho [Thư viện lớp nền tảng Microsoft](https://en.wikipedia.org/wiki/Microsoft_Foundation_Class_Library) dựa trên [C ++](https://en.wikipedia.org/wiki/C%2B%2B) trước đó và phức tạp hơn , nhưng nó không cung cấp một mô hình tương đương và chỉ hoạt động như một nền tảng cho tầng giao diện người dùng trong một giải pháp [đa tầng](https://en.wikipedia.org/wiki/Multitier_architecture) .
3. Một ứng dụng Windows Forms là một [ứng dụng hướng sự kiện](https://en.wikipedia.org/wiki/Event-driven_programming) được hỗ trợ bởi Microsoft [.NET Framework](https://en.wikipedia.org/wiki/.NET_Framework) . Không giống như một [chương trình hàng loạt](https://en.wikipedia.org/wiki/Batch_processing) , nó dành phần lớn thời gian của mình chỉ đơn giản là chờ người dùng làm gì đó, chẳng hạn như điền vào [hộp văn bản](https://en.wikipedia.org/wiki/Text_box) hoặc nhấp vào [nút](https://en.wikipedia.org/wiki/Push-button) .
4. Tại sự kiện [Microsoft Connect](https://en.wikipedia.org/wiki/Microsoft_Connect) vào ngày 4 tháng 12 năm 2018, [Microsoft](https://en.wikipedia.org/wiki/Microsoft) đã công bố phát hành Windows Forms dưới dạng dự án nguồn mở trên [GitHub](https://github.com/dotnet/winforms) . Nó được phát hành theo [Giấy phép MIT](https://en.wikipedia.org/wiki/MIT_License) . Với phiên bản này, Windows Forms đã trở nên khả dụng cho các dự án nhắm mục tiêu khung [.NET Core](https://en.wikipedia.org/wiki/.NET_Core) . Tuy nhiên, khung vẫn chỉ khả dụng trên nền tảng Windows và việc triển khai WinForms chưa hoàn tất [của Mono](https://en.wikipedia.org/wiki/Mono_Framework) vẫn là triển khai đa nền tảng duy nhất.
   1. Các đặc trưng của thuật toán trong form :

- Có tính đơn định: ở mỗi bước của thuật toán, các thao tác phải hết sức rõ ràng, không gây nên sự nhọc nhằng lộn xộn đa nghĩa, thực hiện đúng các bước của thuật toán thì với một bộ dữ liệu vào chỉ cho duy nhất một kết quả ra.

- Có tính dừng: thuật toán không được rơi vào quá trình vô hạn, phải dừng lại thông qua một số sự kiện cho phép .

- Có tính đúng: sau khi thực hiện tất cả các bước của thuật toán theo đúng quy trình đã định, ta đạt được kết quả mong muốn với mọi bộ dữ liệu đầu vào. Kết quả đó được kiểm chứng bằng yêu cầu bài toán.

- **Tính hiệu quả**: Ít tốn kém về số lượng thao tác, về vùng nhớ, về thời gian

* 1. Tổng quang về mô phỏng thuật toán :

1. Bước 1: Cho chương trình hoặc đoạn chương trình đi vào trong vòng lặp vô hạn .
2. Bước 2 : Bắt các sự kiện trong vòng lặp vô hạn .
3. Bước 3 : Ấn vào một hoặc một số sự kiện cho trước buộc ra khỏi vòng lặp .
4. Phương pháp nghiên cứu :
   1. Phương pháp nghiên cứu tài liệu :
   2. Phương pháp quan sát :
      1. Khái niệm :

**Quan sát là phương pháp** tri giác có mục đích, có kế hoạch một sự kiện, hiện tượng, quá trình (hay hành vi cử chỉ của con người) trong những hoàn cảnh tự nhiên khác nhau nhằm thu thập những số liệu, sự kiện cụ thể đặc trưng cho quá trình diễn biến của sự kiện, hiện tượng đó.

* + 1. Ứng dụng phương pháp quan sát trong đề tài :

+ Phương pháp phân tích các sản phẩm hoạt động : Quan sát những ứng dụng ngoài thực tế trên winform như các cửa sổ trong windows .

* 1. Phương pháp nghiên cứu các sản phẩm hoạt động :
     1. Khái niệm :

Phương pháp nghiên cứu sản phẩm hoạt động là phương pháp mà nhà nghiên cứu sẽ thông qua các sản phẩm mà đối tượng được nghiên cứu (học sinh, giáo viên, cán bộ giáo dục,…) tạo ra như bài làm, bài chấm vở ghi, bài soạn, sổ sách, nhật ký, các sáng tạo văn học nghệ thuật, sản phẩm lao động, học tập,… để tìm hiểu tính chất, đặc điểm của con người và hoạt động tạo ra các sản phẩm đó.

* + 1. Ứng dụng :

1. Nghiên cứu bài tập Quản Lý Sinh Viên tại Website : <https://nguyenvanhieu.vn/chuong-trinh-quan-ly-sinh-vien-su-dung-struct/>
2. Nghiên cứu các cửa sổ trên windows từ đó suy ra cách thức hoạt động của chúng .
   1. Phương pháp thu thập số liệu từ những thực nghiệm :
      1. Khái niệm :

Trong phương pháp nầy, số liệu được thực hiện bằng cách quan sát, theo dõi, đo đạc qua các thí nghiệm. Các thí nghiệm trong lĩnh vực khoa học tự nhiên, vật lý, hóa học, kỹ thuật, nông nghiệp, kể cả xã hội thường được thực hiện trong phòng thí nghiệm, nhà lưới, ngoài đồng và cộng đồng xã hội. Để thu thập số liệu, các nhà NCKH thường đặt ra các biến để quan sát và đo đạc (thu thập số liệu). Các nghiệm thức trong thí nghiệm (có những mức độ khác nhau) thường được lặp lại để làm giảm sai số trong thu thập số liệu.

* + 1. Ứng dụng :

1. Nghiên cứu tỉ lệ cân đối cửa các ô Button , Text Box List Box sao cho phù hợp với người sử dụng mà vẫn nhìn rõ được bản chất của chính nó .
   1. Phương pháp chọn mẫu :
      1. Khái niệm :

Phương pháp chọn mẫu là không tiến hành điều tra hết toàn bộ các đơn vị của tổng thể, mà chỉ điều tra trên 1 số đơn vị nhằm để tiết kiệm thời gian, công sức và chi phí. Từ những đặc điểm và tính chất của mẫu ta có thể suy ra được đặc điểm và tính chất của cả tổng thể đó. Vấn đề quan trọng nhất là đảm bảo cho tổng thể mẫu phải có khả năng đại diện được cho tổng thể chung.

* + 1. Ứng dụng :

Điều tra một phần cơ bản của quy cách hoạt động của form và suy ra một số hoạt động khác tương ứng .

1. Tổng quan về ngôn ngữ C :
   1. Định nghĩa :

C ban đầu được phát triển tại [Bell Labs](https://en.wikipedia.org/wiki/Bell_Labs) bởi [Dennis Ritchie](https://en.wikipedia.org/wiki/Dennis_Ritchie) trong khoảng thời gian từ năm 1972 đến năm 1973 để làm cho các tiện ích chạy trên [Unix](https://en.wikipedia.org/wiki/Unix) . Sau đó, nó đã được áp dụng để triển khai lại kernel của hệ điều hành Unix. Trong những năm 1980, C dần dần trở nên phổ biến. Ngày nay, nó là một trong những [ngôn ngữ lập trình được sử dụng rộng rãi nhất](https://en.wikipedia.org/wiki/Measuring_programming_language_popularity) , với [trình biên dịch](https://en.wikipedia.org/wiki/Compiler) C từ các nhà cung cấp khác nhau có sẵn cho phần lớn các [kiến trúc máy tính](https://en.wikipedia.org/wiki/Computer_architecture) và hệ điều hành hiện có. C đã được chuẩn hóa bởi [Viện Tiêu chuẩn Quốc gia Hoa Kỳ](https://en.wikipedia.org/wiki/American_National_Standards_Institute) từ năm 1989 (xem [ANSI C](https://en.wikipedia.org/wiki/ANSI_C) ) và sau đó bởi [tổ chức tiêu chuẩn quốc tế](https://en.wikipedia.org/wiki/International_Organization_for_Standardization) .

* 1. Lịch sử phát triển :

Nguồn gốc của C gắn liền với sự phát triển của [hệ](https://en.wikipedia.org/wiki/Unix) điều hành [Unix](https://en.wikipedia.org/wiki/Unix) , ban đầu được triển khai bằng [ngôn ngữ lắp ráp](https://en.wikipedia.org/wiki/Assembly_language) trên [PDP-7](https://en.wikipedia.org/wiki/PDP-7) của Dennis Ritchie và Ken Thompson, kết hợp một số ý tưởng từ các đồng nghiệp. Cuối cùng, họ quyết định chuyển hệ điều hành sang [PDP-11](https://en.wikipedia.org/wiki/PDP-11) . Phiên bản PDP-11 ban đầu của Unix được phát triển bằng ngôn ngữ lắp ráp. Thompson cần một ngôn ngữ lập trình để tạo ra các tiện ích. Lúc đầu, anh đã cố gắng tạo ra một trình biên dịch Fortran, nhưng ngay sau đó đã từ bỏ ý tưởng và tạo ra một ngôn ngữ mới, [B](https://en.wikipedia.org/wiki/B_(programming_language)) , phiên bản đơn giản hóa của [BCPL](https://en.wikipedia.org/wiki/BCPL) của Thompson . Tuy nhiên, một số tiện ích được viết bằng B vì B quá chậm và B không thể tận dụng các tính năng PDP-11 như địa chỉ [byte](https://en.wikipedia.org/wiki/Byte) .

Năm 1972, Ritchie bắt đầu cải tiến B, kết quả là tạo ra một ngôn ngữ mới C. Trình biên dịch C và một số tiện ích được tạo bởi C đã được đưa vào [Phiên bản 2 Unix](https://en.wikipedia.org/wiki/Version_2_Unix) . Tại [phiên bản 4 Unix](https://en.wikipedia.org/wiki/Version_4_Unix) được phát hành vào tháng 11 năm 1973, [nhân](https://en.wikipedia.org/wiki/Kernel_(operating_system))[Unix](https://en.wikipedia.org/wiki/Unix) được triển khai lại rộng rãi bởi C. Đến thời điểm này, ngôn ngữ C đã có được một số tính năng mạnh mẽ như các loại. struct

Unix là một trong những hạt nhân hệ điều hành đầu tiên được triển khai bằng ngôn ngữ khác ngoài [lắp ráp](https://en.wikipedia.org/wiki/Assembly_language) . Các trường hợp trước đó bao gồm hệ thống [Multics](https://en.wikipedia.org/wiki/Multics) (được viết bằng [PL / I](https://en.wikipedia.org/wiki/PL/I) ) và [Chương trình điều khiển chính](https://en.wikipedia.org/wiki/Burroughs_MCP) (MCP) cho [Burroughs B5000](https://en.wikipedia.org/wiki/Burroughs_large_systems) (được viết bằng [ALGOL](https://en.wikipedia.org/wiki/ALGOL) ) vào năm 1961. Vào khoảng năm 1977, Ritchie và [Stephen C. Johnson](https://en.wikipedia.org/wiki/Stephen_C._Johnson) đã thay đổi thêm ngôn ngữ để tạo điều kiện cho tính di động của hệ điều hành Unix. [Trình biên dịch C di động](https://en.wikipedia.org/wiki/Portable_C_Compiler) của Johnson là cơ sở cho một số triển khai C trên các nền tảng mới.

* 1. Đặc trưng của ngôn ngữ C :

C là một [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh" \o "Ngôn ngữ lập trình) tương đối nhỏ gọn vận hành gần với phần cứng và nó giống với ngôn ngữ [Assembler](https://vi.wikipedia.org/wiki/H%E1%BB%A3p_ng%E1%BB%AF" \o "Hợp ngữ) hơn hầu hết các [ngôn ngữ bậc cao](https://vi.wikipedia.org/w/index.php?title=Ng%C3%B4n_ng%E1%BB%AF_b%E1%BA%ADc_cao&action=edit&redlink=1" \o "Ngôn ngữ bậc cao (trang chưa được viết)). Hơn thế, C đôi khi được đánh giá như là "có khả năng di động", cho thấy sự khác nhau quan trọng giữa nó với [ngôn ngữ bậc thấp](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh_b%E1%BA%ADc_th%E1%BA%A5p" \o "Ngôn ngữ lập trình bậc thấp) như là Assembler, đó là việc mã C có thể được dịch và thi hành trong hầu hết các máy tính, hơn hẳn các ngôn ngữ hiện tại trong khi đó thì [Assembler](https://vi.wikipedia.org/wiki/H%E1%BB%A3p_ng%E1%BB%AF" \o "Hợp ngữ) chỉ có thể chạy trong một số máy tính đặc biệt. Vì lý do này C được xem là [ngôn ngữ bậc trung](https://vi.wikipedia.org/w/index.php?title=Ng%C3%B4n_ng%E1%BB%AF_b%E1%BA%ADc_trung&action=edit&redlink=1" \o "Ngôn ngữ bậc trung (trang chưa được viết)).

* Một [ngôn ngữ cốt lõi](https://vi.wikipedia.org/w/index.php?title=Ng%C3%B4n_ng%E1%BB%AF_c%E1%BB%91t_l%C3%B5i&action=edit&redlink=1) đơn giản, với các chức năng quan trọng chẳng hạn như là những hàm hay việc xử lý tập tin sẽ được cung cấp bởi các bộ [thư viện các thủ tục](https://vi.wikipedia.org/w/index.php?title=Th%C6%B0_vi%E1%BB%87n_(khoa_h%E1%BB%8Dc_m%C3%A1y_t%C3%ADnh)&action=edit&redlink=1).
* Tập trung trên mẫu hình [lập trình thủ tục](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_th%E1%BB%A7_t%E1%BB%A5c), với các phương tiện lập trình theo kiểu [cấu trúc](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_c%E1%BA%A5u_tr%C3%BAc).
* Một hệ thống kiểu đơn giản nhằm loại bỏ nhiều phép toán không có ý nghĩa thực dụng.
* Dùng ngôn ngữ [tiền xử lý](https://vi.wikipedia.org/wiki/Ti%E1%BB%81n_x%E1%BB%AD_l%C3%BD), tức là các câu lệnh [tiền xử lý C](https://vi.wikipedia.org/wiki/Ti%E1%BB%81n_x%E1%BB%AD_l%C3%BD_C), cho các nhiệm vụ như là định nghĩa các [macro](https://vi.wikipedia.org/w/index.php?title=Macro&action=edit&redlink=1) và hàm chứa nhiều tập tin [mã nguồn](https://vi.wikipedia.org/wiki/M%C3%A3_ngu%E1%BB%93n) (bằng cách dùng câu lệnh tiền xử lý dạng #include chẳng hạn).
* Mức thấp của ngôn ngữ cho phép dùng tới [bộ nhớ máy tính](https://vi.wikipedia.org/wiki/B%E1%BB%99_nh%E1%BB%9B) qua việc sử dụng kiểu dữ liệu pointer.
* Số lượng từ khóa rất nhỏ gọn.
* Các [tham số](https://vi.wikipedia.org/wiki/Tham_s%E1%BB%91_(khoa_h%E1%BB%8Dc_m%C3%A1y_t%C3%ADnh)) được đưa vào các hàm bằng giá trị, không bằng địa chỉ.
* Hàm các con trỏ cho phép hình thành một nền tảng ban đầu cho [tính đóng](https://vi.wikipedia.org/w/index.php?title=T%C3%ADnh_%C4%91%C3%B3ng_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh&action=edit&redlink=1) và [tính đa hình](https://vi.wikipedia.org/w/index.php?title=T%C3%ADnh_%C4%91a_h%C3%ACnh&action=edit&redlink=1).
* Hỗ trợ các [bản ghi](https://vi.wikipedia.org/w/index.php?title=B%E1%BA%A3n_ghi_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)&action=edit&redlink=1) hay các kiểu dữ liệu kết hợp do người dùng từ khóa định nghĩa struct cho phép các dữ liệu liên hệ nhau có thể được tập hợp lại và được điều chỉnh như là toàn bộ
  1. Các thư viện trong C :
     1. stdio.h

stdio là một thư viện cơ bản trong C với các trình nhập xuất và tính toán các dữ liệu cơ bản . Trong chương trình này thư viện stdio.h có các tác dụng cơ bản như : Vòng lặp , in và đọc file , … và một số các nội dung khác.

* + 1. conio.h

conio.h là thư viện không thể thiếu trong C . Thư viện chứa các hàm vào ra trong chế độ DOS

* + 1. graphics.h

Đồ họa C sử dụng các hàm Graphics.h hoặc WinBGIM có thể được sử dụng để vẽ các hình dạng khác nhau, hiển thị văn bản trong các phông chữ khác nhau, thay đổi màu sắc và nhiều hơn nữa. Sử dụng các chức năng của Graphics.h trong trình biên dịch Turbo C, bạn có thể tạo các chương trình đồ họa, hoạt hình, dự án và trò chơi. Bạn có thể vẽ hình tròn, đường thẳng, hình chữ nhật, thanh và nhiều hình hình học khác. Bạn có thể thay đổi màu sắc của chúng bằng cách sử dụng các chức năng có sẵn và điền vào chúng. Sau đây là danh sách các chức năng của tệp tiêu đề Graphics.h. Mọi hàm được thảo luận với các đối số cần thiết, mô tả của nó, các lỗi có thể xảy ra trong khi sử dụng hàm đó và chương trình đồ họa C mẫu với đầu ra của nó.stdlib.h

* + 1. windows.h

windows.h là tệp tiêu đề dành riêng cho Windows cho các ngôn ngữ lập trình C và C ++ chứa các khai báo cho tất cả các chức năng trong API Windows, tất cả các macro phổ biến được sử dụng bởi các lập trình viên Windows và tất cả các loại dữ liệu được sử dụng bởi các chức năng khác nhau và hệ thống con. Nó xác định một số lượng lớn các hàm cụ thể của Windows có thể được sử dụng trong C. API Win32 có thể được thêm vào dự án lập trình C bằng cách bao gồm tệp tiêu đề và liên kết với các thư viện thích hợp. Để sử dụng các hàm trong xxxx.dll, chương trình phải được liên kết với xxxx.lib (hoặc libxxxx.dll.a trong MinGW). Một số tiêu đề không được liên kết với một dll nhưng với một thư viện tĩnh (ví dụ: Scrnsave.h cần Scrnsave.lib).

* + 1. string.h

Tiêu đề string.h xác định một loại biến, một macro và các hàm khác nhau để thao tác các mảng ký tự

* + 1. time.h

Tiêu đề time.h xác định bốn loại biến, hai hàm macro và các hàm khác nhau để thao tác ngày và giờ.

* 1. Trình biên dịch trong C :

Trình biên dịch để chạy được chương trình này và gây ít lỗi nhất là : TMD-GCC 32-bit Release có sẵn trong Dev C++ .

CHƯƠNG III : MÔ PHỎNG CÁC THUẬT TOÁN TẠO FORM

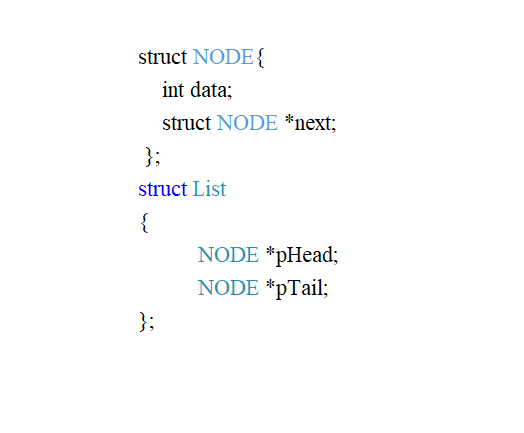
* + - 1. Thuật toán :
  1. Thuật toán tạo một danh sách liên kết :
  2. Khái niệm danh sách liên kết :

Danh sách liên kết là một chuỗi các cấu trúc dữ liệu, được kết nối với nhau thông qua các liên kết. Danh sách liên kết là một chuỗi các liên kết có chứa các mục. Mỗi liên kết chứa một kết nối đến một liên kết khác. Danh sách liên kết là cấu trúc dữ liệu được sử dụng nhiều thứ hai sau mảng.

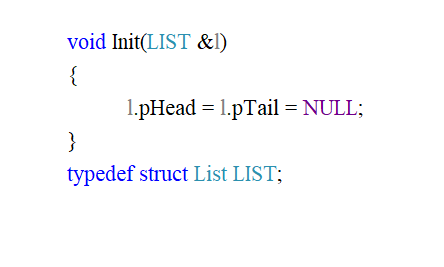
* 1. Khai báo struct :

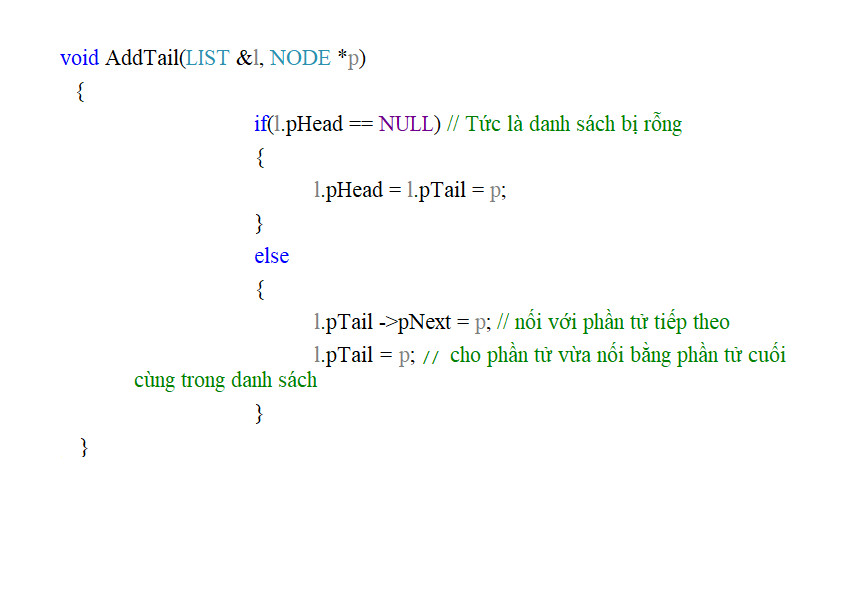
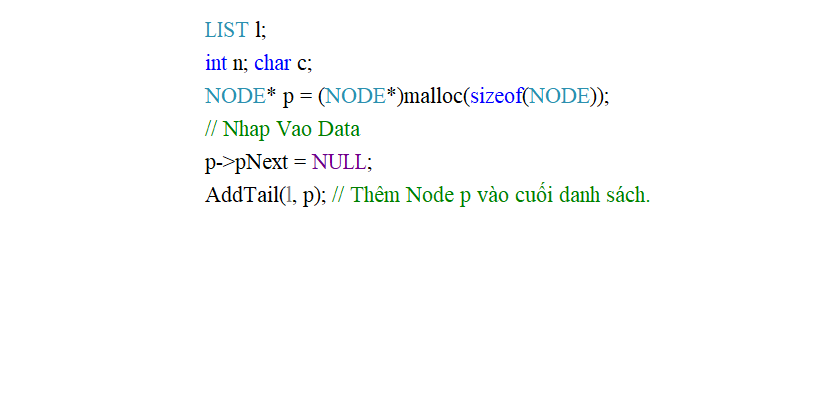
- Danh sách liên kết được định nghĩa trong C là phần dữ liệu và một con trỏ để trỏ đến struct tiếp theo :

- Một struct trong danh sách liên kết trong struct dữ liệu là một số kiểu int được khai báo như sau :



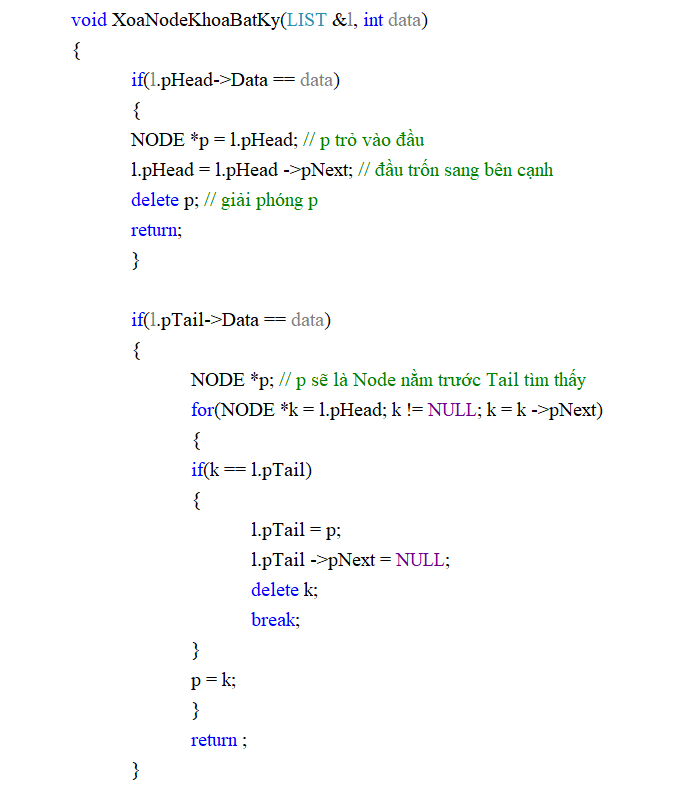
* 1. Khởi tạo danh sách :
* Danh sách muốn hoạt động được đầu tiên phải khởi tạo gán cho phần tử đầu tiên và phần tử cuối cùng = ‘\0’ (NULL) nếu không danh sách này không hoạt động được .
* Code khởi tạo

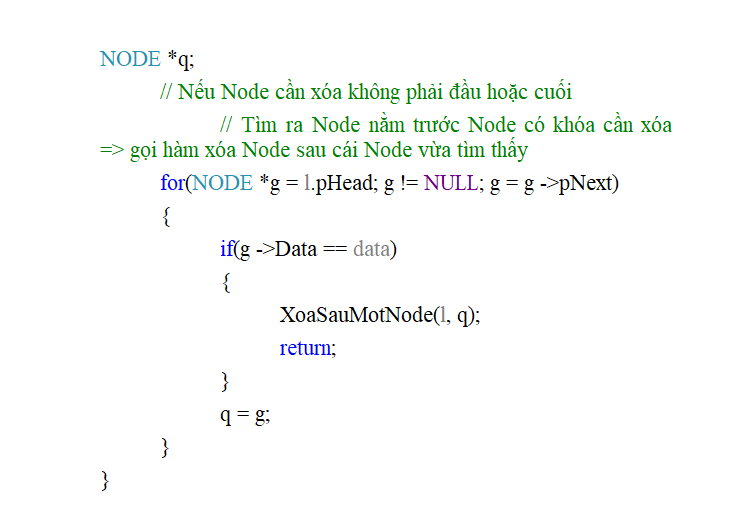


* 1. Thêm một struct vào đằng sau con trỏ next trong danh sách :
* Danh sách liên kết phải thêm các struct đồng dạng vào sau nó để tạo thành một danh sách với các mắc nối liên kết .
* Code thêm vào sau một danh sách liên kết có sẵn :
  1. Dữ liệu trong danh sách liên kết :
* Một danh sách liên kết cần phải có dữ liệu trong nó , dữ liệu dùng để tuy cập và thao tác .
* VD đoạn chương trình thêm một struct vào danh sách liên kết:
  1. Xóa một struct trong danh sách liên kết :

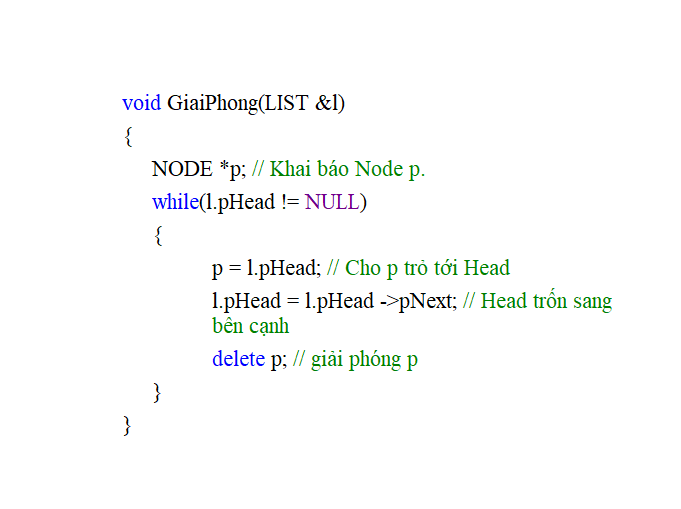
Xóa một struct trong danh sách là một việc hết sức cần thiết khi người dùng không thực sự cần trong danh sách liên kết nữa .

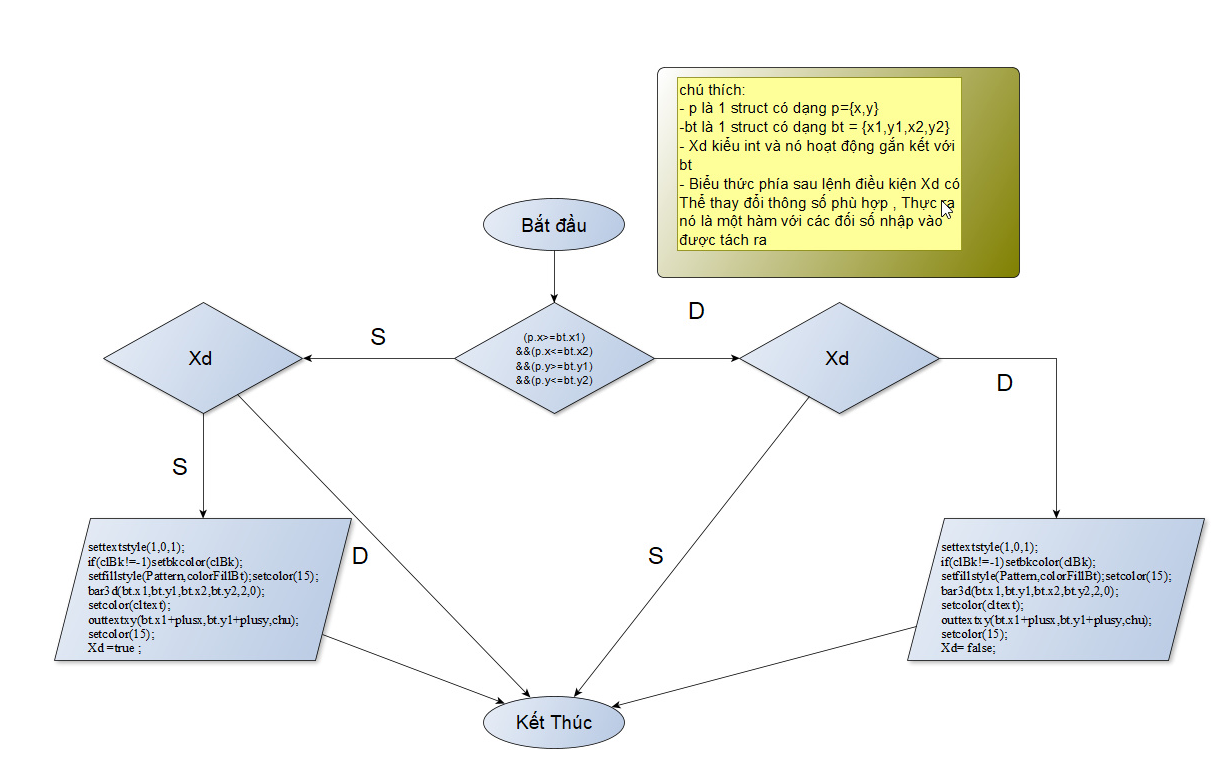
Thuật toán xóa danh sách liên kết :



 Giải phóng vùng nhớ :

1. Ưu điểm của danh sách liên kết so vớ những cách thông thường chính là cấp phát vùng nhớ động , khi không dùng đến nữa ta có thể xóa cả danh sách để tài ng uyên vào việc sử dụng những việc khác .
2. Thuật toán giải phóng vùng nhớ danh sách liên kết :



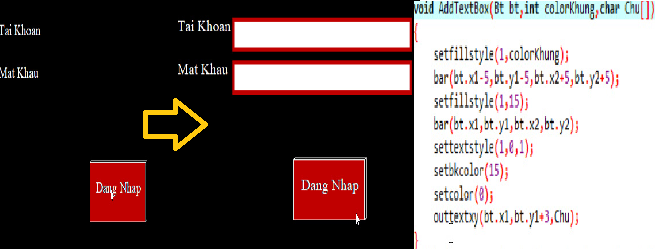
1. Thuật toán nhận biết khi di chuyển chuột vào vị trí trên màn hình :
   1. Hình ảnh :

*Hình 1 : Sơ đồ khối kiểm tra khi di chuyển*

*chuột có chạm vào ô button hay không ??*

* 1. Giải thuật :
* Bước 1 : Khởi tạo biến khác nhau tương ứng với mỗi vị trí hay Button và những dữ liệu cần thiết trong việc tô màu và làm thay đổi màu sắc .
* Bước 2 : Lặp không qua điều kiện cho đến khi thỏa mãn điều kiện chuột có chạm vào ô button ,… và biến xác định = true với có rê chuột vào button hoặc sai nếu không rê chuột vào . nếu thõa mãn điều kiện tiến hành tô màu với các biến cho trước .
* Bước 3 : quay trở lại bước một để làm những tác vụ khác .
  1. Nội dung thuật toán :

Nếu muốn viết một chương trình hoàn chỉnh về mặt đồ họa việc đầu tiên ta cần làm đó chính là tạo những nút button và hiệu ứng , sau đây là một thuật toán kết hợp rê chuột để xác định khi chúng ta rê chuột hay không rê chuột nào nó thì những hành động nào nên lặp lại và không nên lặp lại . Khi đó những hệ sắc thái của màu nó sẽ khác nhau . và nếu không có một biến ngoài đảm đương việc xác định lúc nào lên lặp lại và không nên lặp lại thì nó sẽ rất tốn tài nguyên và thời gian . Vd nếu chúng ta có giá trị của một biến bool = false và không rê chuột vào một nút button thì lệnh điều kiện khi chúng ta chạy chương trình nó sẽ thực hiện một lần đặt giá trị của biến = true, thực tế , nếu như chuột vẫn nằm ngoài vị trí của button thì nó vẫn giữ nguyên màu cũ. Vì vậy , khi có biến ngoài này vòng lặp tiếp theo khi chúng ta không rê chuột vào button này nó sẽ không thực hiện lệnh bên trong vì cho dù thực hiện nó vẫn giống màu cũ , làm như vậy gây tiết kiệm thời gian và tăng hiệu suất của một chương trình đang chạy . màu sẽ thay đổi và chúng ta nên thực hiện lệnh bên trong khi chúng ta rê chuột vào button này giá trị = true sẽ thõa mãn điều kiện rê chuột và sẽ tực hiện điều kiện trong vòng lặp một lần và đặt lại giá trị bằng false . Nói chung chúng ta nên có một biến ngoài nhằm tiết kiệm hiệu năng và khiến chương trình hoạt động một cách trơn tru hơn .

1. Thuật toán tạo nên một textbox :
   1. Hình ảnh :

*Hình 2 : Tạo một textbox trong chương trình*

* 1. Nội dung thuật toán

Ví dụ về sử dụng hàm để vẽ một textbox trên cho ta thấy cách vẽ ra một textbox mặc định có bề dày viền là 5 pixel với màu khung và màu viền tự chọn . Đầu tiên nó sẽ vẽ ra một hình chữ nhật màu tùy chọn do chính người lập trình đặt màu tùy ý , tiếp theo một hình chữ nhật màu trắng được tạo nên , đây cũng chính là màu khung để chúng ta click chuột vào và nhập kí tự từ bàn phím , việc làm cuối cùng của hàm này là in chữ ra trên màn hình với một chuối kí tự được đưa vào hàm . Cả ba thuộc tính trên đều phụ thuộc vào struct Bt vơi kiểu dữ liệu trong đó lần lượt là int x1,y1,x2,y2 .

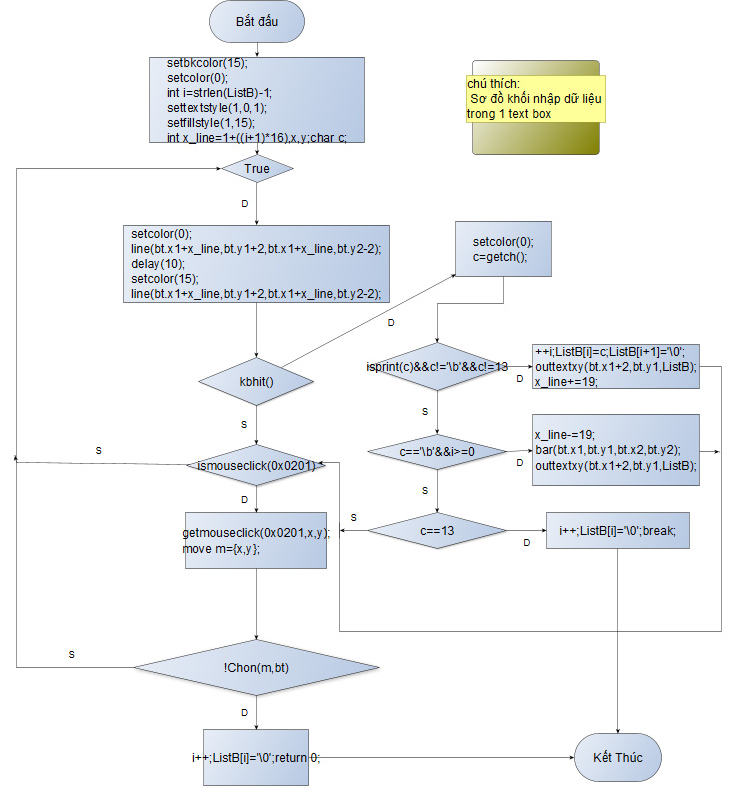
1. Thuật toán tạo nên một List Box :
   1. Hình ảnh :



*Hình 3 : Thêm một List Box trong chương trình*

* 1. Nội dung thuật toán :

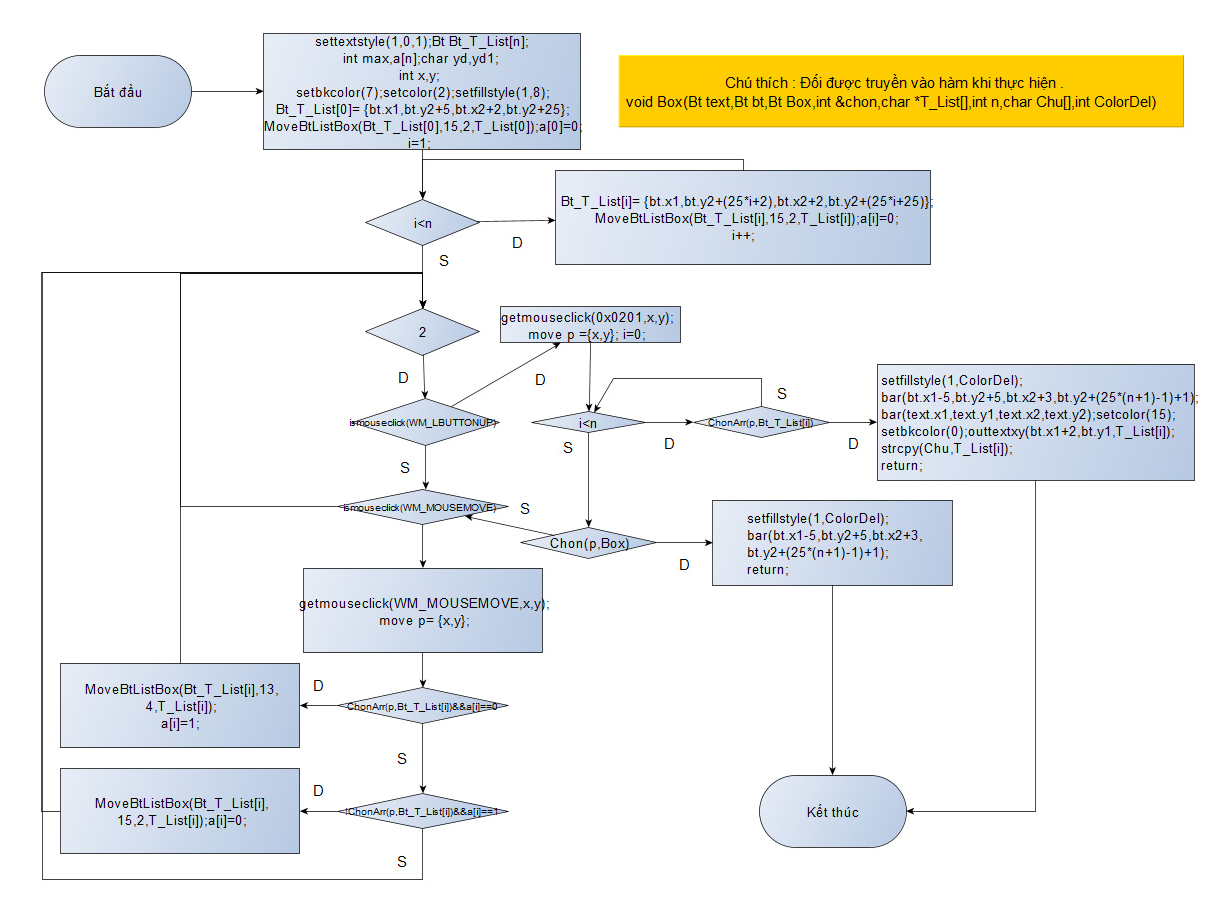
Muốn thêm một List box vào trong chương trình đầu tiên ta sẽ phải cho thông số cảu một hình chữ nhật và sau đó đưa phần text , phần box , tỉ lệ text-box và màu khung vào , nó sẽ tự động tạo cho chúng ta một text và một box riêng với tỉ lệ tự quyết định và màu như ý với phần khung bề dày là 5 pixel . Cả text và box trên đều phụ thuộc vào struct Bt vơi kiểu dữ liệu trong đó lần lượt là int x1,y1,x2,y2 .

1. Thuật toán viết chữ trong đồ họa :
   1. Hình ảnh :

*Hình 2 : Viết chữ trong ô textbox trong màn hình đồ họa*

* 1. Giải thuật :
* Bước 1 : Kiểm tra số kí tự của chuỗi và trừ đi 1 sau đó gán lên một biến i và cho một số màu sắc cơ bản của ô text box .
* Bước 2 : Cho đi vào vòng lặp vô hạn , đầu mỗi vòng lặp sẽ vẽ lên một con trỏ với các thông số ứng với chiều dài nằm trong ô text và đằng sau chữ . Với delay là 10 miligiay . nếu người sử dụng click chuột vào ngoài ô textbox chuyển sang bước 7, nếu có kí tự nhập vào sẽ chuyển sang bước 3 .nếu không click chuột và không ấn phím nào sẽ tiếp tục lặp đi lặp lại bươc này .
* Bước 3 : Kiểm tra xem kí tự đó có phải kí tự in ra được màn hình không ( Kí tự khác phím điều hướng , esc , … ) Nếu thõa mãn chuyển sang bước 4 . Nếu không thõa mãn điều kiện này chuyển sang bước 5 .
* Bước 4 : Nếu kí tự này khác ‘\b’ và Enter thì sẽ tăng i lên 1 đơn vị cho kí tự lấy này thêm vào mảng với vị trí ListB[i] , tăng vị trí con trỏ lên trí tương ứng với chữ sau đó thêm kí tự ‘\0’ ( NULL ) vào sau chuỗi này để kết thúc chuỗi và in chuỗi này ra vị trí gần đầu của text box.
* Bước 5 : Thõa mãn điều kiện i>0 ( Chuỗi không rỗng) và ấn vào đúng phím Backspace thì sẽ thực hiện trừ x\_Line tức vị trí con trỏ đi tương ứng với một chữ cái sau đó cho kí tự tại i = ‘/0’ sau đó trừ đi i một đơn vị , nếu không thỏa mãn điều kiện này chuyển sang bước 6 .
* Bước 6 : Kiểm tra phím nhập vào có phải Enter ? nếu đúng thì kết thúc đoạn chương trình này , nếu sai thì quay lại bước 2 .
* Bước 7 : Thoát đoạn chương trình .
  1. Nội dung thuật toán :

Như chúng ta đã biết . trong ngôn ngữ C không có một hàm hay một key work nào có thể tự động cho chúng ta viết chữ trên màn hình đồ họa được trừ một hàm getch() . Tôi đã ứng dụng hàm này làm một công cụ nhập xuất trong màn hình đồ họa riêng cho mình. Vòng lặp này đầu tiên sẽ cho ta một vòng lặp vô hạn . Vì sao ? đầu tiên , tôi muốn nó giống với những textbox trong ngôn ngữ bậc cao , có nghĩa là khi chúng ta không nhập bất kì một kí tự nào vào thì con trỏ vẫn nháy bính thường , thứ hai là chúng ta không chỉ kết thúc nhập bằng phím mà cần có kết thúc nhập bằng chuột , khi chúng ta click chuột trái ra ngoài button text thì sẽ kết thúc không cho chúng ta nhập nữa, cả hai việc đó chỉ có thể thực hiện được khi chúng ta đưa chúng vào trong vòng lặp vô hạn . Trở lại với vấn đề chính , đầu tiên phải nói đến việc ở đây tôi đặt i= -1, tại sao vậy ?? bởi vì khi chúng ta không nhập bất kì một phím gì mà nhấn enter thì kí tự đầu tiên sẽ là null mà không nhày loạn xạ . khi chúng ta có ấn một phím thì nó sẽ thực hiện những ba công việc sau trong vòng lặp if else : nếu là chữ có thể in ra màn hình được thì sẽ cho vào trong một mảng kí tự và tăng lần , nếu là một Back Space thì sẽ cho kí tự tại i= NULL và đưa con trỏ trở lùi một vị trí . khi chúng ta thực hiện hai lệnh trên đều cho con nháy chuột dịch chuyển sang trái hay sang bao nhiêu đơn vị . Khi chúng ta nhấn phím Enter thì sẽ kết thúc ở phía sau chuỗi bằng một kí tự NULL và kết thúc đoạn chương trình .

1. Thuật toán sử dụng list box :
   1. Hình ảnh :

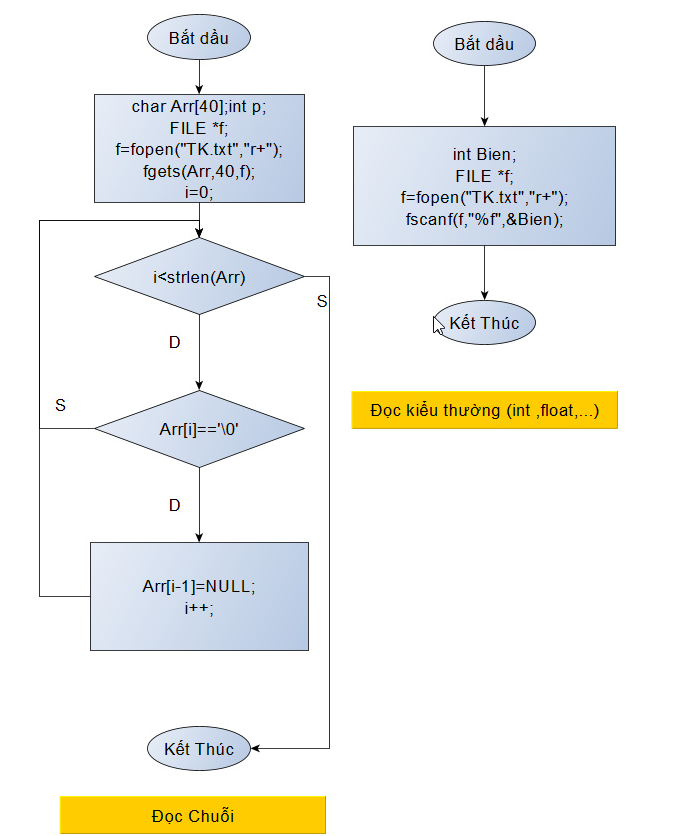
*Hình 4 : hành động xảy ra khi ấn vào box trong ListBox*

* 1. Giải thuật :
* Bước 1 : Cho i=1, n mảng chuỗi kí tự ,Cho cỡ chữ , tạo n struct Bt tên Bt\_T\_List[n] và mảng a kiểu int .tạo ra một khung hình chứ nhật với màu sắc và chữ tương ứng thông qua hàm MoveBtListBox đã định nghĩa ở trên .
* Bước 2 : cho for lặp tới n với điều kiện i<n , trong for là tạo những thông số cho Bt\_T\_List[i] với y1,y2, tăng lên dần sau đó Tạo hình chữ nhật cho các struct I trong bước 2 và tạo màu sắc chữ trong đó bằng hàm MoveBtListBox.
* Bước 3 : Cho vào vòng lặp vô hạn và liên tục làm mới list box và hiệu ứng cho con trỏ chuột nếu có click chuột trái thì chuyển sang bước 4 .
* Bước 4 : nó sẽ chạy vào trong vòng for với số phần tử bằng số hàng trong listbox và chạy lên thuật toán chọn button . nếu như có chọn đúng vị trí của một chuỗi trong danh sách này thì tiến hành copy chuỗi tại vị trí i này cho chuỗi cần lấy và tiến hành in ra màn hình tại vị trí của phần text trong listbox này sau đó xóa phần hình tạo ra để chứa danh sách trên với màu nền là màu nền cũ và tiến hành kết thúc đoạn chương trình. Nếu như không phải tiếp tục kiểm tra xem có chọn vào trong button box không , nếu có thì xóa phần hình tạo ra để chứa danh sách trên với màu nền là màu nền cũ kết thúc đoạn chương trình này, nếu không thì quay trở lại bước 3 .

* 1. Nội dung thuật toán :

Việc đầu tiên trong đoạn chương trình Listbox này cần làm đó là nhận vào một con trỏ mảng n chuỗi kí tự và tiếp theo nó sẽ tạo những ô hình chữ nhật tương ứng với số màu ta có thể đặt ở đối trong hàm sau đó những ô đó sẽ tương ứng với một mảng kí tự tương ứng được thêm vào trước khi gọi hàm này. Khi hàm này chạy nó sẽ in ra một danh sách gồm nhiều hình chữ nhật xếp từ trên xuống dưới , khi ta di chuyển vào những ô hình chữ nhật nó sẽ chuyển tới hàm di chuyển con trỏ chuột vào button đã nói ở trên cho ta thấy tính tương tác với người dùng cao. Hàm này sẽ kết thúc nếu như chúng ta thực hiện một trong hai lệnh tương tác sau đây : Nếu chúng ta chọn một trong danh sách list thì sẽ sao chép nội dung mảng đó vào mảng kí tự ta đưa vào hàm hay nếu chúng ta chọn lại box thì nó sẽ kết thúc hàm với mảng kí tự bây giờ là mảng kí tự giống với mảng kí tự trước khi đưa vào hàm này .

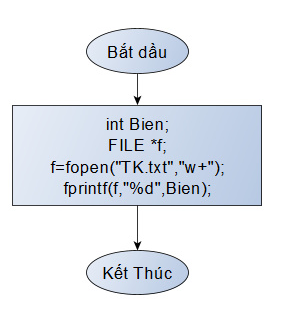
1. Thuật toán đọc dữ liệu từ file .txt :
   1. Hình ảnh :



*Hình 5 : Đọc dữ liệu từ file*

* 1. Giải thuật :
* Đối với mảng kí tự :
* Bước 1 : Cho một con trỏ file f bắt đầu mở file mà ta chỉ định
* Bước 2 : Bắt đầu đọc chuỗi với số kí tự tối đa.
* Bước 3 : Lặp for cho đến khi gặp kí tự ‘\n’ và thay kí tự ‘\n’ tại vị trí đó trong chuỗi bằng kí tự ‘\0’ .
* Đối với số và kí tự :
* Bước 1 : Cho một con trỏ file f bắt đầu mở file mà ta chỉ định
* Bước 2 : đọc số , kí tự bằng hàm fscanf .
  1. Nội dung thuật toán :

Để đọc dữ liệu từ file ra . cụ thể là file \*.txt thường sử dụng hai hàm đọc vào . Nếu như muốn đọc kiểu dữ liệu thường như int , float , … thì ta dùng hàm fscanf đọc dữ liệu từ file theo hàm cơ bản vào , còn cách thứ hai là ta đọc chuỗi bằng hàm fgets( chuỗi ta lấy vào, số kí tự lấy, con trỏ trỏ đến file lấy) .hoạt động của hàm này là đọc kí tự và nó sẽ dừng lại khi có hai trường hợp , trường hợp thứ nhất chính là ta sẽ điền số kí tự cần lấy phía trước chuỗi trong file sau đó đọc số đó vào trước và lấy số đó làm đối số thứ hai trong hàm fgets . có ngĩa là nó sẽ đọc tới kí tự thứ n đó sẽ dừng lại và kết thúc chuối , trường hợp thứ hai là hàm sẽ đọc đến khi gặp kí tự ‘\n’ sẽ tự động ngắt chuỗi . Ta thấy cách đầu sẽ tốn thêm dữ liệu ghi vào file , nếu gặp dữ liệu lớn thì đó ta sẽ tốn thêm bộ nhớ và không tối ưu. ở đây , tôi chọn cách thứ hai , đọc chuỗi vào , tối đa là 50 kí tự sau đó sẽ cho for lặp đến vị trí i= ‘\0’ và cho kí tự trước nó (i-1) = ‘\0’ vì nếu lấy chuỗi theo cách này khi đọc chuỗi đến kí tự ‘\n’ sẽ tự động ngắt chuỗi nhưng vẫn lấy kí tự ‘\n’ vì vậy ta phải lọc nó ra theo cách nêu trên .

1. Thuật toán ghi file :
   1. Hình ảnh :

*Hình 6 : Ghi dữ liệu vào file*

* 1. Nội dung thuật toán :

Ở đây việc ghi dữ liệu vào file có thể dử dụng một hàm duy nhất chính là hàm fprintf , hàm này có thể ghi chuỗi , số , v.v. vào trong file tuy nhiên với cách đọc file ở trên ta phải lưu ý : việc đọc 2 hay nhiều số liền kề ta có thể sử dụng dấu cách nhưng với việc ghi một chuỗi thì phải có kí tự ‘\n’ phía sau vì nó phù hợp với cách đọc file ở bên trên , nếu không sẽ xảy ra lỗi trong trường hợp này .

1. Thuật toán xóa mảng chuỗi và những mảng cấp phát động :
   1. Code xóa :

void FreeMalloc(char\* arr[], int n, int start)

{

for (start; start < n; start++)

{

free(arr[start]);

arr[start] = NULL;

}

}

void FreeMalloc2(int n, int n2, int start, int start2, char\* arr[][2])

{

for (start; start < n; start++)

{

for (int j = start2; j < n2; j++)

{

free(arr[start][j]);

arr[start][j] = NULL;

}

}

}

* 1. Giải thuật :
* Xóa mảng động một chiều.
* Bước 1 : truyền mảng một chiều vào hàm .
* Bước 2 : Cho for chạy từ phần tử đầu tiên đến phần tử cuối cùng
* Bước 3 : Tiến hành xóa mảng thông qua hàm free , sau khi giải phóng vùng nhớ này , cho con trỏ trỏ đến NULL.
* Xóa mảng động hai chiều :
* Bước 1 : Cho các thông số cần thiết và phải khai báo mảng động với phần cuối cùng trong dấu [ phải có tham số .
* Bước 2 : Cho vòng lặp for lồng nhau và tiến hành xóa như bước 3 của mảng một chiều .

1. Thuật toán lấy giờ thêm vào danh sách liên kết :

* Với một người chủ quán và nhân viên , quản lý giờ vào và ra trong một quán café là một việc cần thiết vì nó giúp chủ quán thống kê một cách dễ dàng hơn
* Code :

void gio(LIST& l, QL\* p, int val) {

int bien1;

char ap[] = "%H%M%S";

time\_t now;

struct tm ts;

char buf[10];

time(&now);

ts = \*localtime(&now);

strftime(buf, sizeof(buf), ap, &ts);

bien1 = atoi(buf);

if (val == 0) {

p->gioi = bien1 / 10000;

p->phuti = (bien1 % 10000) / 100;

p->giayi = ((bien1 % 1000) % 100);

}

else {

if (val == 1) {

p->gioo = bien1 / 10000;

p->phuto = (bien1 % 10000) / 100;

p->giayo = ((bien1 % 1000) % 100);

}

else

p->gc = bien1 / 10000;

p->pc = (bien1 % 10000) / 100;

}

}

1. Giải thuật :

Bước 1 : đầu tiên chúng ta sẽ cho một mảng kí tự ap bên trong đó chứa %H (Giờ) %M(phút )%S(Giây) sau đó lấy thời gian hiện tại trong struct tm đã được định nghĩa sẵn.

Bước 2 : chuyển từ chuỗi đó sang số nguyên

Bước 3 : tách giờ , phút , giây ra khỏi số đó bằng các thuật toán tách số nguyên .

1. Biểu mẫu ứng dụng thuật toán:
2. Đoạn chương trình đọc file :
3. Code :

void adddataNV(LK& lk) {

FILE\* f;

f = fopen("DSNV.txt", "r+");

char c, manv[10], ten[50], cmnd[13], qq[70]; int t;

while (!feof(f)) {

fgets(manv, 10, f);

if (strcmp(manv, "") == 0)return;

for (int i = 0; i <= 10; i++) {

if (manv[i] == NULL) {

manv[i - 1] = NULL;

break;

}

}

fgets(ten, 50, f);

for (int i = 0; i <= 50; i++) {

if (ten[i] == NULL) {

ten[i - 1] = NULL; break;

}

}

fscanf(f, "%d", &t);

fgets(qq, 70, f);

for (int i = 0; i <= 70; i++) {

if (qq[i] == NULL) {

qq[i - 1] = NULL; break;

}

}

fgets(cmnd, 13, f);

for (int i = 0; i <= 13; i++) {

if (cmnd[i] == '\0') {

cmnd[i - 1] = NULL; break;

}

}

NV\* p = (NV\*)malloc(sizeof(NV));

strcpy(p->Manv, manv);

strcpy(p->Tennv, ten);

p->age = t;

strcpy(p->Que, qq);

strcpy(p->CMND, cmnd);

p->next = NULL;

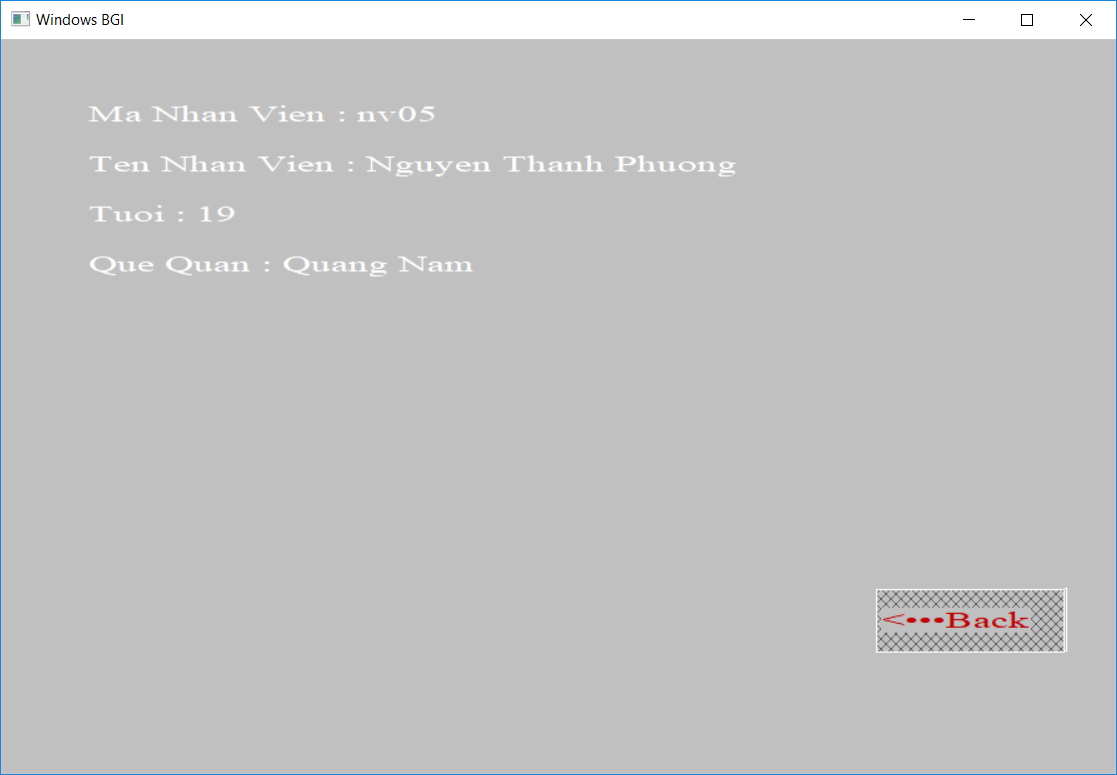
AddTTailNV(lk, p);

}

fclose(f);

}

1. Nội dung code :

Để thực hiện việc thêm này đầu tiên ta thực hiện cho vào vòng lặp lặp đến khi file rỗng . đầu file sẽ là một thuật toán kiểm tra xem kí tự đọc vào phải là kí tự rỗng , nếu đúng là nó thì không cần đọc tiếp nữa file thực tế đã rỗng rồi, không có bước này thì nó sẽ tự thêm một hàng dữ liệu với dữ liệu đầu tiên bị rỗng , điều này gây nên lỗi chương trình và lần in hay đọc file tiếp theo có khả năng xảy ra lỗi, tiếp theo tiếp tục đọc các dữ liệu dạng chuỗi ( Mã NV,Tên NV,Quê Quán , chứng minh nhân dân) dùng hàm fgets và loại bỏ kí tự rỗng trong chuỗi . đến đoạn đọc tuổi ta sử dụng hàm fscanf để đọc số . kết quả xuất ra màn hình ta được :

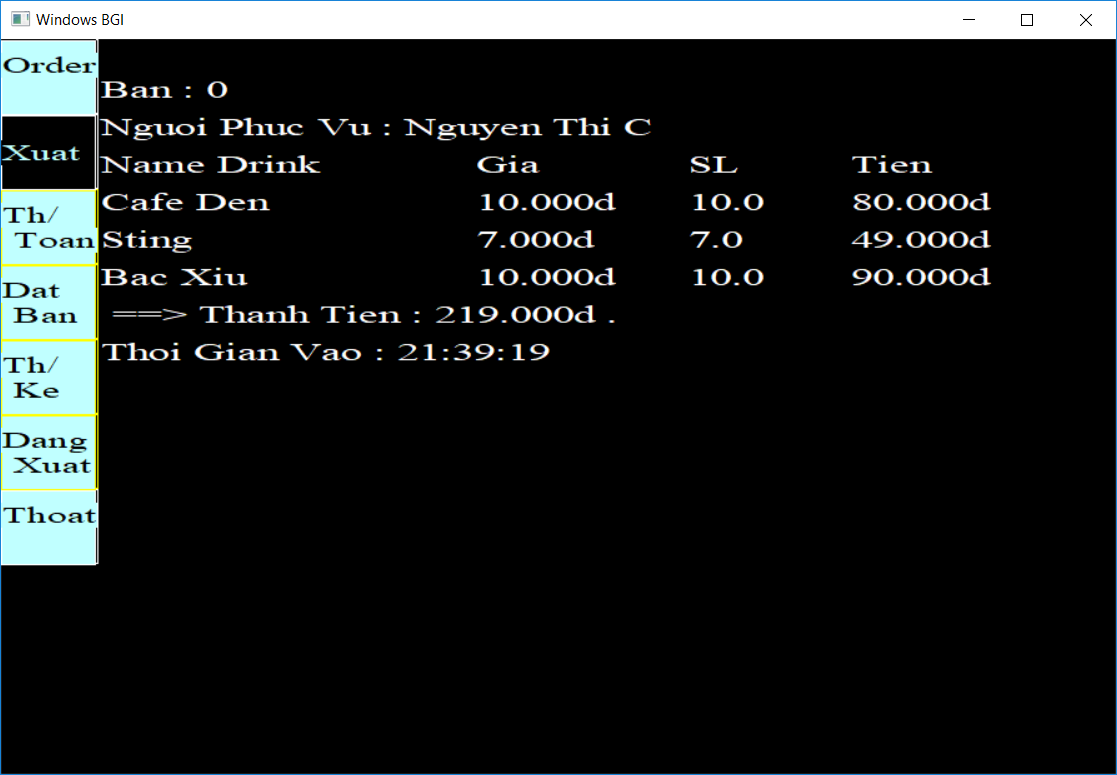
1. Tạo form order thông qua các thuật toán trên :

*Hình 7 : Form order trong chương trình quản lý quán café*

Một chương trình quản lý quán café việc làm quan trọng nhất là phải order được . chương trình của tôi cũng vậy, form trên gồm có những nội dung chính như : những nút button là những cái bàn hay nút ok, exit . Textbox như những ô màu đen bên cạnh Listbox hay những ô màu trắng hình chữ nhật có viền ngoài 5 pixel .

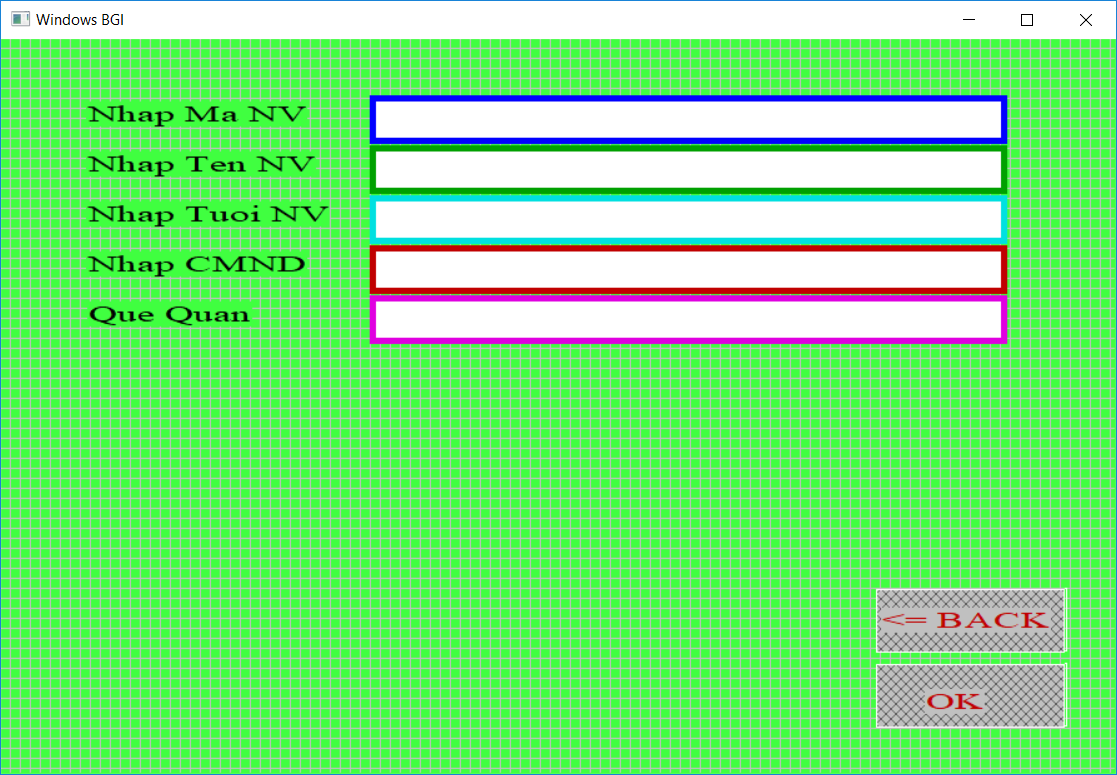
Liên kết trong form : đầu tiên ta sẽ tạo ra những chuối phù hợp để chứa những kí tự hay mảng kí tự lấy vào , khởi tạo ban đầu luôn gán cho mảng không có kí tự nào “” . Trong form liên kết giữa danh sách liên kết thức uống và danh sách liên kết nhân viên và danh sách order được thể hiện như sau : Đầu tiên ta sẽ cho dữ liệu cần vào một mảng chuỗi một chiều hay hai chiều tùy ý và sau đó xuất ra màn hình dưới dạng List hay in trực tiếp ra màn hình . sau khi ta chọn những dữ liệu trong list hoăc nhập dữ liệu vào từ Textbox . sau khi nhập xong và nhấn OK nó sẽ bắt đầu kiểm tra những dữ liệu cơ bản cần phải có trong Form Order , một bảng thông báo rằng chúng ta chưa nhập đầy đủ dữ liệu sẽ hiện ra khi chúng ta nhập thiếu những trường cơ bản chương trình sẽ kiểm tra những dữ liệu đó có được thay đổi hay không bằng cách so sánh với giá trị rỗng ( “” ) tức là giá trị khởi tạo ban đầu , chúng ta bắt buộc phải ấn OK trong thông báo và trở về làm việc với form . còn nếu chúng ta nhập đầy đủ những dữ liệu bắt buộc của form thì sẽ có một bảng thông báo xác nhận hiện lên thông báo chũng ta có chắc chắn thực hiện không , nếu có thì những dữ liệu chúng ta nhập nãy giờ nó sẽ bắt đầu thêm vào danh sách liên kết QL và thêm ngày giờ vào của một bản kể từ lúc chúng ta xác nhận bước này , Nếu chọn không thì chúng ta tiếp tục quay lại hoạt động làm việc với form này . Quay lại hoặc thoát thì chỉ cần ấn một lần vào Button Exit có sẵn trên màn hình .

1. Xuất order ứng dụng danh sách liên kết :



Hình 8 : Một form trong xuất hóa đơn

Trong form này tập trung vào xuất một hóa đơn lên trên màn hình , trong form có những thứ cơ bản giúp nhân viên hay chủ quán có thể xem xét nhanh lại tình hình order tại một bàn nào đó ,…

1. Một form thêm thông tin của một nhân viên :

*Hình 9 : Một form thêm nhân viên*

Form này chứa đầy đủ dữ liệu cơ bản cần có của một nhân viên trong một quán café hạng trung bao gồm các trường như : Mã Nhân Viên , Tên Nhân Viên , Tuổi Nhân Viên , SDT liên hệ , CMND , quê quán . Form này chủ yếu là những ô textbox và những dữ liệu trong form này tuân theo một số ràng buộc nhất định . Đầu tiên mã nhân viên nó sẽ tự động nhảy autonumber từ NV01… . Tên nhân viên phải là một kiểu chuỗi , tuổi nhân viên nhập từ kiểu chuỗi và chuyển sang kí tự nếu độ tuổi thích hợp từ 1- 100 . Chứng minh nhân dân sẽ thuộc dạng 13 số nếu không phải sẽ yêu cầu thông báo ra màn hình . Quê quán sẽ là một kiểu chuỗi , ta có thể kiểm tra người dùng nhập đúng quê quán không bằng cách nhập vào một file rồi kiểm tra chuỗi hợp lệ trong đó so với chuỗi nhập vào nhưng đây là việc làm cần dữ liệu lớn , đối với dữ liệu chương trình nhỏ thì chương trình này ta không cần quá sức lưu ý tơi . Cuối cùng sẽ có hai nút button ta cần lưu ý : Chọn Back nếu bạn không muốn tiếp tục nhập và toàn bộ dữ liệu được lưu hiện thời trong form sẽ bị xóa , chọn OK nếu bạn cảm thấy dữ liệu mình nhập vào là đầy đủ và nhờ máy tính kiểm tra những trường nào nhập vào sai , thiếu dữ liệu hoặc không phù hợp ràng buộc dữ liệu.

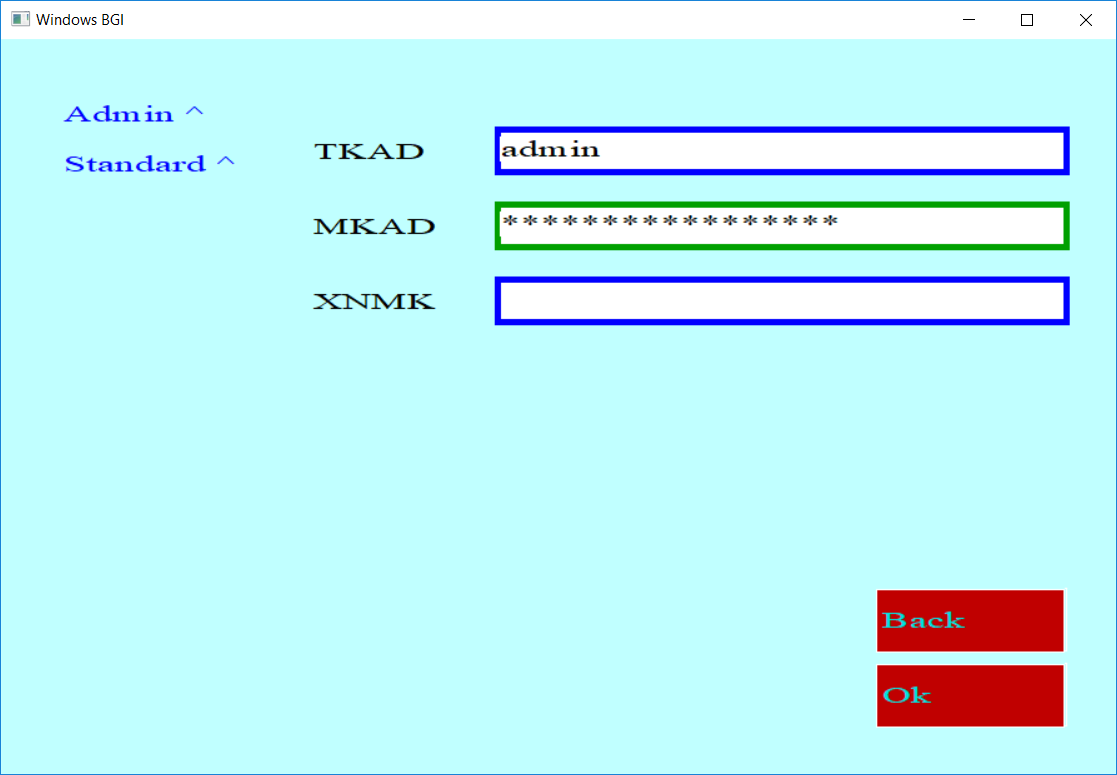
1. Ứng dụng các thuật toán làm mới , buton, … để tạo nên một form xóa nhân viên :



*Hình 10 : Một form xóa nhân viên.*

Về cơ bản những giao diện về thức uống và nhân viên trong việc thêm , xóa sửa đêu có một phần giống nhau . Về Xóa thức uống đầu tiên sẽ hiện ra một bảng gồm các Button để ta chọn thức uống còn xóa , Sau đó một bảng thông báo sẽ hiện lên để hỏi chúng ta có tiếp tục thao tác về những vấn đề này nữa không , nếu chúng ta ấn xác nhận có thì danh sách thức uống sẽ bị xóa vĩnh viễn kèm theo những thông tin theo như mã thức uống , tên thức uống , giá , tình trạng nhập kho , xuất kho , tất cả sẽ không còn được thống kê hay báo cáo cũng như danh sách thức uống này hoàn toàn bị loại ra khỏi danh sách order nhưng những giữ liệu về những lần order về dữ liệu xóa này vẫn được giữ nguyên trong bảng xem thông tin . Xác nhận xóa trong Form này sẽ dẫn đến xóa một “ nút “ trong danh sách liên kết . để thoát form này ta có thể chọn Back khi chưa chọn thức uống để xóa này hay ấn NO nếu như đã chọn thức uống để xóa mà lại không muốn xóa nó .

1. Ứng dụng các thuật toán trên chế tạo form đổi mật khẩu :



*Hình 11 : Đổi tên tài khoản hoặc mật khẩu .*

Vẫn đề bảo mật của một chương trình ứng dụng là một điều được quan tâm rất nhiều . Trong Form này ta chủ yếu trình bày về mật khẩu và xác nhận giữ lại hay thay đổi mật khẩu , Trong Form gồm hai tài khoản Admin và standard tương đương với tài khoản khách và chủ và chúng ta chỉ có thể thực hiện bước này sau khi đã đăng nhập thành công vào tài khoản admin . Đầu tiên ta phải chọn một trong hai text button admin hay standard sẽ hiện ra một bảng gồm các text box trong nếu ta thay đổi giá trị trong này và xác nhận sau khi ấn Button OK . lần đăng nhập tiếp theo của tài khoản bạn thay đổi sẽ là tài khoản bạn vừa đổi . Để bảo mật khi có người muốn nhìn mật khẩu , ngoại trừ tên tài khoản ra mật khẩu cứ ước định nhập xong trong vòng 200 mi-li-giây sẽ tự động được chuyển sang dấu \* . Đây là một thuật toán dựa trên hàm getch() lập nên thay vì in ra kí tự trong mỗi lần thì đầu tiên sẽ in ra kí tự và sau một khối thời gian mi-li-giây thích hợp sẽ tự động điền thêm dấu \* vào chỗ vừa in ra kí tự .

1. Ứng dụng thuật toán làm nên form nhập kho trong chế độ đồ họa :

Đây là một form cho ta thấy chủ yếu những tác vụ sau . đầu tiên sẽ cho ta một List box để chọn thức uống mà ta có thể nhập kho vào , bạn cũng có thể tao tác bước này bằng cách chèn dữ liệu vào các ô text box nhưng nếu nhập sai dữ liệu thì bạn không thể đi đến các bước tiếp theo , nếu bước này không có vấn đề gì thì ta có thể nhập kho thức uống bằng một trong hai cách dưới đây . cách đầu tiên là chúng ta nhập với số lượng cơ bản đã được thêm vào ở bảng thêm thức uống ( Chai , Kg ,…) . Nếu có một mã số lượng nào khác với cách quy đổi khó tình thì ấn vào text button advanced sẽ giúp chúng ta dễ dàng thực hiện hơn những bước này . Trong Text Button này có hai text box cần chúng ta nhập dữ liêụ vào , Text box Tỉ lệ chuyển đổi là tỉ lệ giữa đơn vị cơ bản với đơn vị nâng cao này . VD : ta có thức uống sting với đơn tị tính cơ bản là chai tỉ lệ 24-1 tương ứng với 24 chai sting sẽ tương ứng với một đơn vị này . sau khi nhập bước này và không bị thông báo lỗi thì sẽ tiến hành nhập số lượng nâng cao trong ô text box trên , ta có thể theo dõi SLHC để xem thử thức uông này có tăng hay giảm không nếu tăng hoặc giảm thì số lượng nhập hợp lệ . Sau khi cảm thầy nhập hoàn tất dữ liệu rồi ta chọn button OK sẽ hiện ra một bảng thông báo với số lượng nhập được đổi ra số lượng cơ bản hoàn toàn xác nhận bước này để thành công nhập kho và lưu lại vào tệp , nếu rê chuột và ấn Button NO xác nhận sẽ quay trở lại giao diện làm việc trước và bạn có thể hoàn thành một số việc mà bạn muốn .

1. Kết luận và kiến nghị :
2. Kết quả đạt được :

Qua đề tài “ Tìm hiểu lập trình đồ họa và xử lí file trong C – Xây dựng ứng dụng minh họa quản lý quán café ”. Tôi đã rút ra được những kết luận :

* Qua chương trình minh họa giúp cho những người chưa biết về những nội dung cơ bản của một form tự làm và quy cách hoạt động của nó .
* Thành thạo hơn về quy trình in và xử lí tập tin trong C .
* Xây dựng được một ví dụ ứng dụng cơ bản bằng ngôn ngữ C.
* Hiểu hơn về quy trình sử dụng chuột và đưa chuột vào trong đồ họa.

1. Hướng phát triển :

* Dự dịnh trong thời gian tiếp theo sẽ tiến hành nghiên cứu sâu vào C để hoàn thiện thêm chương trình .
* Phát triển chương trình qua những ngôn ngữ bậc cao phát làm phong phú hơn .
* Xây dựng thêm nhiều chương trình ứng dụng nữa .

TÀI LIỆU THAM KHẢO

[1]. Phạm Văn Ất – ThS. Nguyễn Hiếu Cường – ThS. Đỗ Văn Tuấn ,Giáo trình kĩ thuật lập trình C.

# [2]. Chương trình quản lý sinh viên sử dụng struct trong C - <https://nguyenvanhieu.vn/chuong-trinh-quan-ly-sinh-vien-su-dung-struct/>