LỜI NÓI ĐẦU

Ngày nay, ***công nghệ thông tin*** đang phát triển mạnh mẽ và nó đang trở thành một ngành mũi nhọn. Nó đã được ứng dụng rộng rãi trong tất cả các lĩnh vực của đời sống xã hội. Có thể nói sự phát triển của ***công nghệ thông tin*** đã giúp con người giải quyết các bài toán khó trong thời gian ngắn, mà trước đây đòi hỏi con người phải mất nhiều thời gian và công sức với độ chính xác và độ tin cậy cao. Điều này đánh dấu một bước ngoặt vĩ đại trong ngành tin học nói riêng và trong các lĩnh vực đời sống xã hội nói chung.

Sinh viên CNTT ngày nay phải không ngừng học hỏi, cập nhật những cái mới và biết vận dụng kiến thức đã được học hỏi vào thực tiễn của cuộc sống.

Đợt thực tập cơ sở này chính là bước đầu tiên đi sâu tìm hiểu và cũng là cơ hội để sinh viên tổng hợp lại những gì đã được học trong những năm qua.

Đề tài thực tập cơ sở do nhóm 03 – Lớp 48KTin trình bày sẽ cho thấy được phần nào hữu ích của ***công nghệ thông tin*** trong cuộc sống. Đề tài nghiên cứu bao gồm: *Sử dụng thư điện tử(Email), Bài toán dùng thuật toán sinh* hoặc *thuật toán quay lui, Tìm hiểu và khai thác dịch vụ Windows Firewall của Windows, Tìm hiểu và cài đặt thuật toán nén và giải nén dữ liệu Run Length Code(RLE) cho một tệp* đã ít nhiều nói lên sự phát triển của ***Công nghệ thông tin*** ảnh hưởng tới mọi mặt của cuộc sống như thế nào?

Chúng em xin được gửi lời cảm ơn chân thành của mình đến cô ***Thsy.Hồ Thị Huyền Thương*** cùng các thầy cô trong tổ ***Phương Pháp Giảng Dạy*** đã giúp đỡ chúng em trong quá trình thực hiền đề tài này.

Do nhiều yếu tố khách quan cũng như tầm hiểu biết của chúng em còn hạn chế, đề tài thực tập cơ sở của nhóm sẽ còn có nhiều sai sót. Chúng em mong được sự góp ý chân thành của thầy cô và các bạn để kịp thời sửa chữa để thực hiện được tốt hơn trong những lần sau.

Chúng em xin chân thành cảm ơn!

Nhóm SV thực hiện

Nội dung đề tài : 021

<u>Câu 1:</u> Sử dụng thư điện tử (Email)

1. Các thành phần cấu trúc của một địa chỉ thư điện tử
2. Đăng ký hộp thư, thực hiện các chức năng gửi và nhận thư.
3. Vấn đề an toàn khi dùng thư điện tử.

<u>Câu 2:</u> Bài toán dùng thuật toán sinh hoặc thuật toán quay lui

1. Sinh hoán vị
2. Sinh xâu nhị phân
3. Sinh tổ hợp

<u>Yêu cầu:</u>
- Nêu bài toán
- Nêu thuật toán
- Ví dụ minh họa
- Cài đặt thuật toán trên một ngôn ngữ nào đó (Pascal, C, C++ ..)

<u>Câu 3:</u> Tìm hiểu và khai thác dịch vụ Windows Firewall của Windows.

<u>Câu 4:</u> Nén dữ liệu:

Tìm hiểu và cài đặt thuật toán nén và giải nén dữ liệu RLE(Run Length Code) cho một tệp dữ liệu.

<u>Yêu cầu:</u>

1. Ngôn ngữ cài đặt : C hoặc C++
2. Báo cáo:
   - Đề bài toán.
   - Mô tả thuật toán.
   - Mô tả các modul thiết kế bài toán.
3. Chương trình.

PHẦN I: SỬ DỤNG THƯ ĐIỆN TỬ (EMAIL)

Việc trao đổi thông tin trong thời đại công nghệ thông tin đòi hỏi phải nhanh gọn, đơn giản, chính xác..vì vậy mà hòm thư điện tử đã ra đời thay thế một phần cho việc trao trổi thông tin truyền thống mà bạn từng biết đến.

Lá thư được gửi trên hệ thống bưu chính là vật liệu không cần máy nhận hay máy gửi. Trong khi đó, nếu gửi thư điện tử, chỉ có các tín hiệu điện mã hoá nội dung bức thư điện tử được truyền đi đến máy nhận. Do đó, chỉ có nội dung hay cách trình bày lá thư điện tử là được bảo toàn. Trong khi đó, dùng đường bưu điện người ta có thể gửi đi các vật liệu hàm chứa thêm nội dung hay ý nghĩa khác. Điều này có thể rất quan trọng đối với nhiều người.

Dùng thư điện tử thì bất kỳ lúc nào cũng có thể mở phần mềm thư điện tử ra đọc nên tiện lợi hơn là việc phải bỏ thư ở các thùng thư. Đồng thời, vì mỗi người dùng thư đều phải nhập mật khẩu vào máy nên thư điện tử sẽ khó bị người ở chung đọc lén so với thư gửi bưu điện. Nhưng ngược lại, các tay tin tặc xa lạ có thể xâm nhập vào hệ thống thư điện tử của cá nhân nếu như các mật mã hay các hệ thống an toàn phần mềm bị bẻ gãy.

Khối lượng gửi và nhận thư điện tử có thể nhiều hơn thư bưu điện rất nhiều lần. Đối với các dịch vụ thư điện tử mới thì dung lượng có thể lên đến hàng Gbyte như dịch vụ của Gmail chẳng hạn, hay nhiều hơn. Số thư có thể dự trữ trong dung lượng này tương đương với vài bộ tự điển bách khoa.

Các trường hợp thư phá hoại trên hệ thống bưu điện (như là thư có bột antrax, thư bom, ...) rất hiếm có nhưng có thể gây thương vong. Ngược lại, hệ thống thư điện tử, không thể gây thương tích mà thường rất phải

đương đầu với nhiều vấn nạn như virut máy tính các thư nhũng lam(*spam mail)* các thư quảng cáo (*advertisement mail*) và các thư khiêu dụ tình dục (*pornography mail*), đặc biệt là cho trẻ em, thì lại rất nhiều. Đối với các loại thư độc hại (*malicious mail*) này người dùng cần phải cài đặt thêm tiện ích và chức năng lọc swanx có trong phần mềm hay phải mua thêm.

Các dạng **chuyển tiếp** (*chain mail*) trong đó người nhận lại chuyển đi nội dung lá thư cho một hay nhiều người khác thường cũng phổ biến trong cả hai hệ thống bưu chính và thư điện tử. Khả năng ảnh hưởng về thông tin của hai loại này là tương đương mặc dù thư điện tử chuyển tiếp có nhiều xác suất gây nhiễm virus máy tính.

Hộp thư là nơi cất giữ các thư từ với địa chỉ hẳn hoi. Tương tự, trong hệ thống thư điện tử, thì hộp thư này tương đương với phần dữ liệu chứa nội dung các email cộng vói địa chỉ của người chủ thư điện tử. Điểm khác biệt ở đây là hộp thư điện tử sẽ có nhiều chức năng hơn là việc xoá bỏ các thư cũ.
Mỗi người có thể có một hay nhiều địa chỉ email (và phải được đăng ký qua một hệ thống nào đó). Mỗi hộp thư sẽ có một địa chỉ phân biệt không bao giờ trùng với địa chỉ email khác.
Như vậy có thể hoàn toàn không nhầm lẫn khi dùng danh từ **hộp thư điện tử** hay **hòm thư điện tử** (*email account*) để chỉ một phần mềm email đã được đăng kí dùng để nhận và gửi email cho một cá nhân.

- **Cấu trúc của một địa chỉ gmail (hay email)**

Một địa chỉ email sẽ bao gồm ba phần chính có dạng Tên_định_dạng_thêm tên_email@tên_miền

1. Phần tên_định_dạng_thêm: Đây là một dạng tên để cho người đọc có thể dễ dàng nhận ra người gửi hay nơi gửi. Tuy nhiên, trong các thư điện tử người ta có thể không cần cho tên định

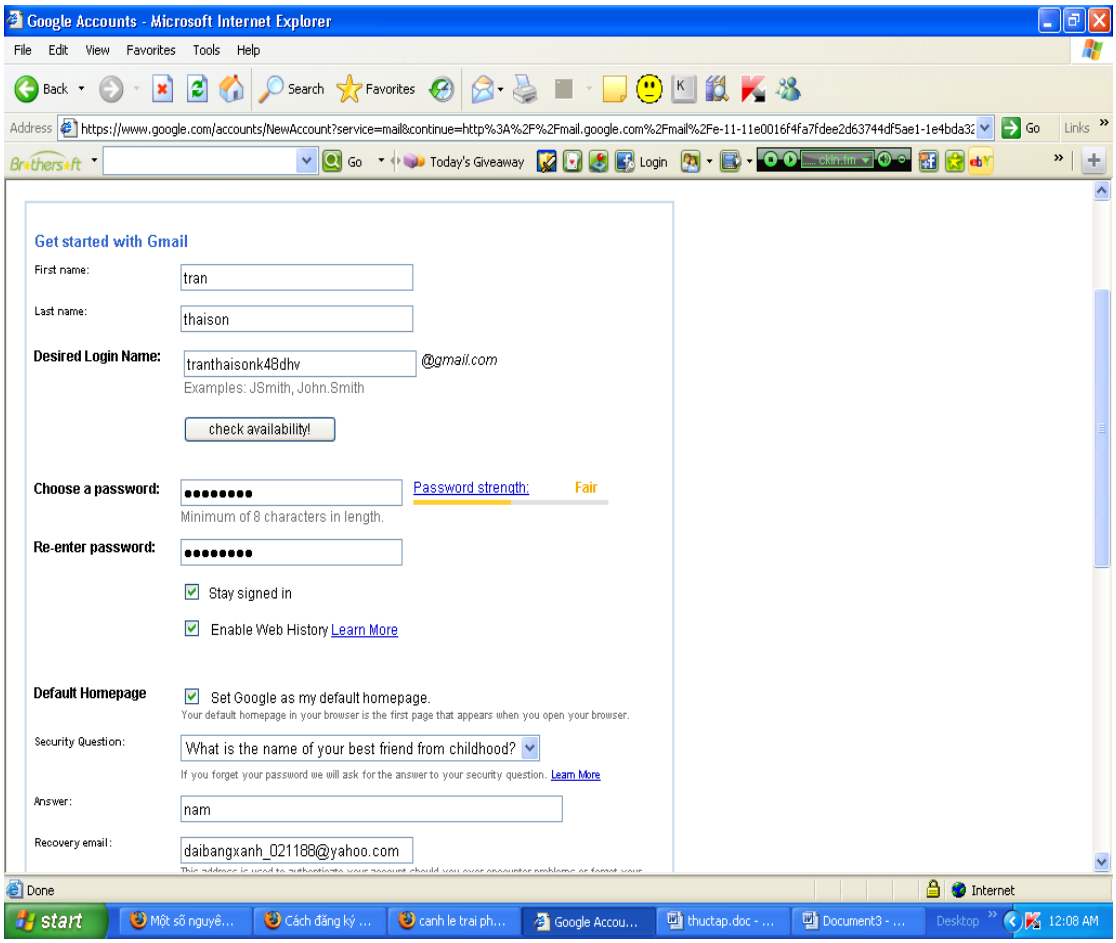
dạng và lá thư điện tử vẫn được gửi đi đúng nơi. Thí dụ: Trong địa chỉ gửi thư tới viết dưới dạng Nguyễn Thị A nguyenthia111@yahoo.com hay viết dưới dạng nguyenthia111@yahoo.com thì phần mềm thư điện tử vẫn hoạt động chính xác và gửi đi đến đúng địa chỉ.

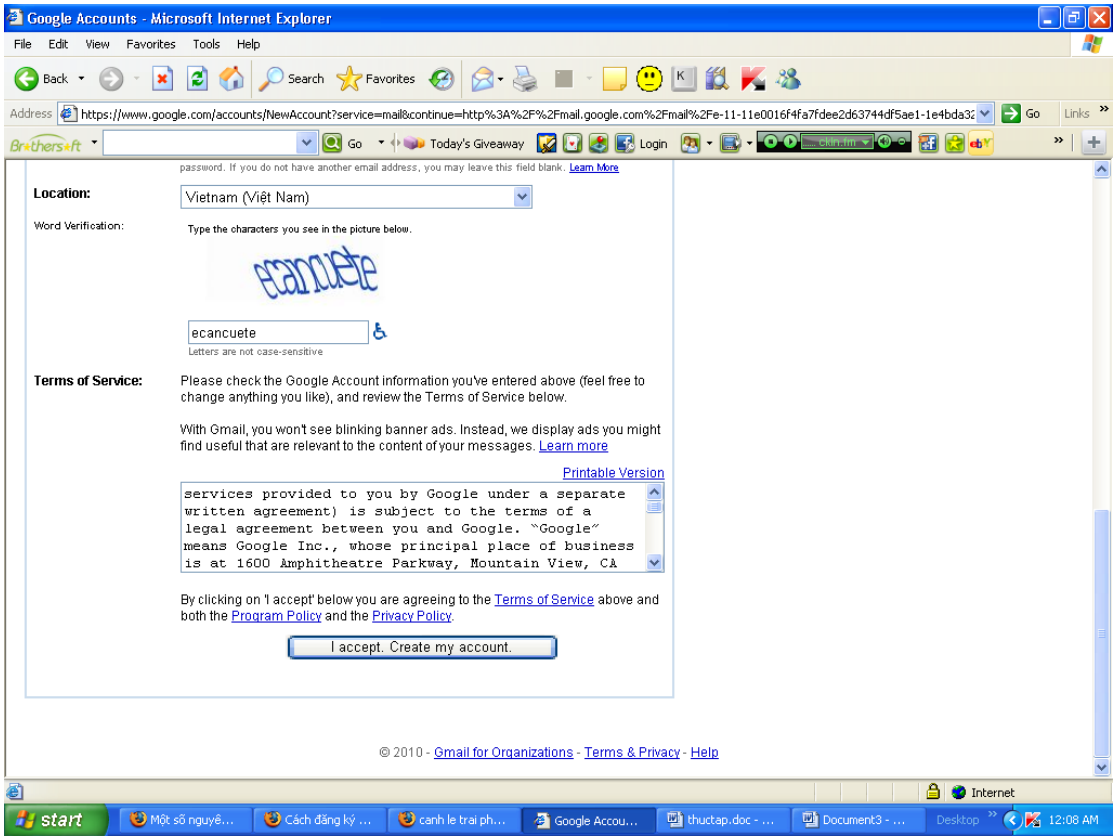
2. Phần tên_email: Đây là phần xác định hộp thư. Thông thường, cho dễ nhớ, phần này hay mang tên của người chủ ghép với một vài kí tự đặc biệt. Phần tên này thường do người đăng kí hộp thư điện tử đặt ra. Phần này còn được gọi là **Tên đại phương**
3. Phần tên_miền: Đây là tên miền nơi cung cấp thư điện tử.Ngay sau phần tên_email bắt đầu bằng chữ "@" nối liền sau đó là tên miền.

Cách đăng ký một thư điện tử. hiện nay có nhiều phần mềm thư điện tử tiện ích và dễ dáng sử dụng tuy nhiên đại đa số người dùng quan tâm đến Gmail bởi vì nó thông dụng hơn và khả năng lưu trữ lớn hơn. Để đăng ký một tài khoản Gmail làm như sau

Đầu tiên vào https://www.google.com sau đó chọn Gmail cửa sổ hiện ra

Chọn tạo tài khoản xong điền đầy đủ thông tin vào các mục ghi trên màn hình lưu ý ở màn hình bạn chú ý phần lựa chọn ngôn ngữ có thể lựa chọn ngôn ngữ mà người dung tùy chọn

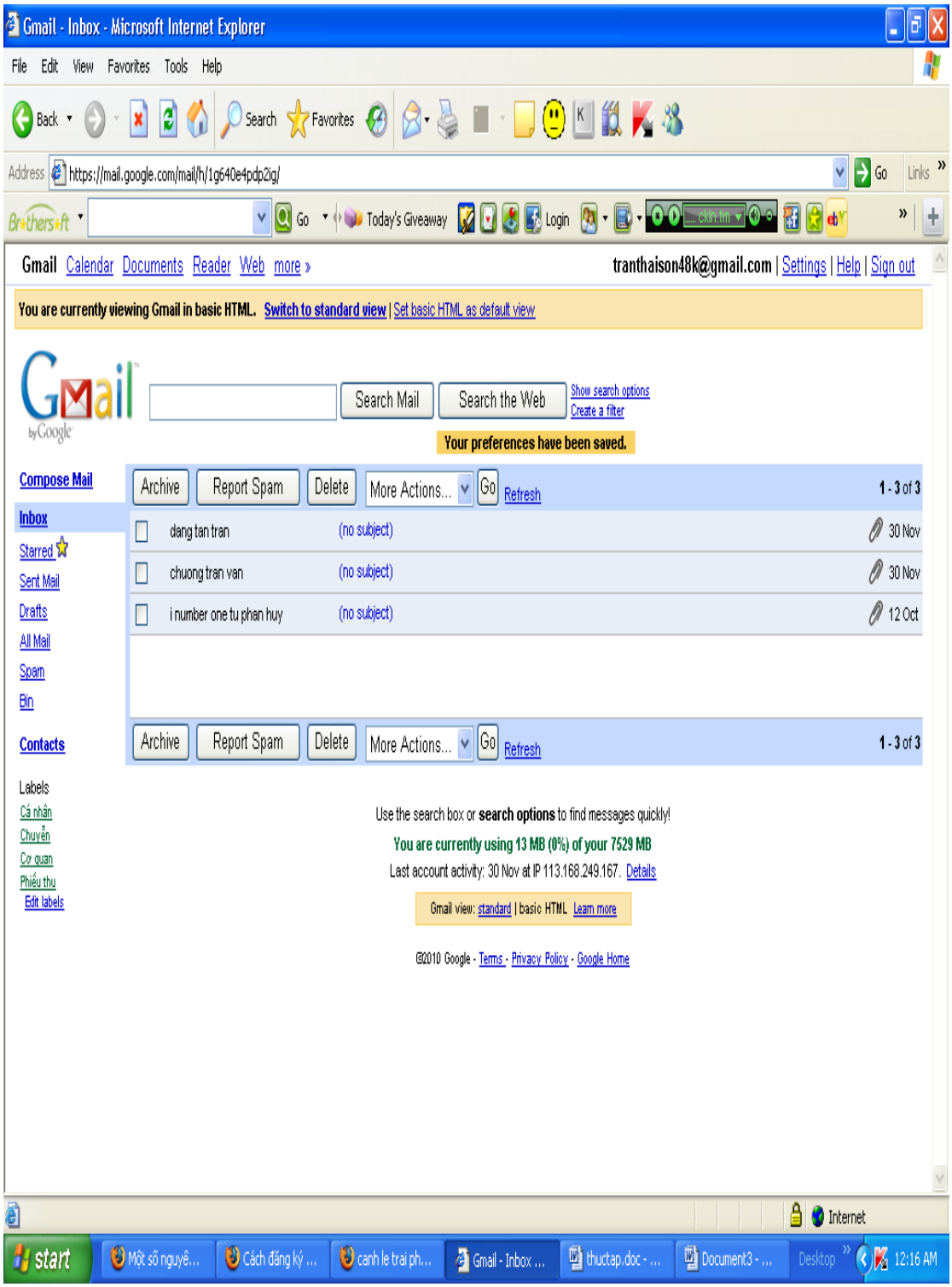

Google Accounts - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address https://www.google.com/accounts/NewAccount?service=mail&continue=http%3A%2F%2Fmail.google.com%2Fmail%2Fe-11-11e0016f4fa7fdee2d63744df5ae1-1e4bda3
Go Links
Today's Giveaway Login
Get started with Gmail
First name: tran
Last name: thaison
Desired Login Name: tranthaisonk48dhv @gmail.com
Examples: JSmith, John.Smith
check availability!
Choose a password: ••••••••
Password strength: Fair
Minimum of 8 characters in length.
Re-enter password: ••••••••
Stay signed in
Enable Web History Learn More
Default Homepage
Set Google as my default homepage.
Your default homepage in your browser is the first page that appears when you open your browser.
Security Question: What is the name of your best friend from childhood?
If you forget your password we will ask for the answer to your security question. Learn More
Answer: nam
Recovery email: daibangxanh_021188@yahoo.com
Done
Internet
start
Một số nguyê...
Cách đăng ký ...
canh le trai ph...
Google Accou...
thuctap.doc - ...
Document3 - ...
Desktop
12:08 AM


Google Accounts - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address https://www.google.com/accounts/NewAccount?service=mail&continue=http%3A%2F%2Fmail.google.com%2Fmail%2Fe-11-11e0016f4fa7fdee2d63744df5ae1-1e4bda3
Go Links
Today's Giveaway Login
password. If you do not have another email address, you may leave this field blank. Learn More
Location: Vietnam (Việt Nam)
Word Verification:
Type the characters you see in the picture below.
ecancuete
Letters are not case-sensitive
Terms of Service:
Please check the Google Account information you've entered above (feel free to change anything you like), and review the Terms of Service below.
With Gmail, you won't see blinking banner ads. Instead, we display ads you might find useful that are relevant to the content of your messages. Learn more
Printable Version
services provided to you by Google under a separate written agreement) is subject to the terms of a legal agreement between you and Google. "Google" means Google Inc., whose principal place of business is at 1600 Amphitheatre Parkway, Mountain View, CA
By clicking on 'I accept' below you are agreeing to the Terms of Service above and both the Program Policy and the Privacy Policy.
I accept. Create my account.
© 2010 - Gmail for Organizations - Terms & Privacy - Help
Internet
start
Một số nguyê...
Cách đăng ký ...
canh le trai ph...
Google Accou...
thuctap.doc - ...
Document3 - ...
Desktop
12:08 AM

Cửa sổ đăng ký

Sau khi điền đầy đủ thông tin vào trên chọn I accepe Create my account (tạo tài khoản)

Sau khi lam xong sẽ được



Chức năng của một thư điện thử

Ngoài chức năng thông thường để nhận và soạn thảo email, các phần mềm thư điện tử có thể còn cung cấp thêm những chức năng khác như là:
Lịch làm việc (*calendar*): người ta có thể dùng nó như là một thời khoá biểu. Trong những phần mềm mạnh, chức năng này còn giữ nhiệm vụ thông báo sự kiện đã đăng kí trong lịch làm việc trước giờ xảy ra cho người chủ hộp thư.
Sổ địa chỉ (*addresses* hay *contacts*): dùng để ghi nhớ tất cả các địa chỉ cần thiết cho công việc hay cho cá nhân.
Sổ tay (*note book* hay *notes*): để ghi chép, hay ghi nhớ bất kì điều gì.
Công cụ tìm kiếm thư điện tử (*find* hay *search mail*).
Để hiểu hết tất cả các chức năng của một phần mềm thư điện tử người dùng có thể dùng chức năng giúp đỡ (thường có thể mở chức năng này bằng cách nhấn nút <F1> bên trong phần mềm thư điện tử

Các mệnh lệnh Anh ngữ để đi vào các ngăn chứa thư

Đây thực ra chỉ là các ngăn chứa thư từ đã được phân loại theo tình trạng của các email cho tiện dùng. Người chủ thư có thể tự mình xếp loại các mail này hay chúng được xếp một cách tự động (do cài đặt hay do mặc định).

Inbox có nghĩa là Hộp thư nhận hay Hộp thư vào: Đây là ngăn đựng các thư mới nhận về.

ở thư mục inbox là nơi mà mà lưu trữ những thư mà người khác người gửi đến bạn có thể truy nhập xem thông tin những nội dung mà hộp thư đến bất cứ lúc nào mà ban đăng nhập vào Gmail

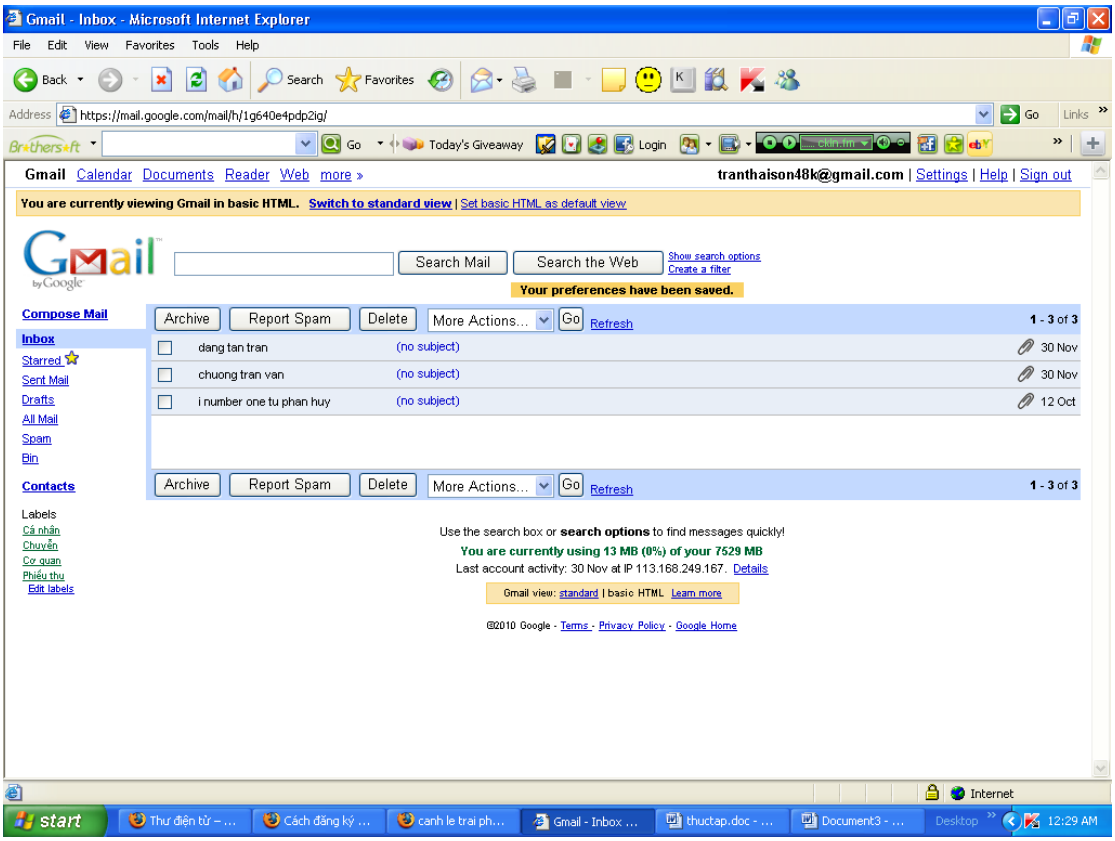
Outbox có nghĩa là Hộp thư gửi hay Hộp thư ra: Đây là ngăn đựng các thư đang chờ được gửi đi. Thông thường, nếu hệ thống email hoạt động tốt thì các thư nằm trong hộp này chỉ trong thời gian rất ngắn (vài giây đến vài phút là tối đa). Do đó, ngăn chứa này thường là một ngăn trống.

Draft có nghĩa là Ngăn nháp: Để chứa các email chưa hoàn tất hay đã hoàn tất nhưng chủ thư chưa muốn gửi đi.

Trash, *Trash can* hay *Deleted Item* có nghĩa là Ngăn xóa: Còn có thể gọi là Thùng rác hay Ngăn thư đã xóa. Đây là chỗ dự phòng tạm thời chứa các email đã xóa bỏ trong một thời gian. Chức năng này tiện lợi để phục hồi hay đọc lại các thư điện tử cần thiết đã lỡ tay bị xóa.

Sent, *sent Messages* hay *Sent Item* có nghĩa là Ngăn đã gửi: Nơi này dùng để chứa các thư đã gửi

Junk hay *Bulk* có nghĩa là Ngăn thư linh tinh: Đây là nơi chứa các mail đã được lọc và bị loại ra một cách tự động, còn được gọi là Thùng thư rác hay Ngăn chứa tạp thư. Thường thì nơi này sẽ chứa các thư quảng cáo, các thư nhũng lạm, các thư được gởi đến một số lượng lớn địa chỉ có cùng một nội dung, hay các loại thư độc hại ...

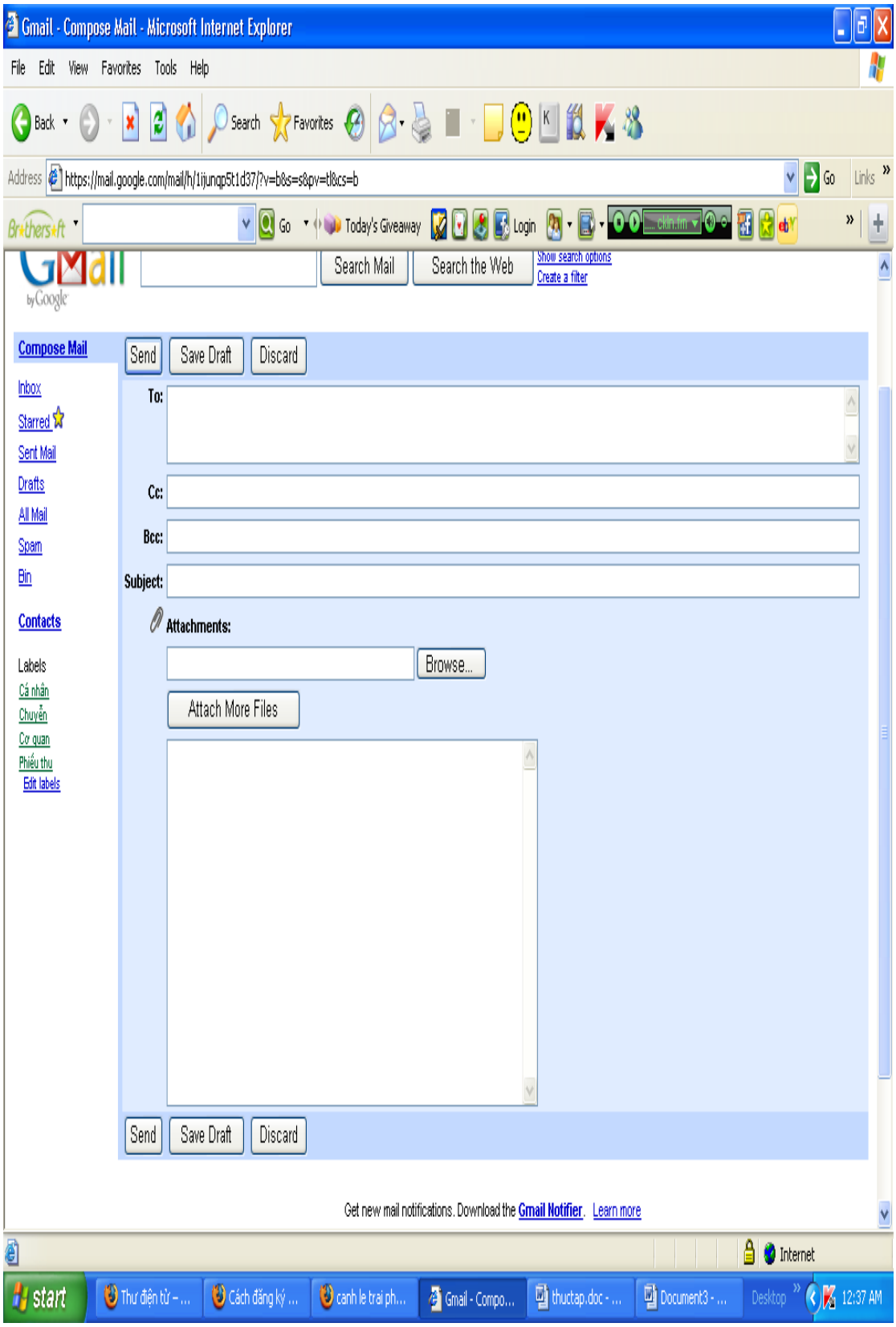


Inbox(thư đến)

Các mệnh lệnh Anh ngữ thường thấy trong một phần mềm thư điện tử

New hay *compose* có nghĩa là Thảo thư mới: Đây là mệnh lệnh cho phép bắt đầu soạn thảo một email mới.

Send có nghĩa là Gửi: Mệnh lệnh này sẽ tức khắc gửi thư tới các địa chỉ trong phần *To*, *CC*, và *BCC*



Soạn thư

To có nghĩa là Đến: Chổ chứa địa chỉ của các người nhận.

CC (từ chữ *carbon copies*) có nghĩa là Gửi kèm: Đây là chỗ chứa thêm địa chỉ gửi kèm, ngoài địa chỉ chính trong phần *To* bên trên. Các hộp thư nhận sẽ đọc được các địa chỉ người gửi và các địa chỉ gửi kèm này.

BCC (từ chữ *blind carbon copies*) có nghĩa là Gửi kèm kín: Đây cũng là chỗ ghi các địa chỉ mà lá thư sẽ được gửi kèm tới, nhưng các địa chỉ này sẽ được dấu kín không cho những người trong phần *To* hay phần *CC* biết là có sự đính kèm đến các địa chỉ nêu trong phần *BCC*.

Subject có nghĩa là Đề mục: Chỗ này thường để tóm tắt ý chính của lá thư hay chỗ ghi ngắn gọn điều quan trọng trong thư

Save as Draft hay *Save Draft* có nghĩa là Lưu bản nháp: Mệnh lệnh này sẽ giúp lưu giữ lá thư đang soạn thảo và đưa vào ngăn chứa *Darft* để có thể dùng lại về sau.

Attach hay *Attach Files* có nghĩa là Đính kèm: Đây là lệnh để người soạn email có thể gửi đính kèm theo lá thư các tập tin khác. Các tập tin này không giới hạn kiểu cấu trúc của nó, nghĩa là chúng có thể là các loại tập tin hình vẽ, phim, nhạc,... và ngay cả

Vấn đề an toàn khi sử dụng thư điện thử

Sự bảo mật của các thư từ điện tử còn có nhiều khuyết điểm. Trong hệ thống máy vi tính, những người có quyền đặc biệt vẫn có thể đọc thư của người khác trong bất cứ hộp thư nào trên máy. Ngoài ra thư có thể bị đọc tại các trạm phục vụ thư hoặc trên đường đi. Để tránh tình trạng này, người sử dụng có thể dùng mật mã để làm đảo lộn vị trí và mặt chữ để bảo tồn sự bí mật của lá thư. Ngoài ra người sử dụng còn phải tuân thủ các nguyên tắc sau đây để bảo đảm an toàn khi sử dụng thư điện tử

Không mở bất kỳ file đính kèm được gởi từ một địa chỉ email mà bạn không biết rõ hoặc không tin tưởng

Không mở bất kỳ email nào mà bạn cảm thấy nghi ngờ, thậm chí cả khi email này được gởi từ bạn bè hoặc khách hàng của bạn. Hầu hết virus

được lan truyền qua đường email. Do vậy, nếu bạn không chắc chắn về một email nào thì hãy tìm cách xác nhận lại từ phía người gửi

Không mở các file đính kèm theo các email có tiêu đề hấp dẫn hoặc thu hút. Ví dụ như: "Look,my beautiful girl friend","Congratulations","SOS",... Nếu bạn nhất quyết muốn mở các file đính kèm này, hãy lưu chúng vào đĩa cứng và dùng một chương trình diệt virus được cập nhật thông tin về virus mới nhất để kiểm tra.

Không mở các file đính kèm theo các email có tên file liên quan đến sex như "PORNO.EXE", "PAMELA_NUDE.VBS", "Britney Spears.scr",... Đây là các thủ đoạn dùng để đánh lừa người dùng của những kẻ viết virus

Xóa các email không rõ hoặc không mong muốn. Đừng forward email này cho bất kỳ ai hoặc reply lại cho người gửi. Những email này thường là các spam email. Mục đích của các spam email chỉ để quảng cáo hay làm nghẽn đường truyền Internet.

Không copy vào đĩa cứng bất kỳ file nào mà bạn không biết rõ hoặc không tin tưởng về nguồn gốc xuất phát của nó

Hãy cẩn thận khi tải các file từ Internet về đĩa cứng của máy tính. Dùng một chương trình diệt virus được cập nhật thường xuyên để kiểm tra các file này. Nếu bạn nghi ngờ về một file chương trình hoặc một email thì đừng bao giờ mở nó ra hoặc tải về máy tính của mình. Cách tốt nhất trong trường hợp này là xóa chúng hoặc không tải về máy tính của bạn.
Dùng một chương trình diệt virus tin cậy và được cập nhật thường xuyên như: Norton Anti-Virus, Mcaffee, Trend Micro,... Dùng các chương trình diệt virus có thể chạy thường trú trong bộ nhớ để chúng có thể giám sát thường xuyên các hoạt động trên máy tính của bạn

Nếu máy tính bạn có cài chương trình diệt virus, hãy cập nhật chúng thường xuyên. Trung bình mỗi tháng có tới 500 virus mới được phát hiện.

Do vậy, một chương trình diệt virus được cập nhật thường xuyên sẽ mang đầy đủ các thông tin về các loại virus mới và cách diệt. Việc cập nhật thường xuyên này sẽ giúp cho máy tính của bạn trở nên miễn nhiễm trước các loại virus mới.

Thực hiện việc sao lưu các dữ liệu quan trọng thường xuyên. Nếu chẳng may virus xóa tất cả các dữ liệu trên máy tính của bạn thì vẫn còn có khả năng phục hồi các dữ liệu quan trọng này. Các bản sao lưu này nên được cất giữ tại một vị trí riêng biệt hoặc cất giữ trên máy tính khác

Các lời khuyên để hạn chế Spam

Vấn đề gì làm phiền người dùng trên Internet nhất? Những email độc hại, không mong muốn, những bức thư chào hàng, sản phẩm, giải trí... Không như những bức thư bằng giấy thông thường người gửi phải trả tiền. Spammer không phải trả tiền cho hàng nghìn hoặc hàng trăm nghìn bức điện thư. Nếu nhà cung cấp dịch vụ cho bạn không ngăn chặn thì bạn sẽ phải chịu đựng tác động của những email không mong muốn trên mạng. Những nhà cung cấp dịch vụ ví dụ như AOL, MSN ... thường là cung cấp ngăn chặn spam trước khi gửi đến hộp thư của bạn

Một số nước trên thế giới có luật ngăn chặn việc sử dụng spam và các spammer phải chịu trách nhiệm các hành động của mình. Nhưng sự thực là spam vẫn ngày một tăng, bạn sẽ tự hỏi tại sao spammer bị lên án như vậy mà họ vẫn spam? Bởi vì công việc gửi thư quảng cáo thì sẽ rẻ hơn và đơn giản hơn và đem lại nhiều lợi nhuận cho spammer.

Đáng tiếc là spam đã và đang tồn tại. Và điều đó có nghĩa bạn sẽ có lúc bị ảnh hưởng của spam và sau đây là các lời khuyên để chống lại spam

Bảo vệ địa chỉ e-mail

Spammers (người sử dụng spam) có thể mua được danh sách địa chỉ email hoặc sử dụng các chương trình phần mềm để lấy địa chỉ email trên Internet. Nếu địa chỉ của bạn được đưa lên các nhóm thảo luận, trên các website, chat room... nhiều thì khả năng bị spam càng lớn. Do đó bạn chỉ nên đưa địa chỉ email ra công cộng khi nào thực sự cần thiết

Thiết lập nhiều địa chỉ email

Bạn nên sử dụng nhiều địa chỉ email. Mỗi một địa chỉ sử dụng cho một mục đích riêng. Ví dụ: địa chỉ sử dụng cho công việc, địa chỉ tham gia các nhóm thảo luận, địa chỉ cho bạn bè và gia đình.

Sử dụng lọc spam

Rất nhiều chương trình email như Outlook Express có tích hợp các dụng cụ cho phép chặn các bức thư từ những địa chỉ xác định hoặc dựa trên các từ khoá được bạn xác định trước (như filter). Hãy sử dụng chức năng đó để hạn chế các thư không cần thiết và tiết kiệm thời gian của bạn.

Sử dụng các chương trình chống spam (anti-spam software)

Bạn có thể cài các chương trình để giảm thiểu spam. Một số chương trình sử dụng phương thức so sánh các message đến giống nhau và điền nó vào danh sách là spammer. Một số khác chỉ cho phép các địa chỉ được chấp nhận mới được phép gửi đến.

Không trả lời

Spammers tiếp tục hành động của họ là vì nó có hiệu quả. Dừng hành động của họ bằng cách tẩy chay, không mua hàng hoá mà họ quảng cáo. Một số spammer thông minh còn thêm cả các hướng dẫn phía cuối của message của họ làm thế nào để ra khỏi danh sách nhận message từ họ nhưng thực ra khi bạn bấm vào đó là khẳng định rằng địa chỉ email của bạn là có tồn tại, bạn có đọc thư của họ và kết quả là bạn sẽ nhận nhiều hơn thư làm phiền
(junk mail).

Không trả đũa

Sau khi nhận được hàng tá message không mong muốn và nó làm phiền bạn. Một cách tự nhiên bạn sẽ tức giận và trả đũa lại người gửi và kết quả là bạn nhận được nhiều thư spam hơn và tài nguyên mạng bị lãng phí nhiều hơn.

Lựa chọn tham gia (Opt-out)

Rất nhiều website bây giờ yêu cầu bạn phải đăng ký để sử dụng dịch vụ.

Trước khi bạn đăng ký, bạn nên xem các yêu cầu, chính sách để xem người cung cấp web sẽ sử dụng thông tin cá nhân như địa chỉ email của bạn như thế nào. Có thể họ sẽ sử dụng địa chỉ cho mục đích thương mại bạn có thể chọn có hoặc không.

Loại bỏ địa chỉ khỏi các dịch vụ không cần thiết

Địa chỉ của bạn có thể trong một danh sách dịch vụ như yahoo group ... rất dễ dàng cho spammer lấy được địa chỉ của bạn. Để ngăn cho địa chỉ của bạn bị tiết lộ thì khi không cần thiết yêu cầu hãy thoát khỏi danh sách dịch vụ.

Xoá spam messages

Cách đề kháng hiệu quả nhất là đánh dấu và bỏ hết chúng vào thùng rác. Nếu mọi người đều lờ đi các spam message thì nó sẽ

PHẦN II: DÙNG THUẬT TOÁN SINH HOẶC THUẬT TOÁN QUAY LUI

I.Mô tả thuật toán sinh

Phương pháp sinh có thể áp dụng để giải quyết bài toán liệt kê tổ hợp đặt ra nếu như hai điều kiện trên thỏa mãn điều kiện:

- Có thể xác định được một thứ tự trên tập các cấu hình tổ hợp cần liệt kê,từ đó xác định được cấu hình đầu tiên và cấu hình cuối cùng.
- Từ một cấu hình bất kỳ chưa phải là cấu hình cuối cùng,đều có thể xây dựng được một thuật toán để suy ra cấu hình kế tiếp.

Tổng quát,thuật toán sinh kế tiếp có thể được mô tả bằng thủ tục genarate,trong đó Sinhketiep sẽ gán cho stop giá trị true,ngược lại cấu hình sẽ được sinh ra.

```
Procedure generate;
 Begin
```

```
<Xây dựng cấu hình ban đầu>
Stop:=false;
While not stop do
Begin
        <Đưa ra cấu hình đang có>
        Sinhketiep;
End;
End;
```

Khái niệm "Thứ tự từ diển"

Trên các kiểu dữ liệu đơn giản chuẩn,người ta thường nói tới khái niệm thứ tự.Ví dụ trên kiểu số thì có quan hệ: 1<2 ; 2<3 ; 3<10… trên kiểu ký tự thi có quan hệ

'A' < 'B';'C' < 'c'…

Trên các dãy hữu hạn người ta cũng xác định một quan hệ thứ tự:

Xét a[1…n] va b[1…n] là dãy 2 ngôi độ dài n,trên các phần tử của a và b đã có quan hệ thứ tự "≤".Khi đó a ≤ b nếu như:

Hoặc a[i]=b[i] với i:1≤i≤n.

Hoặc tồn tại một số nguyên dương k:1≤k<n để:

```
a[1]=b[1]
a[2]=b[2]
….
a[k-1]=b[k-1]
a[k]=b[k]
a[k+1]<b[k+1]
```

Trong trường hợp này có thể viết a<b.

Thứ tự đó gọi là thứ tự từ điển trên các dãy có độ dài n.

Khi độ dài a và b không bằng nhau,người ta cũng xác định được thứ tự từ điển bằng cách thêm vào cuối dãy a hoặc dãy b những phần tử đặc biệt gọi là phần tử để độ dài của a và b bằng nhau,và coi những phần tử này nhỏ hơn tất cả các phần tử khác.ta lại đưa về xác định thứ tự từ điển của hai dãy cùng độ dài.Ví dụ:

(1,2,3,4)<(5,6)

(a,b,c)<(a,b,c,d)

2.1.Sinh hoán vị

Bài toán:

Nhập vào một số tự nhiên n, một mảng a[ ] gồm n phần tử nguyên liên tiếp từ 1 đến n. Hãy liệt kê tất cả các hoán vị n phần tử của a[ ] theo thứ tự từ điển.

Mô tả bài toán:

- Cho tập X = { 1,2,3,..., n}. Hãy liệt kê tất cả các hoán vị của tập này.
- Một hoán vị của X là một bộ A = (a_1, a_2,..., a_n) với $a_i \neq a_j$ nếu $i \neq j$
- Định nghĩa 1 thứ tự:

A = (a_1, a_2,...,a_{k-1}, a_k, ... a_n) là hoán vị trước của

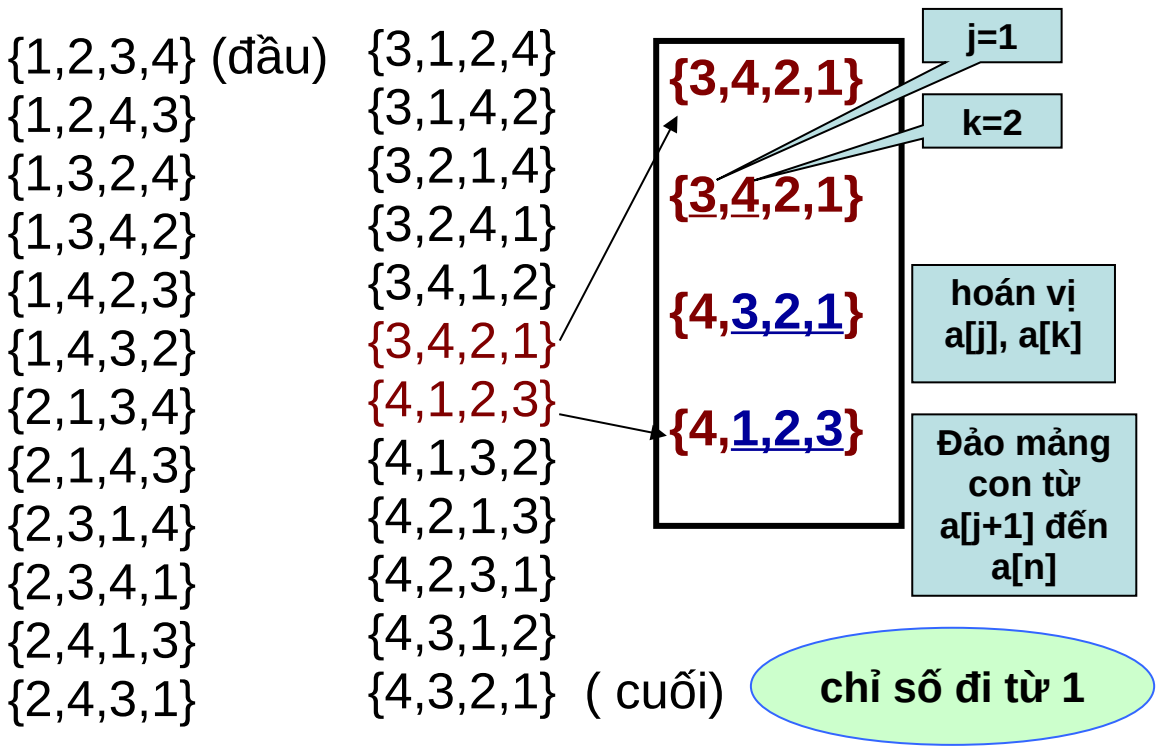
A'= (a'$_1$, a'$_2$,...,a'$_{k-1}$, a'$_k$, ... a'$_n$) nếu tìm được vị trí k sao cho $a_k < a'_k$

- Ví dụ : 1234567 là hoán vị trước của

1234657

- Đây chính là thứ tự từ điển.

Các hoán vị của X={1,2,3,4}



Giải thuật tìm hoán vị kế tiếp

Nếu ghép các phần tử trong một hoán vị thành một số nguyên thì phải đảm bảo hoán vị sau phải lớn hơn hoán vị trước. Ta xây dựng thuật toán sinh hoán vị kế tiếp từ một hoán vị đã cho bằng phương pháp sinh hoán vị

Khai báo một mảng a[] là mảng chứa các số nguyên từ 1->n.

Bước 1. Tìm từ phải sang trái để được một dãy con tăng dần (tăng dần theo kiểu từ cuối về đầu). Khi nào gặp một phần tử a[i] lớn hơn phần tử a[i-1] thì dừng lại. Ví dụ với hoán vị: 2, 8, 4, 7, 6, 5, 3, 1, ta tìm được dãy con: 7, 6, 5, 3, 1. Để làm được điều này ta dùng vòng lặp for:

```
for(i = n - 1; i > 0; i--)
 if(a[i] > a[i-1]) break;
```

Bước 2. Sau khi tìm được dãy con tăng dần, ta chỉ làm việc với dãy này và một phần tử trước đó mà thôi, các phần tử còn lại không hề thay đổi. Xảy ra 2 khả năng:

Khả năng 1: Không tìm được phần tử a[i] nào sao cho a[i] > a[i-1]. Khi đó, i = 0 và ta được hoán vị cuối cùng, ví dụ: 8, 7, 6, 5, 4, 3, 2, 1. Ta viết lệnh return để dừng đệ quy.

```
if(i == 0) return;
```

Khả năng 2: Trường hợp còn lại. Trường hợp này ta thực hiện hai công việc *Công việc 1*. Đảo phải ngược dãy con tìm được từ dãy tăng dần từ phải sang trái thành trái sang phải. Ví dụ dãy: 2, 8, 4, 7, 6, 5, 3, 1, sau khi đảo ngược dãy con ta được: 2, 8, 4, 1, 3, 5, 6, 7. Ta khao báo hàm void DaoNguoc(int a[ ], int x, int y) để đảo ngược các phần tử từ vị trí thứ x đến vị trí thứ y của mảng a[ ].

```
void DaoNguoc(int a[], int x, int y)
  {
    int i = x, j = y;
    while(i < j)
    {
       TraoDoi(a[i], a[j]);
```

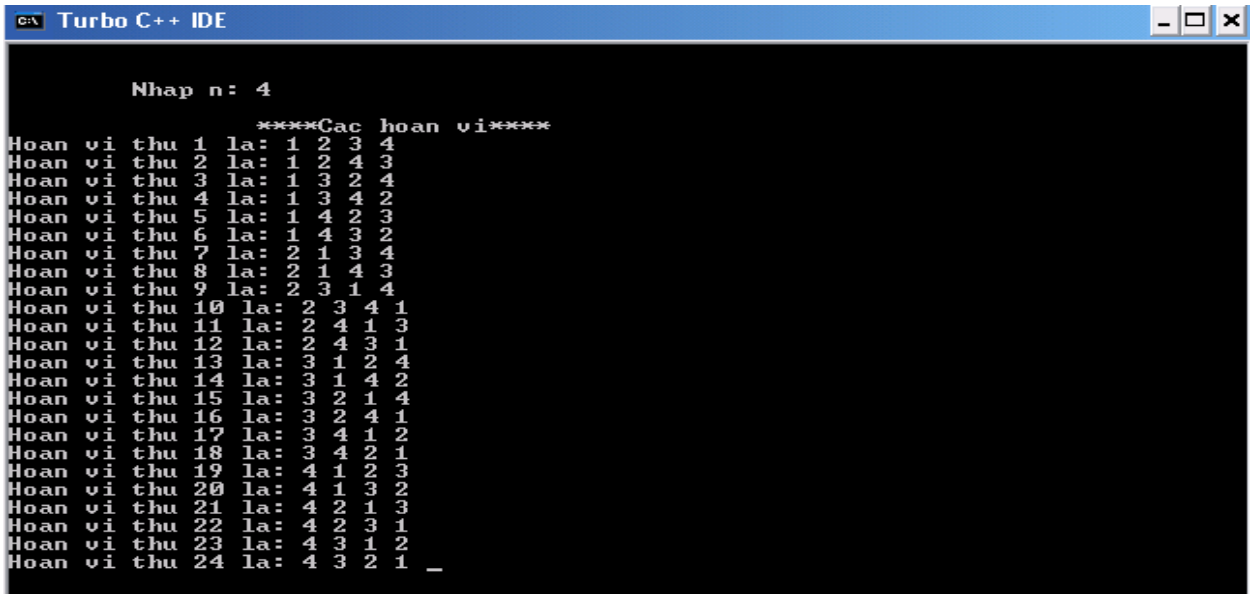
```
        i++;
        j--;
      }
  }
```

Công việc 2. Tìm trong dãy con vừa đảo ngược (tìm từ trái sang phải), nếu gặp phần tử a[j] nào đó mà lớn hơn a[i-1] (a[i-1] là phần tử liền trước của dãy con) thì đổi giá trị của hai phần tử này và dừng lại quá trình duyệt. Ví dụ hoán vị: 2, 8, 4, 1, 3, 5, 6, 7 ta được 2, 8, 5, 1, 3, 4, 6, 7.

```
for(j = i; j < n; j++)
    if(a[j] > a[i-1]) {TraoDoi(a[j], a[i-1]); break;}
```

Như vậy, từ 2, 8, 4, 7, 6, 5, 3, 1 cuối cùng ta được hoán vị kế tiếp là: 2, 8, 5, 1, 3, 4, 6, 7.Đến đây ta được hoán vị kế tiếp từ một hoán vị đã cho theo thứ tự từ điển

Khi nhập n=4 kết quả:



2.2.Bài toán sinh xâu nhị phân có độ dài n

Bài toán:

Nhập vào một số tự nhiên N,sinh các các xâu nhị phân có độ dài N có thể có theo thứ tự từ điển

Mô tả bài toán

một dãy nhị phân có độ dài n là một dãy $x=x_1x_2\ldots x_n$ trong đó các x_i {0,1} ($_i$:1≤i≤n).

Dễ thấy :một dãy nhị phân x độ dài n=số các số nguyên [0,2^n -1]=2^n. Ta sẽ lập chương trình liệt kê các dãy nhị phân theo thứ tự từ điển có nghĩa là sẽ liệt kê lần lượt các dãy nhị phân biểu diễn các số nguyên theo thứ tự 0,1…2^n-1.

Ví dụ:khi n=3 các dãy nhị phân độ dài bằng 3 được liệt kê như sau:

p(x)	0	1	2	3	4	5	6	7
x	000	001	010	011	100	101	110	111

Nhìn vào ví dụ trên ta thấy dãy đầu tiên sẽ là 00..0 và dãy cuối cùng sẽ là 11..1.Nhận xét rằng nếu dãy $x=(x_1x_{2,\ldots,}x_n)$ là dãy đang có và không phải là dãy cuối cùng thì dãy kế tiếp nhận được bằng cách cộng thêm 1(theo cơ số 2 có nhớ)vào dãy hiện tại.

Ví dụ:khi n=8:

Dãy đang có:	10010000	dãy đang có:	10010111
cộng thêm 1:	+1	cộng thêm 1:	+1
Dãy mới:	10010001	dãy mới:	10011000

Giải thuật tìm dãy nhị phân kế tiếp

Ta lấy số nhị phân đang có cộng với 1 thì được số nhị phân tiếp theo ứng với tổ hợp tiếp theo. Ví dụ: 1100111 + 1 = 1101000. Cách thực hiện phép cộng một số nhị phân với 1 theo phong cách của lập trình như sau:

– Nếu số nhị phân có tận cùng là bit 0 thì ta biến đổi bit này thành 1. Ví dụ 1100110 + 1 = 1100111.

– Nếu không, ta duyệt từ bít cuối trở về đầu, mỗi khi gặp bit 1 ta biến đổi nó thành bit 0 và chuyển sang bit tiếp theo. Cho đến khi gặp được bit 0, ta biến đổi bit này thành bit 1 và dừng ta được kết quả. Ví dụ: 10101111 + 1 =

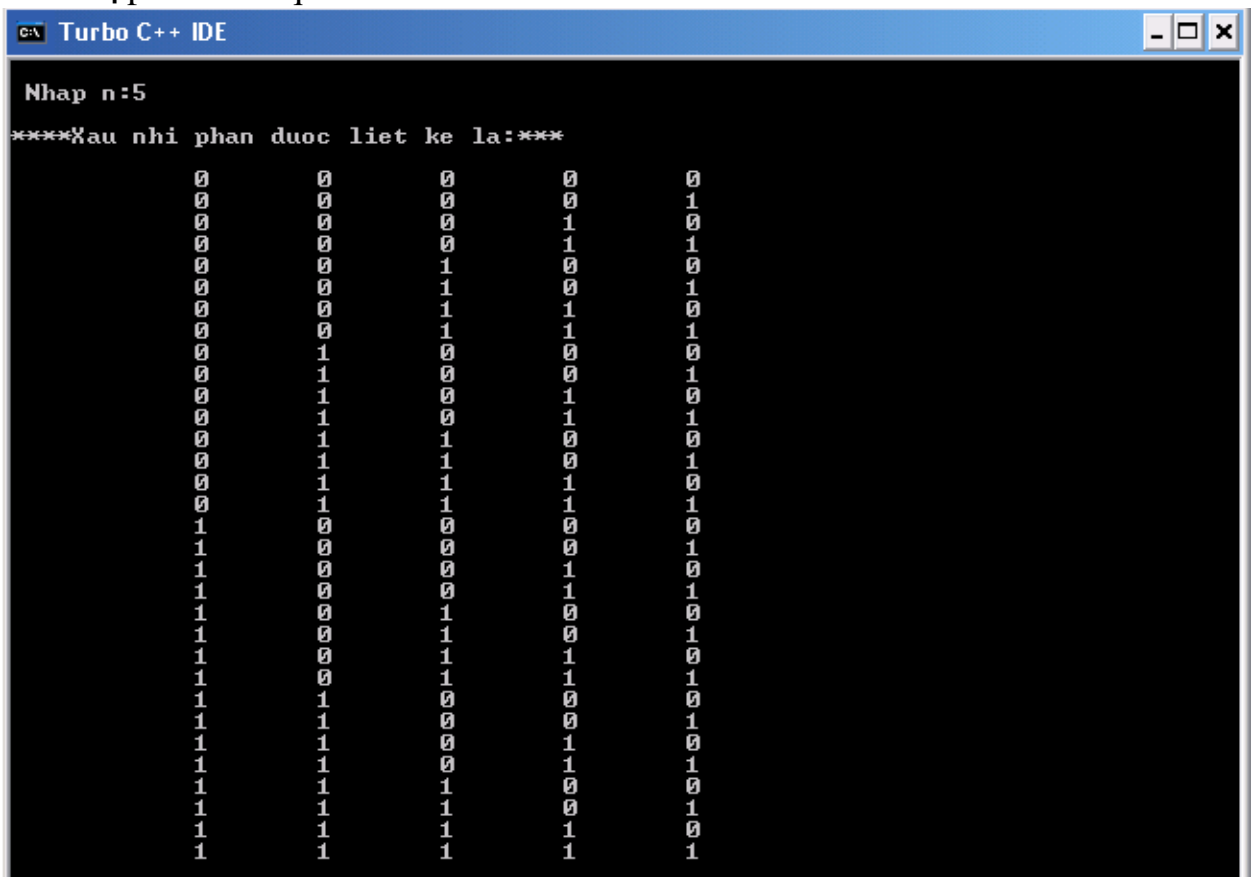
10110000.

– Nếu trường hợp duyệt về đến bit đầu tiên rồi mà vẫn chưa tìm thấy bit 0 thì có nghĩa là ta đã được tổ hợp cuối cùng.

Như vậy thuật sinh cấu hình kế tiếp từ cấu hình hiện tại có thể mô tả như sau:Xét từ cuối dãy về đầu (xét từ hàng đơn vị lên),gặp số 0 đầu tiên thì thay nó bằng số 1 và đặt tất cả các phần tử phía sau vị trí đó bằng 0.

```
void sinhketiep(int *A,int n, int &stop)
{
        int i=n;
         while(A[i]==1)
        {
                A[i]=0;
                i--;
        }
         if(i==0) stop= 1;
        else
        A[i]=1;
}
```

Khi nhập n=5 kết quả:



2.3. Sinh các tổ hợp

Bài toán:

Nhập vào một số tự nhiên N. Khởi tạo tập hợp A gồm N phần tử từ 1 đến N: A = {1, 2, ... , N). Hãy in ra tất cả các tổ hợp của tập A.

Mô tả bài toán:

Để giải quyết bài toán này chúng ta in ra toàn bộ các tập con của A, mỗi tập con k phần tử tương ứng là một tổ hợp chập k gồm N phần tử của A. Nhưng để in ra các tập con thì chúng ta phải biểu diễn tập con dưới dạng tập các bit nhị phân.

Để biểu diễn tập con của một tập hợp ta dùng một tập gồm các bít nhị phân (0 hoặc 1). Giả sử ta có tập A gồm N = 5 phần tử: A = {1, 2, 3, 4, 5}, nếu C là tập con của A gồm 3 phần tử: C = {2, 3, 5} thì ta biểu diễn C dưới dạng các bit nhị phân như sau: C = {0, 1, 1, 0, 1}.

Giải thuật sinh tổ hợp

Ta sẽ dùng biến n để biểu diễn số phần tử N, dùng mảng a[ ] để biểu diễn tập A, dùng mảng c[ ] để biểu diễn tập con C của A dưới dạng các bít nhị phân

– Khởi tạo mảng c[ ] gồm n bít 0: c[ ] = {0, 0, 0, ... , 0}, in ra màn hình tập C (lúc này là tập rỗng).

– Biến đổi c[ ] thành tổ hợp tiếp theo, và in ra màn hình tập C (dưới dạng các số nguyên chứ không phải số nhị phân)

– Tiếp tục biến đổi c[ ] thành tổ hợp tiếp theo và in ra màn hình cho đến khi được tổ hợp c[ ] = {1, 1, 1, ... , 1) thì dừng lại.

Trước hết khai báo n, a[20], c[20] ngoài hàm main() và không nằm trong bất kỳ hàm nào. Khi đó các biến này có phạm vi hoạt động là tất cả các hàm - nghĩa là toàn cục. Trong hàm main(), khởi tạo mảng a[ ] và c[ ]: a[0] = 1, a[1] = 2, ... , a[n-1] = n và toàn bộ các phần tử của c[ ] bằng 0.

```
#include<iostream.h>
#include<conio.h>
int n, a[20], c[20];
main()
```

```
{
   cout<<"Nhap vao so phan tu N: "; cin>>n;
   for(int i = 0; i < n; i++) { a[i] = i + 1; c[i] = 0; }
   int k = 0;
   SinhToHop(k); //Sinh tổ hợp thứ k
   getch();
}
```

Ta khai báo hàm này trước hàm main(). Hàm này cần thực hiện những công việc như sau:

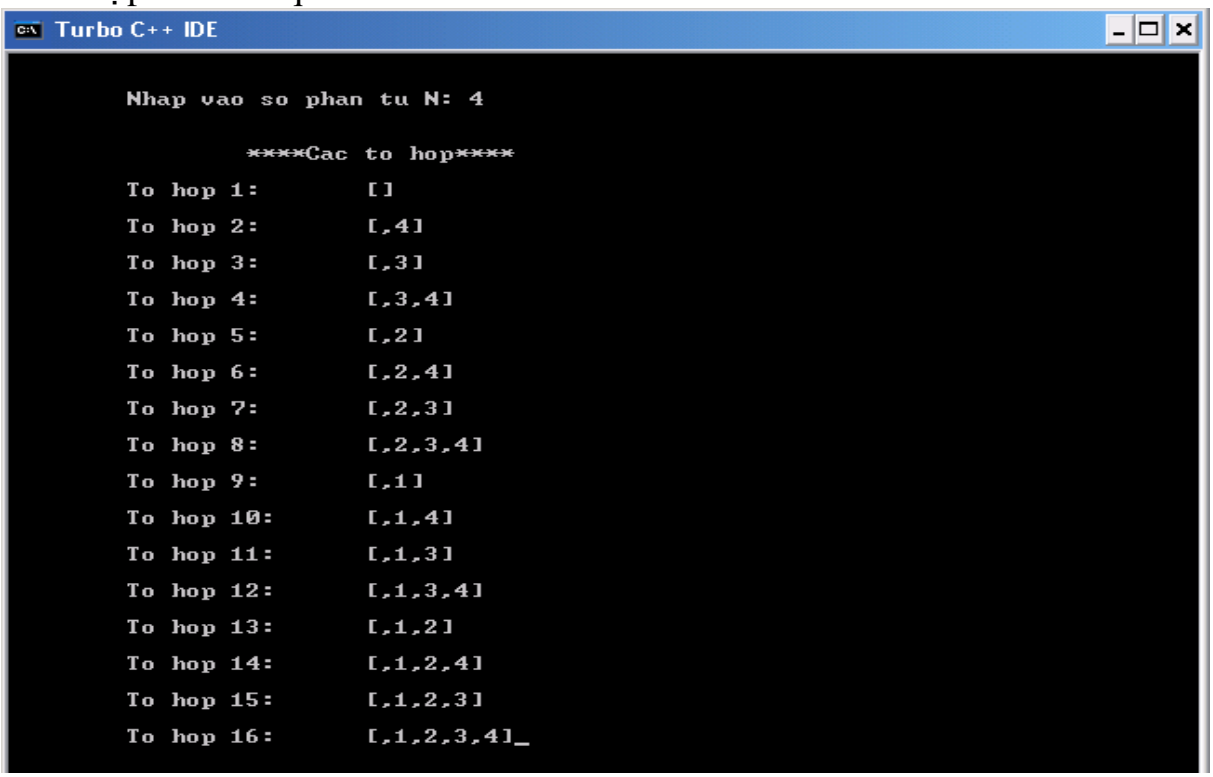
- Công việc 1: In ra tổ hợp thứ k, nghĩa là in ra mảng c[ ] nhưng không phải là dạng nhị phân mà là dạng số nguyên.
- Công việc 2: Biến đổi mảng c[ ] để trở thành tổ hợp tiếp theo.
- Công việc 3: In ra tổ hợp thứ k + 1 bằng cách gọi hàm SinhToHop(k + 1).

Hàm sinh tổ hợp được mô tả như sau:

```
void SinhToHop(int k)
{
   int i;
   //In ra tổ hợp thứ k
   if(k > 0) //Trường hợp k = 0 là tập rỗng
   {
      cout<<"\nTo hop "<<k<<": ";
      for(i = 0; i < n; i++) if(c[i] == 1) cout<<a[i];
   }
//Biến đổi c[] thành tổ hợp tiếp theo
   for(i = n - 1; i >= 0; i--)
      if(c[i] == 1)
      {
         c[i] = 0;
         if(i == 0) return; //Trường hợp ứng với dãy toàn bit 1
      }
```

```
        else
        {
           c[i] = 1;
           break; //Thoát khỏi vòng for
        }
    //In ra tổ hợp thứ k + 1
       SinhToHop(k + 1);
    }
```

Khi nhập n=5 kết quả:



PHẦN III: TÌM HIỂU VÀ KHAI THÁC DỊCH VỤ WINDOWS FIREWALL CỦA WINDOWS

1.1. Firewall là gì?

1.1.1.Lịch sử:

Công nghệ tường lửa bắt đầu xuất hiện vào cuối những năm 1980 khi Internet vẫn còn là một công nghệ khá mới mẻ theo khía cạnh kết nối và sử dụng trên toàn cầu. Nó được hình thành sau các vụ xâm phạm nghiêm trọng an ninh liên mạng. Năm 1988, một nhân viên nghiên cứu gửi một bản ghi nhớ qua thư điện tử tới đồng nghiệp rằng: "Chúng ta đang bị một con VIRUS Internet tấn công!" Cộng đồng mạng lúc đó không hề chuẩn bị cho một cuộc tấn công như vậy và hoàn toàn bị bất ngờ. Sau đó, cộng đồng Internet quyết định ưu tiên tối cao là phải ngăn chặn không cho một cuộc tấn công bất kỳ nào có thể xảy ra, họ bắt đầu cộng tác đưa ra các ý tưởng mới, những hệ thống và phần mềm mới làm cho mạng Internet có thể trở lại an toàn.

Năm 1988, bài báo đầu tiên về công nghệ tường lửa được công bố. Từ năm 1980 đến năm 1990, hai nhà nghiên cứu phát triển thế hệ tường lửa thứ hai, được biết đến với tên tường lửa tầng mạch (circuit level firewall). Các bài báo mô tả thế hệ tường lửa thứ ba, với tên gọi tường lửa tầng ứng dụng (application layer firewall)

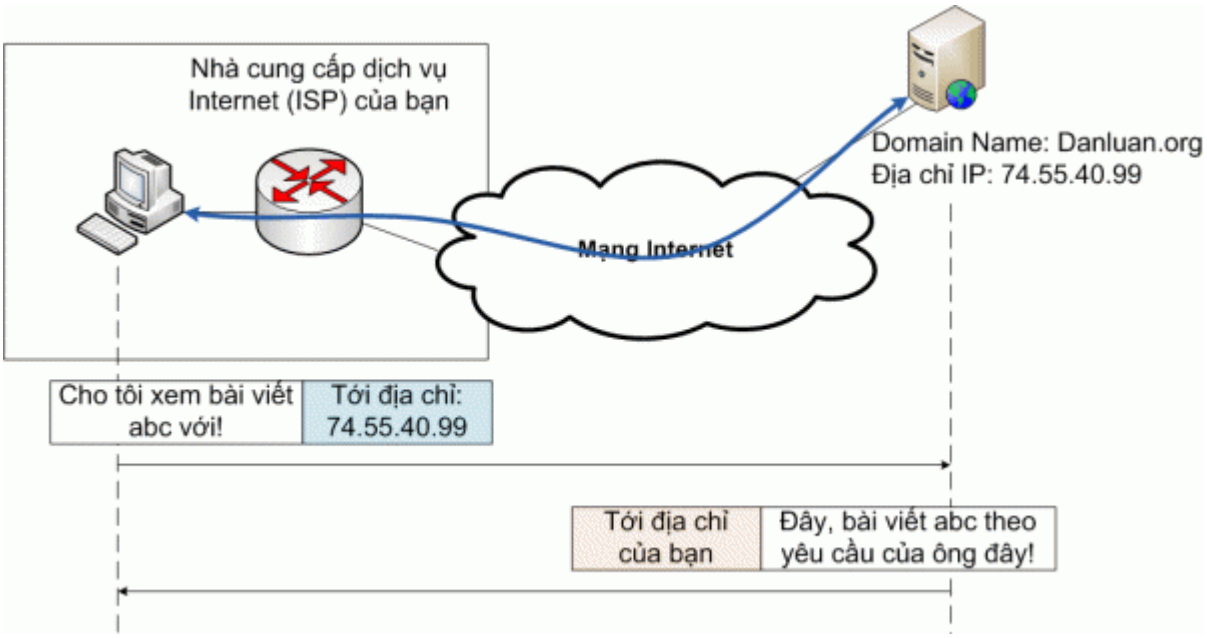
Bill Cheswick và Steve Bellovin tiếp tục nghiên cứu của họ về lọc gói tin và đã phát triển một mô hình chạy được cho công ty của họ. Năm 1992, Bob Braden và Annette DeSchon đã phát triển hệ thống tường lửa lọc gói tin thế hệ thứ tư. Sản phẩm có tên "Visas". Năm 1994 đã xây dựng sản phẩm phần mềm sẵn sàng cho sử dụng, đó là FireWall-1. Cisco, một trong những công ty an ninh mạng lớn nhất trên thế giới đã phát hành sản phẩm này năm 1997.

1.1.2.Kiến thức cơ bản về firewall:

Mạng Internet được xây dựng trên một bộ luật, hay tập hợp của các quy tắc chung, có tên là Giao thức Internet (viết tắt là IP). Giao thức chung này đảm bảo rằng các thiết bị Internet có thể "nói chuyện" được với nhau, như hệ thống mạng Internet ở Mỹ có thể giao tiếp được với hệ thống mạng Internet

ở Việt Nam, ở Lào, ở Campuchia v.v… Để giao tiếp được với nhau, mỗi thiết bị trên mạng được đánh một số riêng biệt, gọi là địa chỉ IP. Địa chỉ IP có tác dụng giống như số nhà để gửi thư vậy

Quá trình chuyển yêu cầu đọc một trang web và trả lời của server diễn ra (H1). Máy tính của bạn gửi yêu cầu tới nhà cung cấp dịch vụ ISP của bạn (giống bỏ thư vào thùng thư), nhà cung cấp sẽ đọc địa chỉ trên phong bì, và chuyển lá thư tới người nhận là server Danluan.org.

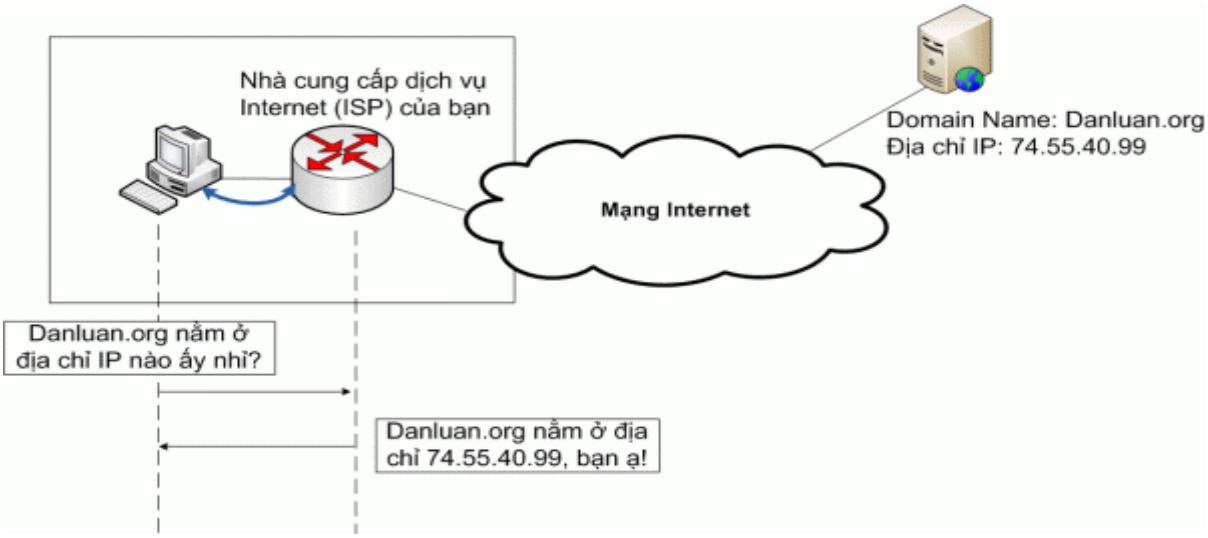


H1:Máy tính của bạn muốn xem một trang web từ Danluan.org

Một điểm cần chú ý là địa chỉ IP dưới dạng số rất thích hợp với máy tính, nhưng rất khó nhớ đối với con người. Do đó, người ta đưa ra cái gọi là Domain Name System (DNS), để đặt tên cho địa chỉ IP. Từ tên miền, dễ nhớ và gần gũi với con người, bạn có thể tìm ra địa chỉ IP của một trang web.

Như vậy, trước khi gửi bất cứ thứ gì tới Danluan.org, máy tính của bạn sẽ đi hỏi, đa phần là hỏi chính ISP của bạn, xem danluan.org tương ứng với

địa chỉ IPnào (H2)



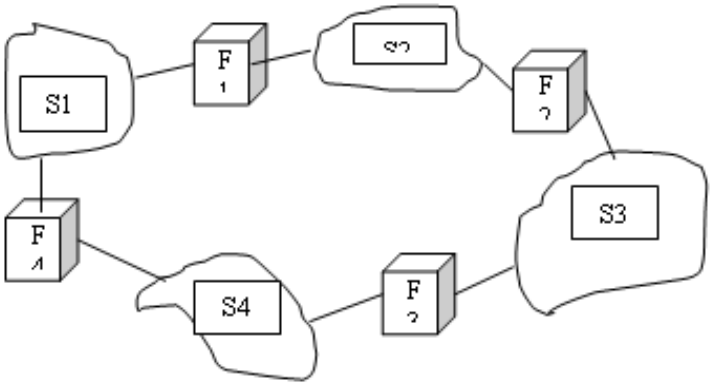
H 2: Tra địa chỉ IP của tên miền Danluan.org

Ở trên là trường hợp ISP của bạn không lắp đặt tường lửa. Chuyện gì xảy ra nếu ISP của bạn lắp tường lửa, và nó quyết định chặn trang *Danluan.org?*

1.1.3. Firewall hay tường lửa là gì?

Thuật ngữ FireWall có nguồn gốc từ một kỹ thuật thiết kế trong xây dựng để ngăn chặn, hạn chế hoả hoạn. Trong CNTT, FireWall là một kỹ thuật được tích hợp vào hệ thống mạng để chống lại sự truy cập trái phép nhằm bảo vệ các nguồn thông tin nội bộ cũng như hạn chế sự xâm nhập vào hệ thống của một số thông tin không mong muốn.Thông thường FireWall được đặt giữa mạng bên trong (intranet) của một tổ chức, một công ty, hay một quốc gia và Internet để thiết lập trong một mạng nội bộ hoặc cô lập các miền an toàn. Ví dụ như mô hình dưới đây thể hiện một mạng cục bộ sử dụng Firewall để ngăn cách phòng máy và hệ thống mạng ở tầng dưới:

-



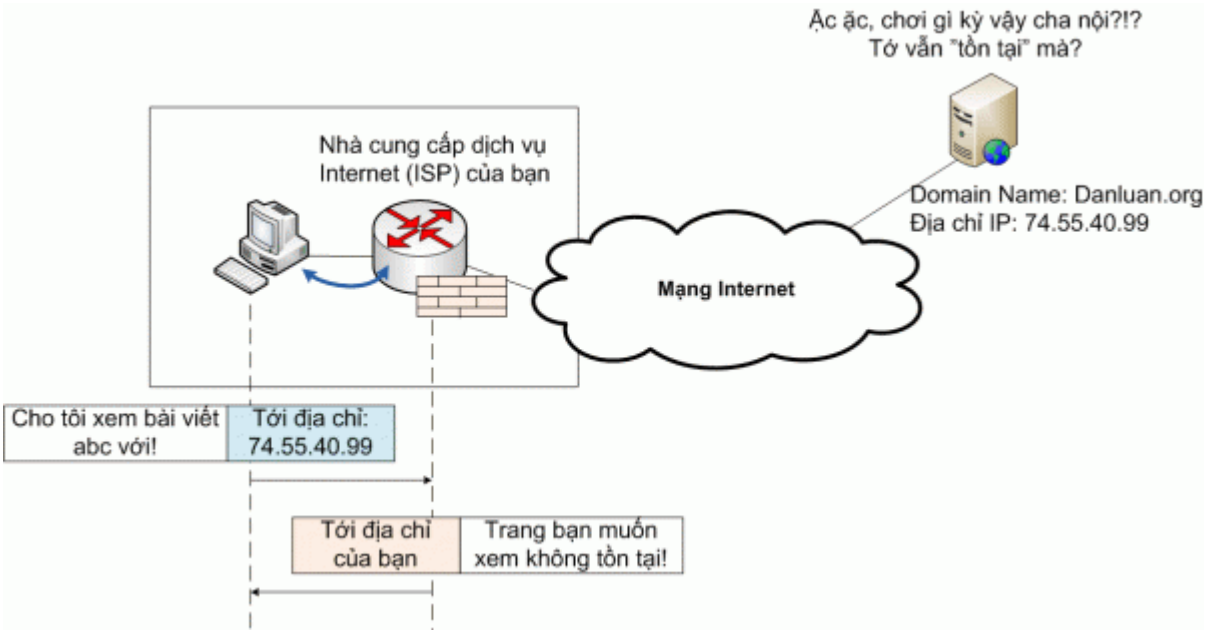
Nói cách khác, tường lửa (firewall) là hệ thống gồm cả phần cứng và phần mềm làm nhiệm vụ ngăn chặn các truy nhập "không mong muốn" từ trong ra ngoài hoặc từ ngoài vào trong. Tường lửa thường được đặt ở cổng giao tiếp giữa hai hệ thống mạng, ví dụ giữa mạng trong nước và mạng quốc tế, giữa mạng nội bộ của doanh nghiệp và mạng Internet công cộng v.v... để lọc thông tin theo các nguyên tắc được định trước.Vì thế các công ty lớn, các trung tâm nghiên cứu quan trọng cần tường lửa để loại bỏ các cuộc tấn công của tin tặc từ bên ngoài vào, hoặc để ngăn nhân viên gửi thông tin mật ra ngoài hay sử dụng các hệ thống khác trong giờ làm việc.
Có nhiều phương pháp lọc thông tin:

1.1.3.1.Lọc theo tên miền:

Khi máy của bạn hỏi FPT rằng danluan.org nằm ở địa chỉ IP nào, thì FPT sẽ từ chối trả lời, nếu trang danluan.org nằm trong danh sách "bị cấm". Nói cách khác, danluan.org không nằm trong cuốn "danh bạ điện thoại" của FPT. Phương pháp này có ưu điểm là... gọn và rẻ. Số lần người sử dụng hỏi DNS server mỗi ngày ít hơn nhiều so với số yêu cầu đọc trang web, do đó số lần phải đọc, kiểm tra xem trang này có bị cấm hay không, sẽ ít hơn. Ngược lại, đây là loại tường lửa dễ bị phá nhất. Vì nó chỉ ngăn cản truy cập dựa theo tên miền, nếu người sử dụng biết địa chỉ IP, họ có thể dùng trực tiếp địa chỉ IP để truy cập tới trang muốn xem mà khụng gặp trở ngại nào

1.1.3.2.Lọc theo địa chỉ IP

Loại tường lửa này chặt hơn tường lửa nói ở trên. Giả sử bạn đã tra được danluan.org nằm ở địa chỉ IP là 74.55.40.99, và bạn gửi yêu cầu đọc trang abc tới địa chỉ vừa tra. Tường lửa sẽ đọc gói tin, kiểm tra địa chỉ IP, nếu thấy nằm trong danh sách bị cấm, nó sẽ trả về thông báo: "Trang web không tồn tại" (H3), nó giống như người đưa thư, sau khi đọc địa chỉ thư, thấy có trong danh sách bị cấm, sẽ vứt thư đi.



H3: Tường lửa lọc IP chặn yêu cầu của người sử dụng dựa trên địa chỉ IP.

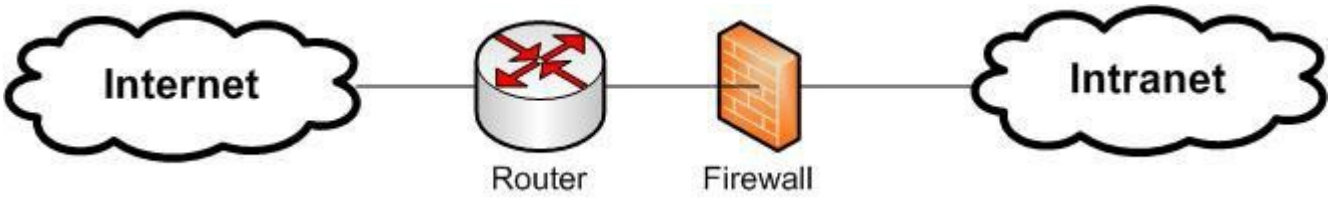
Ưu điểm của phương pháp lọc IP này là chặt chẽ hơn. Tuy nhiên, nhược điểm của nó là tốn kém phải đầu tư hệ thống xử lý tốc độ cao, đủ để lọc tất cả các gói tin gửi ra Internet.

1.2. Tìm hiểu về firewall

1.2.1. Phân loại:

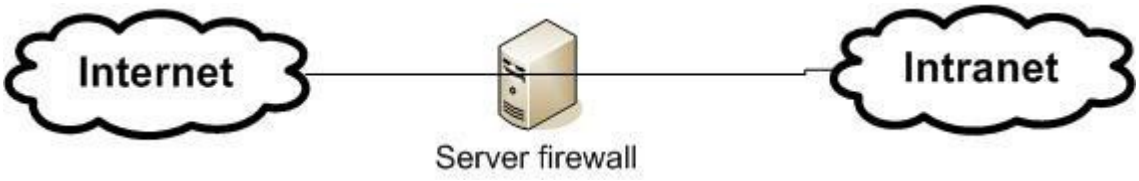
Firewall được chia làm 2 loại, gồm Firewall cứng và Firewall mềm:

1.2.1.1. Firewall cứng: Là những firewall được tích hợp trên Router.



+ Đặc điểm của Firewall cứng:
- Không được linh hoạt như Firewall mềm
- Firewall cứng hoạt động ở tầng thấp: tầng Network và tầng Transport
- Firewall cứng không thể kiểm tra được nội dung của gói tin. Ví dụ Firewall cứng: NAT (Network Address Translate).

1.2.1.2. Firewall mềm: Là những Firewall được cài đặt trên Server.



+ Đặc điểm của Firewall mềm:
- Tính linh hoạt cao: Có thể thêm, bớt các quy tắc chức năng.
- Firewall mềm hoạt động ở tầng cao hơn Firewall cứng (tầng ứng dụng)
- Firewal mềm có thể kiểm tra được nội dung của gói tin (thông qua các từ khóa). Ví dụ về Firewall mềm: Zone Alarm, Norton Firewall…

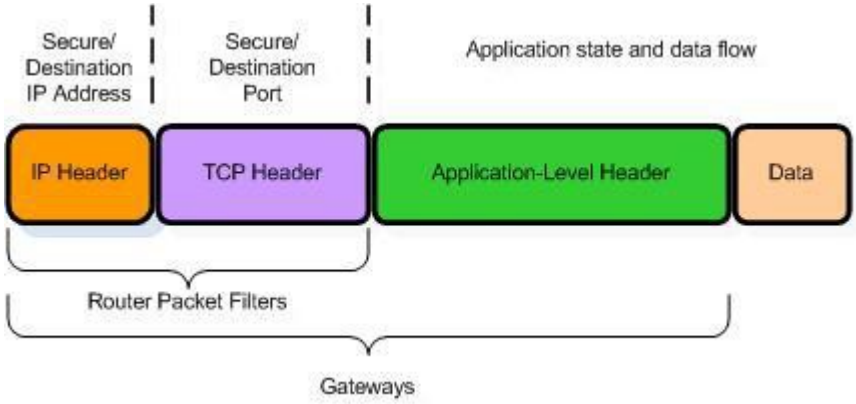
1.2.2.Thành phần của firewall

Một FireWall bao gồm một hay nhiều thành phần sau :

+ Bộ lọc packet (packet- filtering router).

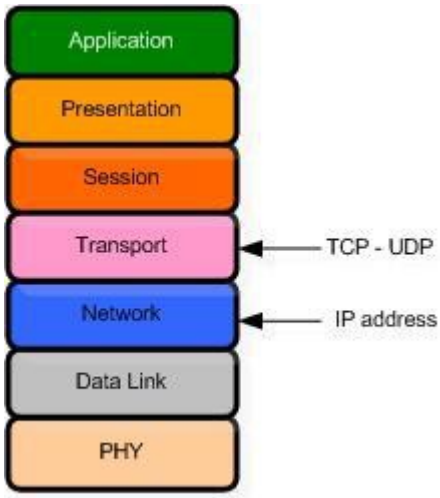
+ Cổng ứng dụng (Application-level gateway hay proxy server).

+ Cổng mạch (Circuite level gateway).



1.2.2.1. Bộ lọc paket (Paket filtering router):

**Nguyên lý hoạt động*



Khi nói đến việc lưu thông dữ liệu giữa các mạng với nhau thông qua Firewall thì có nghĩa rằng Firewall hoạt động chặt chẽ với giao thức TCI/IP. Vì giao thức này làm việc theo thuật toán chia nhỏ các dữ liệu nhận được từ các ứng dụng trên mạng. Bộ lọc packet cho phép hay từ chối mỗi packet mà nó nhận được. Nó kiểm tra toàn bộ đoạn dữ liệu để quyết định xem đoạn dữ liệu đó có thoả mãn một trong số các luật lệ của lọc packet hay không. Các luật lệ lọc packet dựa trên các thông tin ở đầu mỗi packet (packet header), dùng để cho phép truyền các packet đó trên mạng. Đó là:

.Địa chỉ IP nơi xuất phát (IP Source address)

.Địa chỉ IP nơi nhận (IP Destination address)

.Những thủ tục truyền tin (TCP, UDP, ICMP, IP tunnel)

.Cổng TCP/UDP nơi xuất phát (TCP/UDP source port)

.Cổng TCP/UDP nơi nhận (TCP/UDP destination port)
.Dạng thông báo ICMP (ICMP message type)
.Giao diện packet đến (incomming interface of packet)
.Giao diện packet đi (outcomming interface of packet)
Nếu luật lệ lọc packet được thoả mãn thì packet được chuyển qua firewall. Nếu không packet sẽ bị bỏ đi. Nhờ vậy mà Firewall có thể ngăn cản được các kết nối vào các máy chủ hoặc mạng nào đó được xác định, hoặc khóa việc truy cập vào hệ thống mạng nội bộ từ những địa chỉ không cho phép. Hơn nữa, việc kiểm soát các cổng làm cho Firewall có khả năng chỉ cho phép một số loại kết nối nhất định vào các loại máy chủ, hoặc chỉ có những dịch vụ nào đó (Telnet, SMTP, FTP...) được phép chạy trên hệ thống mạng cục bộ.

**Ưu điểm*

Đa số các hệ thống firewall đều sử dụng bộ lọc packet. Một trong những ưu điểm của phương pháp là chi phí thấp vì cơ chế lọc packet đó được bao gồm trong phần mềm router.

**Hạn chế*

Khá phức tạp, đòi hỏi người quản trị mạng cần có hiểu biết chi tiết về các dịch vụ Internet, các dạng packet header, và các giá trị cụ thể. Khi đòi hỏi về sự lọc càng lớn, các luật lệ lọc càng trở nên dài và phức tạp, rất khó quản lý và điều khiển.

1.2.2.2. Cổng ứng dụng (application-level getway):

**Nguyên lý hoạt động.*

Đây là một loại Firewall được thiết kế để tăng cường chức năng kiểm soát các loại dịch vụ, giao thức được cho phép truy cập vào hệ thống mạng. Cơ chế hoạt động của nó dựa trên cách thức gọi là Proxy service. Proxy service là các bộ code đặc biệt cài đặt trên gateway cho từng ứng dụng. Nếu người quản trị mạng không cài đặt proxy code cho một ứng dụng nào đó, dịch vụ tương ứng sẽ không được cung cấp và do đó không thể chuyển thông tin qua firewall. Ngoài ra, proxy code có thể được định cấu hình để hỗ trợ một số đặc điểm trong ứng dụng mà người quản trị mạng cho là chấp nhận được. Một

cổng ứng dụng thường được coi như là một pháo đài(Bastion host). Những biện pháp đảm bảo an ninh của pháo đài là:

- Bastion host luôn chạy các version an toàn (secure version
- Chỉ những dịch vụ mà người quản trị mạng cho là cần thiết mới được cài đặt trên bastion host
- Bastion host có thể yêu cầu nhiều mức độ xác thực khác nhau
- Mỗi proxy được đặt cấu hình để cho phép truy nhập chỉ một số máy chủ nhất định
- Mỗi proxy duy trì một quyển nhật ký ghi chép lại toàn bộ chi tiết của giao thông qua nó, như sự kết nối, khoảng thời gian kết nối, nhật ký này rất có ích trong việc tìm theo dấu vết hay ngăn chặn kẻ phá hoại. - Mỗi proxy đều độc lập với các proxies khác trên bastion host. Điều này cho phép dễ dàng quá trình cài đặt một proxy mới, hay tháo gỡ một proxy đang có vấn đề

**Ưu điểm*

- Cho phép người quản trị hoàn toàn điều khiển được từng dịch vụ trên mạng
- Cho phép người quản trị hoàn toàn điều khiển được những dịch vụ nào cho phép
- Cổng ứng dụng cho phép kiểm tra độ xác thực rất tốt, và nó có nhật ký ghi chép lại thông tin về truy nhập hệ thống.
- Luật lệ lọc filltering cho cổng ứng dụng là dễ dàng cấu hình và kiểm tra hơn so với bộ lọc packet.

**Hạn chế*

Yêu cầu các users thay đổi thao tác, hoặc thay đổi phần mềm cài đặt trên máy khách cho truy nhập vào các dịch vụ proxy. Chẳng hạn, Telnet truy nhập qua cổng ứng dụng đòi hỏi hai bước để nối với máy chủ chứ không phải là một bước. Tuy nhiên, cũng đã có một số phần mềm client cho phép ứng dụng trên cổng ứng dụng là trong suốt, bằng cách cho phép user chỉ ra máy đích chứ không phải cổng ứng dụng trên lệnh Telnet.

1.2.2.3. Cổng vòng (circuit-Level Gateway)

Cổng vòng là một chức năng đặc biệt có thể thực hiện được bởi một cổng ứng dụng. Cổng vòng đơn giản chỉ chuyển tiếp (relay) các kết nối TCP mà không thực hiện bất kỳ một hành động xử lý hay lọc packet nào. Cổng vòng đơn giản chuyển tiếp kết nối telnet qua firewall mà không thực hiện một sự kiểm tra, lọc hay điều khiển các thủ tục Telnet nào.Cổng vòng làm việc như một sợi dây,sao chép các byte giữa kết nối bên trong (inside connection) và các kết nối bên ngoài (outside connection). Tuy nhiên, vì sự kết nối này xuất hiện từ hệ thống firewall, nó che dấu thông tin về mạng nội bộ. Cổng vòng thường được sử dụng cho những kết nối ra ngoài, nơi mà các quản trị mạng thật sự tin tưởng những người dùng bên trong.

Ưu điểm lớn nhất là bastion host có thể được cấu hình như là một hỗn hợp cung cấp Cổng ứng dụng cho những kết nối đến, và cổng vòng cho các kết nối đi. Điều này làm cho hệ thống bức tường lửa dễ dàng sử dụng cho những người trong mạng nội bộ muốn trực tiếp truy nhập tới các dịch vụ Internet, trong khi vẫn cung cấp chức năng bức tường lửa để bảo vệ mạng nội bộ từ những sự tấn công bên ngoài.

1.2.3. Chức năng

-Chức năng chính của Firewall là kiểm soát luồng thông tin giữa Intranet và Internet

- Nó còn là hàng rào chắn đầu tiên chống lại những kẻ chuyên rình mò trên Internet

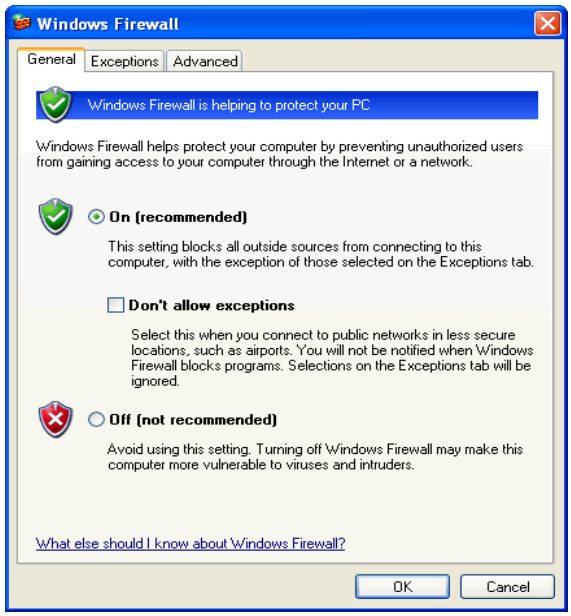
-Bên cạnh việc chống lại những tên trộm chương trình và người dùng bất hợp pháp, firewall còn có tác dụng trong việc ngăn chặn virus tin học

-Ngoài ra, firewall còn giúp chống các loại virus bằng cách hạn chế truy cập tới người dùng không hợp pháp.

1.3. Khai thác Firewall Windown của windown

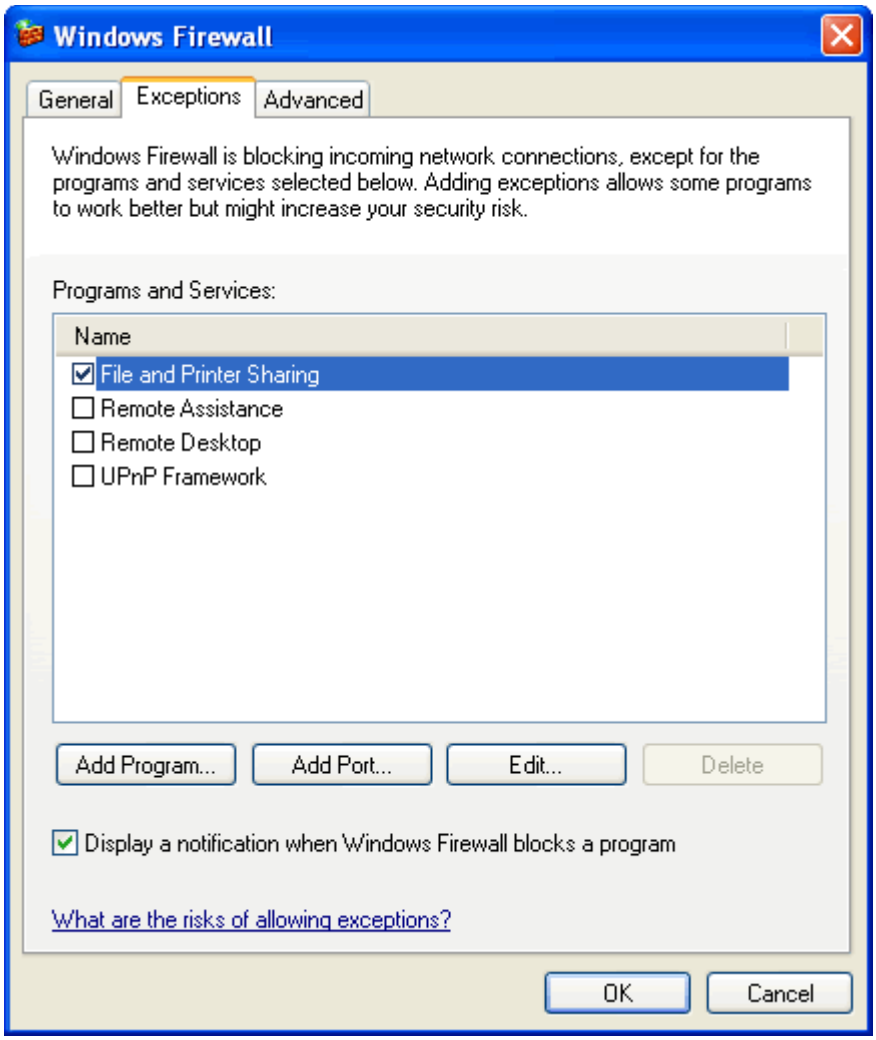
1.3.1. Một số giao diện cơ bản

1.3.1.1. Giao diện window firewall của window xp:



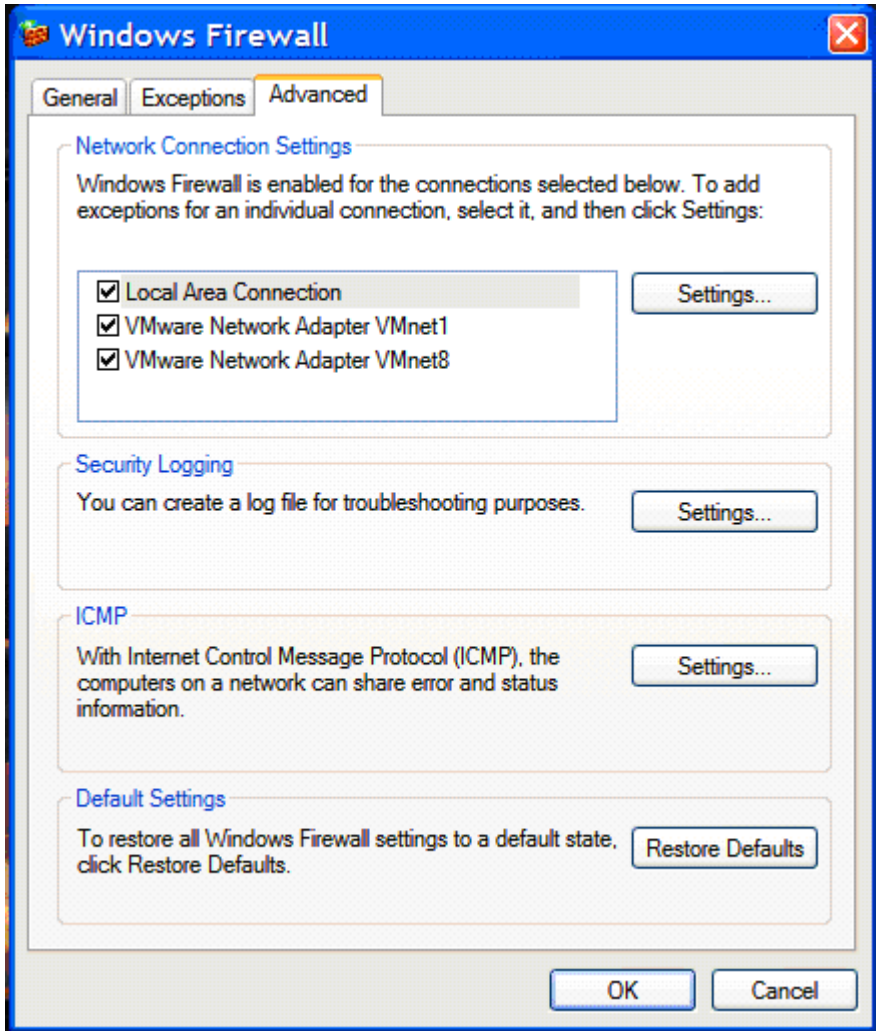
Hình 2.

ở cửa sổ General này cho phép chúng ta bật (on) hay tắt (off) tường lửa



Hình 3.

ở cửa sổ Exeptiopns cho phép chúng ta thêm, chỉnh sữa các chương trình, các cổng ngoại lệ. Tức là các cổng các chương trình này không bị windown firewall ngăn chặn khi vào ra máy tính của bạn.

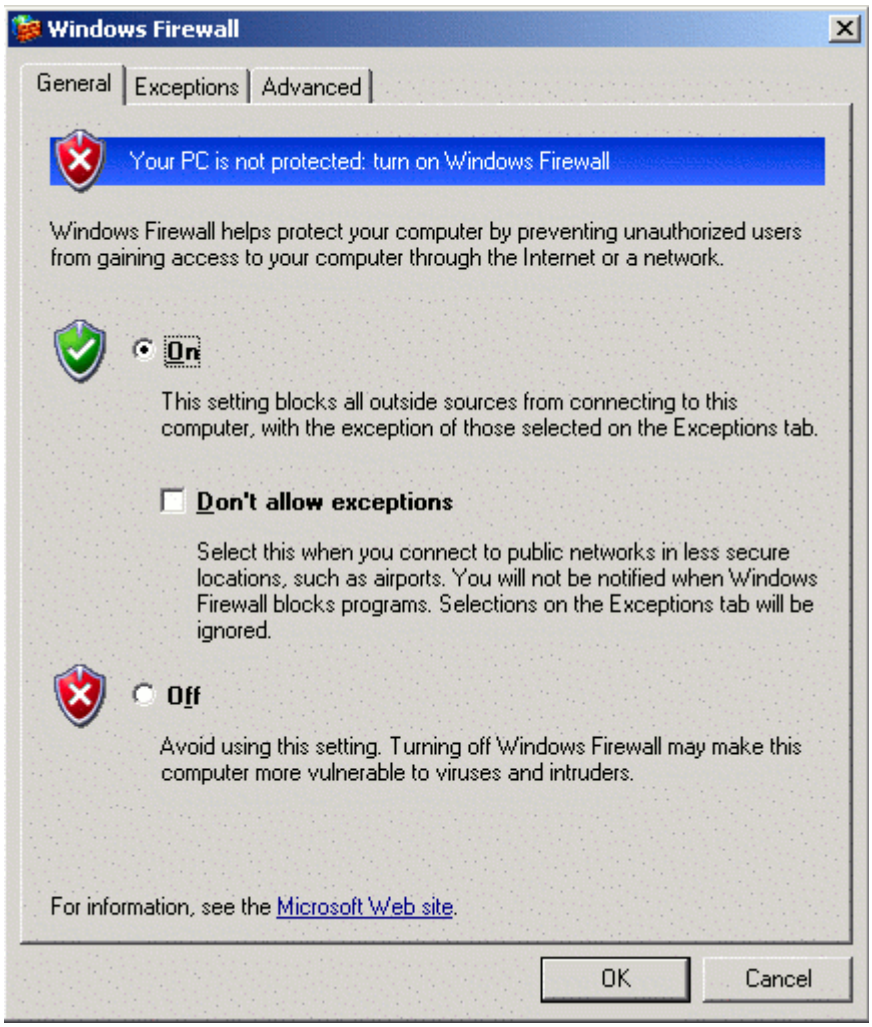


Hình 4.

Với cữa sổ Advanced chúng ta có thể thiết lập các kết nối internet nào được bảo vệ bởi Windown firewall

1.3.1.2. Giao diện của window firewall của window server.

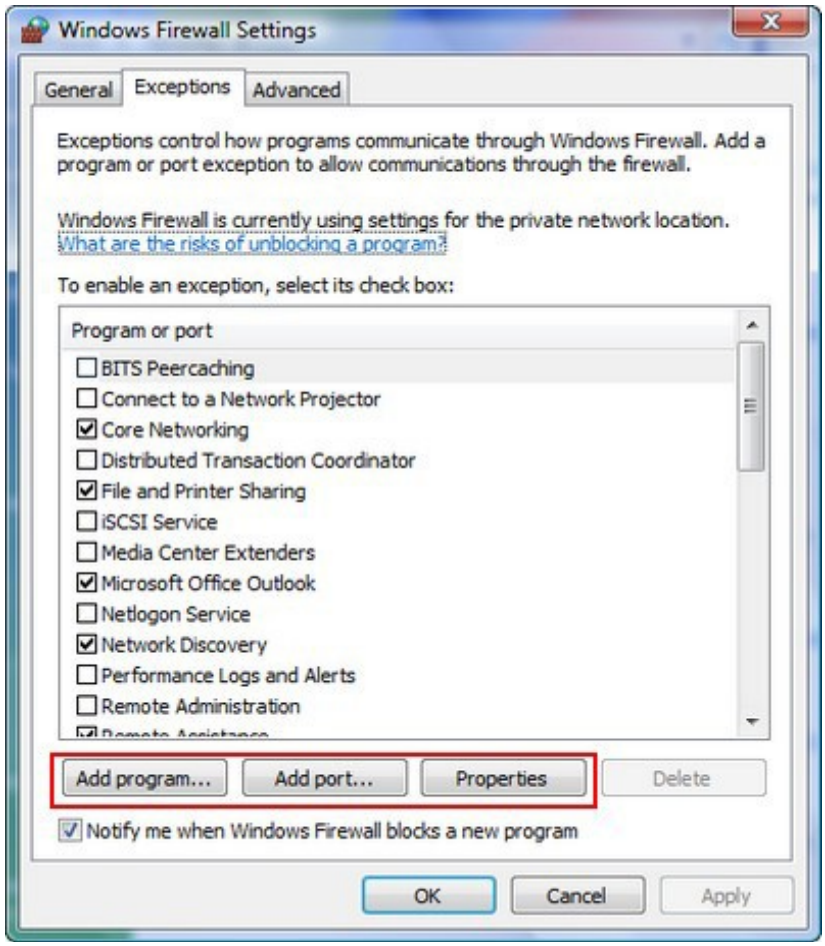
Cũng giống như firewall của windown xp, firewall của windown server cũng có 3 dịch vụ cơ bản là: Exceptions,Advanced,General.



Hình 5.

1.3.1.3. Giao diện của window firewall của window vista:

Do Windown vista có nhiều tính năng và cấu hình nỗi trội hơn so với các windown khác nên firewall của nó cũng có nhiều tính năng và dịch vụ mới giúp bảo vệ máy tính tốt hơn



Hình 6.

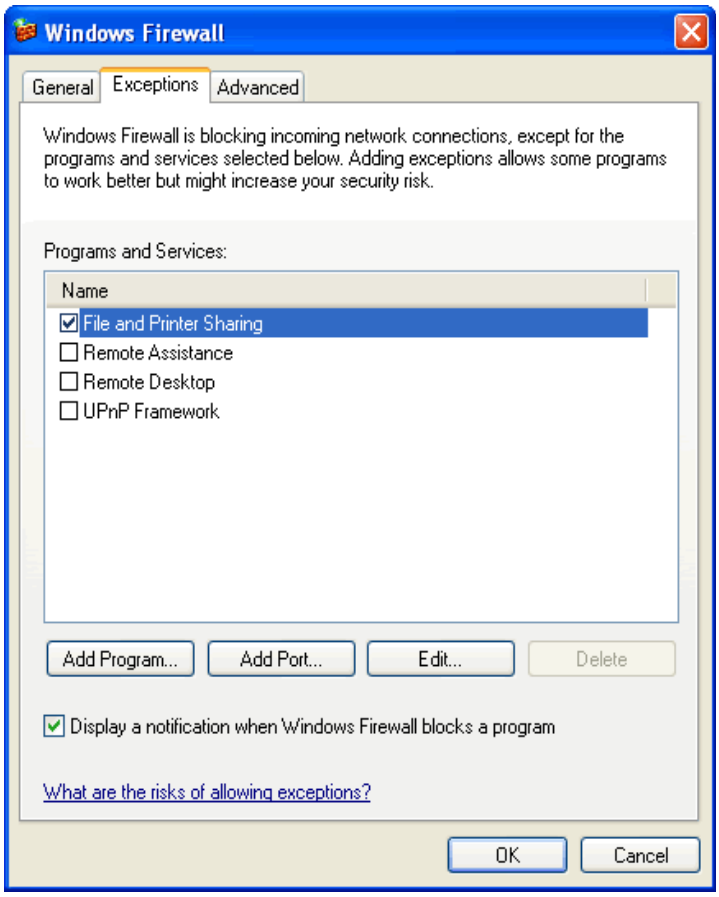
1.3.2. Khai thác dịch vụ firewall của windown:

1.3.2.1.General:

ở cửa sổ Ganeral cho phép chúng ta bật(on) hay tắt(off) tường lửa. Nếu bật và chọn Don't allow Exception thì những dịch vụ ngoại lệ sẽ không còn, tất cả những liên kết sẽ phải qua tường lửa và bị tường lửa kiểm soát.

1.3.2.2.Exceptions:

ở cửa sổ Exceptions chứa những liên kết ngoại lệ tức là nếu chọn vào những lựa chọn này thì những liên kết này sẽ không thông qua tường lửa khi giao tiếp với bên ngoài,và không bị tường lửa chặn.



Hình 8.

Một số lựa chọn :

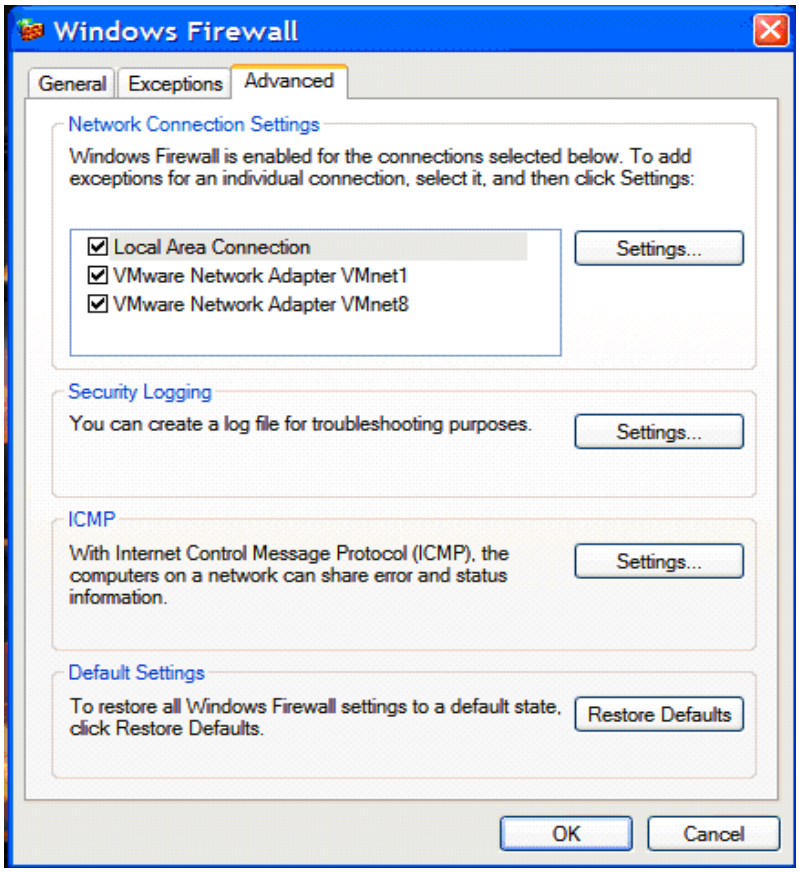
File and printer sharing : Chia sẻ tệp và máy in.

Remote assistance : Hổ trợ từ xa.

Remote desktop : Truy cập từ xa.

Upnp Framework : Khung Upnp

1.3.2.3.Advanced:

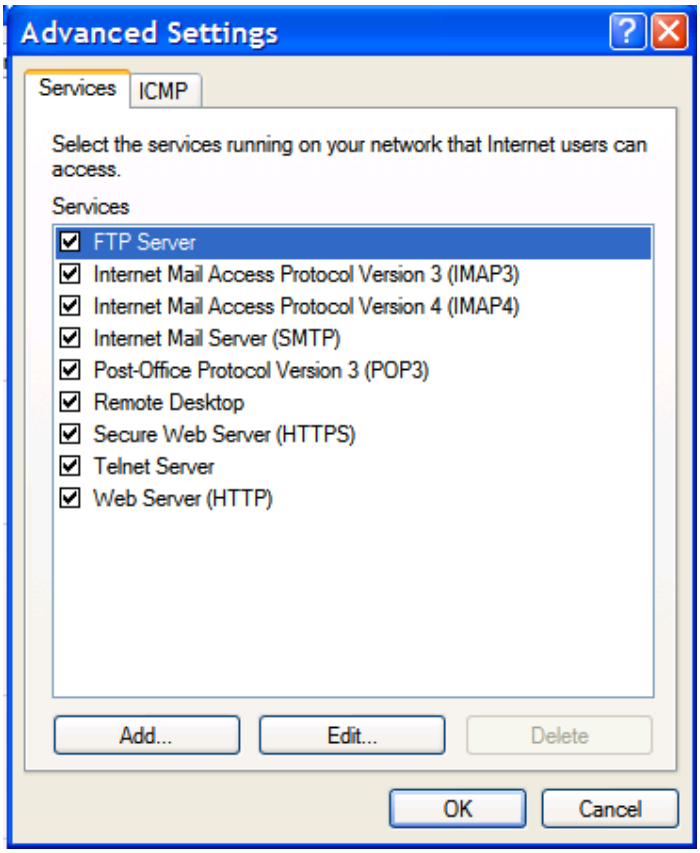


Hình 9.

Trong cửa sổ này cho ta 4 lựa chọn để thiết lập các liên kết Internet được bảo vệ bởi window firewall:

- Network connection settings: Thiết lập liên kết mạng.
- Security logging : An ninh đăng nhập.
- ICMP(internet control message protocol) :kiểm soát giao thức internet.
- Default settings : Thiết lập ngầm định.

*** Networkconnection settings: Thiết lập liên kết mạng.**



Hình 11.

*** Service (Dịch vụ)**

Ở lựa chọn này cho phép ta lựa chọn các dich vụ đang chạy trên mạng của bạn mà người dụng Internet có thể truy cập được, có những lựa chọn sau:

- FTP Server (File transfer protocol): dịch vụ chuyển file cho phép file được truyền, sao chép từ một hệ thống máy này đến hệ thống máy khác

- Internet mail access protocol version 3 (IMAP3): giao thức truy cập thư tín Internet cho phép quản lý thư từ ngoài gửi vào và từ trong gửi đi.

- Internet mail access protocol version 4 (IMAP4) : giống như IMAP3.

- Internet mail server :dịch vụ mail internet.

- Pos – office protocol version 3 (pop3) giao thức bưu điện tức là hoạt động để quản lý lưu lượng thư khi vào ra máy tính.

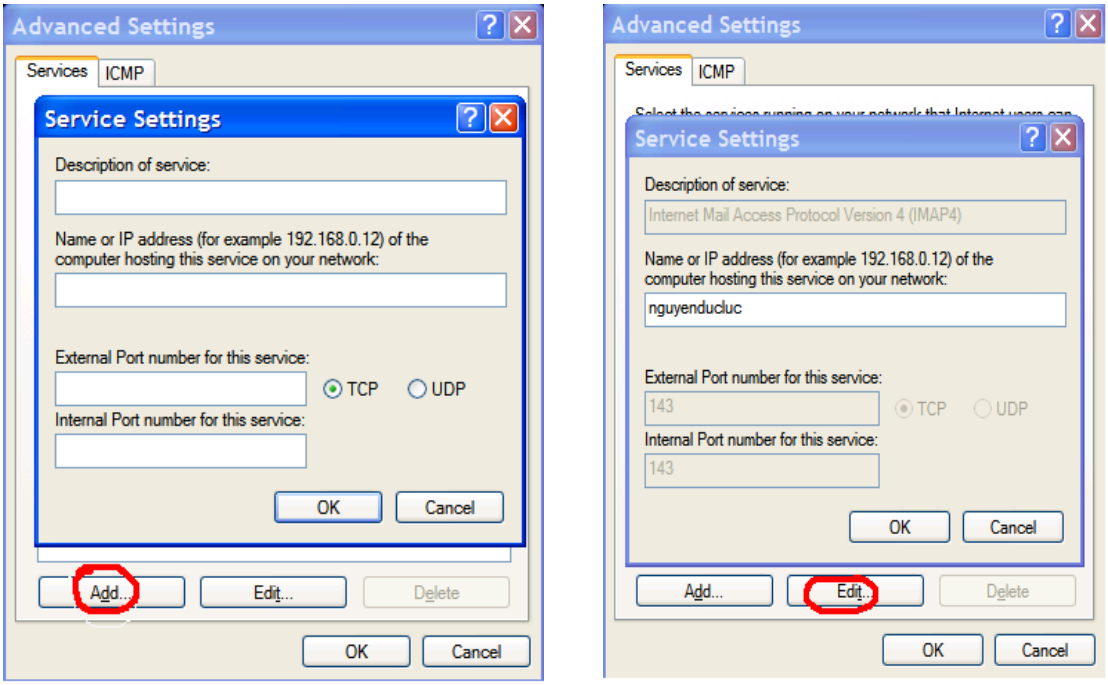
- Remote Desktop: cho phép truy cập, điều khiển máy tính từ xa

- Secure web server : Dịch vụ bảo mật web.

- Telnet server : Tạo ra một liên kết từ xa giữa máy tính của bạn với các thiết bị có trong mạng.

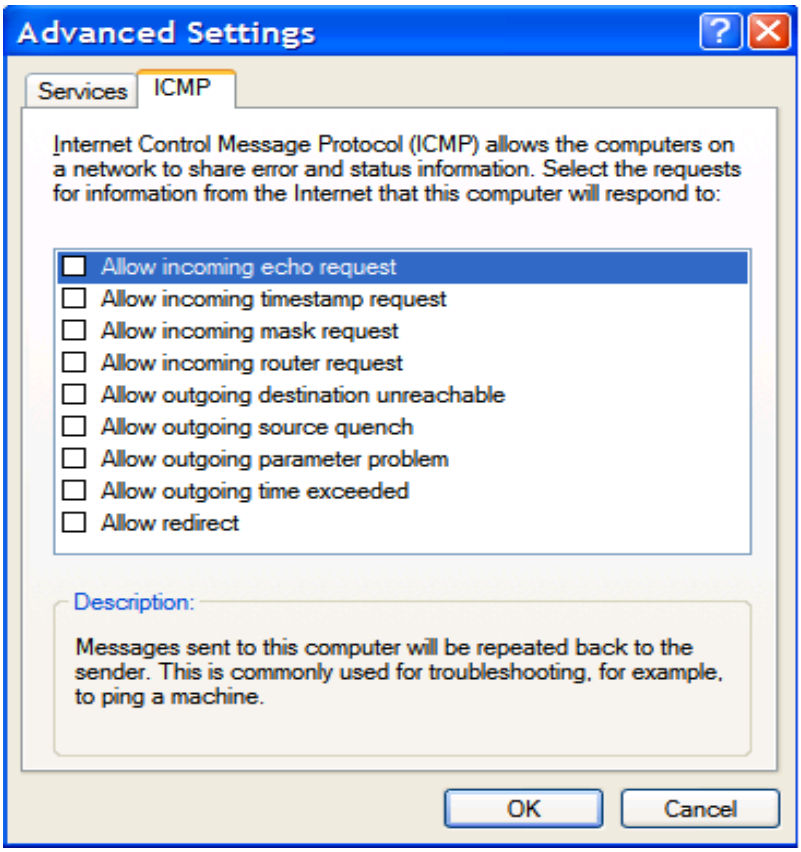
- Web server : Dịch vụ web kiểm soát hoạt động của máy khi truy cập web.

Ngoài ra chúng ta có thể thêm các kết nối bằng cách chọn Add,chỉnh sữa các lựa chọn kết nối bằng cách chọn Edit.



Hình 11.

*** ICMP (Internet control messasge protocol)**



Hình 12.

Kiểm soát giao thức internet cho phép các tin nhắn trên một máy tính, một mạng lưới chia sẻ thông tin và tình trạng lỗi, lựa chọn những yêu cầu về thông tin từ internet rằng máy tính này sẽ trả lời cho phép những dịch vụ nào được đi qua. Dưới đây là một số lựa chọn:

- Allow incoming echo request: Cho phép các yêu cầu phản hồi lại thông tin.

- Allow incoming timestamp request :Cho phép các yêu cầu về thời gian.

- Allow incoming mask request: Cho phép các yêu cầu về mặt nạ (mask).

- Allow incoming router request: Cho phép các yêu cầu định tuyến.

- Allow outgoing destination unreachable: Cho phép gửi thư tới unreachable (không thể đến,không thể tới).

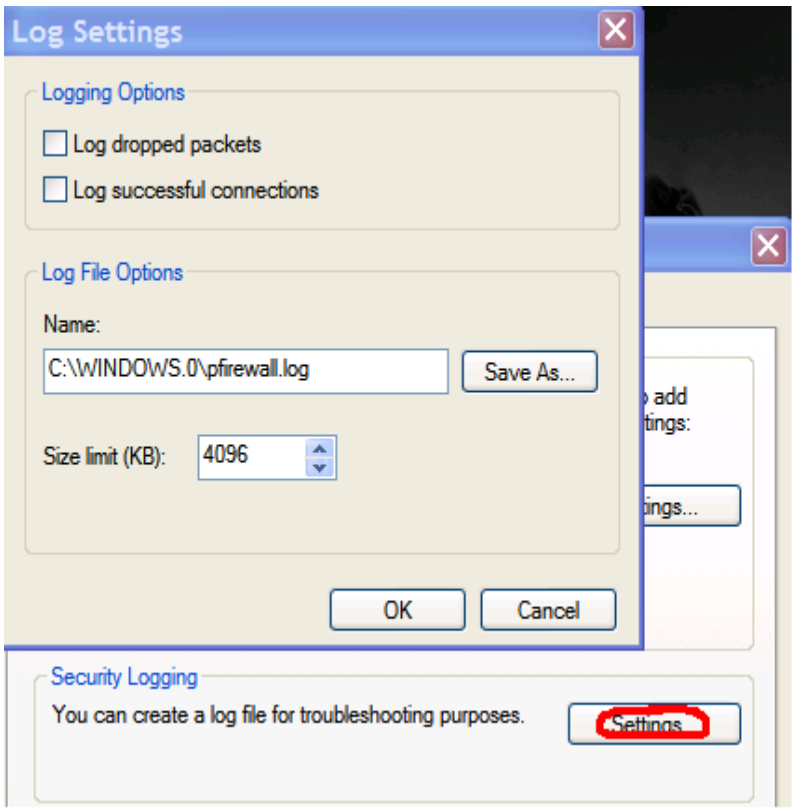
- Allow outgoing source quench: Gửi thư cho phép mỡ nguồn nhúng (quench).

- Allow outgoing parameter problem: Gửi thư cho phép tham số vấn đề.

- Allow outgoing time exceeded: Gửi thư cho phép vượt thời gian .

- Allow redirect: Cho phép chuyển hướng.
- Allow outgoing pasket too big: Cho phép gửi thư gói quá lớn.

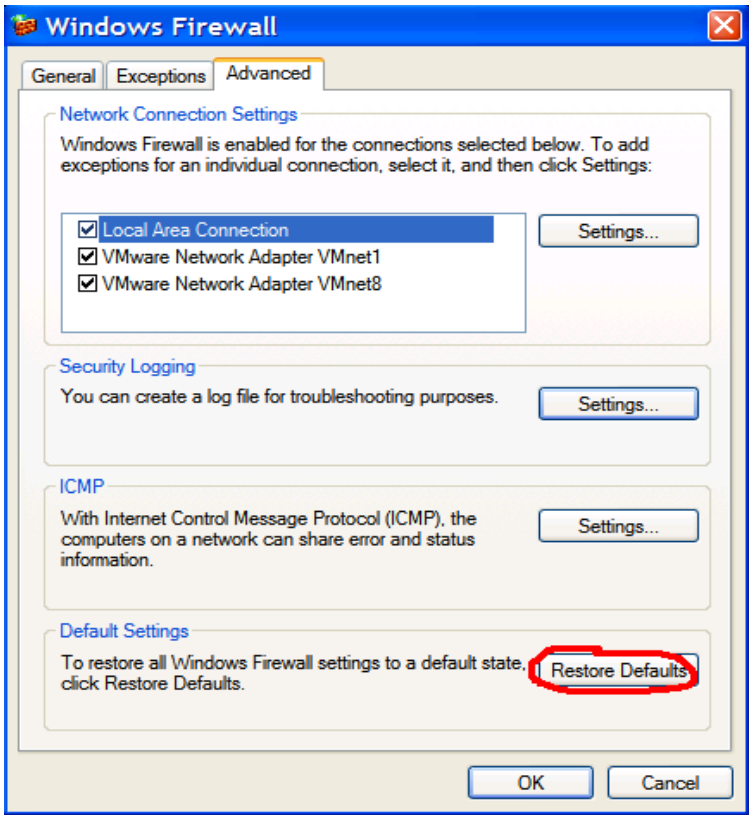
*** Security logging : An ning đăng nhập.**



Hình 13

Bạn có thể tạo một tập tin đăng nhập cho mục đích xử lý sự cố.

*** Default settings. Thiết lập ngầm định.**



Hình 14.

Khi ta chọn **restore Defaults** thì nó sẻ khôi phục lại tất cả các cửa sổ cài đặt bức tường lửa về trạng thái ngầm định

1.4. Khai thác dịch vụ firewall của windown service:

Ngoài những chức năng của firewall windown thì firewall windown service còn có những ưu thế hơn như:

• Windows Firewall : Protect all network connections: thiết lập này buộc tường lửa tắt hay mở cho một định danh

• Windows Firewall : Do not allow exceptions: Tựy chọn chỉ thị cho tường lửa từ chối các trường hợp đặc biệt đó được chỉ định. Kích hoạt thiết lập này tương đương với việc chọn " Don't allow exceptions " (Không cho phép các trường hợp đặc biệt) trên thẻ General trong Windows Firewall Control Panel.

• Windows Firewall: Define program exceptions Properties: Thiết lập cho phép bạn tuỳ chọn chỉ định các chương trình, giúp bạn cấp phép cho các trường hợp đặc biệt "tấm vé" để qua tường lửa.

• Windows Firewall: Prohibit notifications: Thiết lập dừng các thông báo của tường lửa khi một chương trình yêu cầu Windows Firewall bổ sung nó vào danh sách các chương trình cho phép.
• Windows Firewall: Allow logging: Tuỳ chọn cho phép bạn cấu hình cấp bậc bản ghi lưu trữ thông tin cho tường lửa, kích cỡ bản ghi, tên và vị trí…

Ngoài ra còn firewall còn có tính năng bảo mật nâng cao được tích hợp Windows Server 2008 Group Policy, vì vậy sẽ cho phép bạn có thể sử dụng giao diện điều khiển của Group Policy và Group Policy Editor để tạo chính sách tường lửa cho các máy tính trong toàn bộ miền, trong một OU hoặc trong một site.

PHÂN IV: NÉN DỮ LIỆU

4.1 Đề bài toán:

Tìm hiểu và cài đặt thuật toán nén và giải nén dữ liệu RLE (Run Length Code) cho một tệp dữ liệu.

4.2 Tìm hiểu chung:

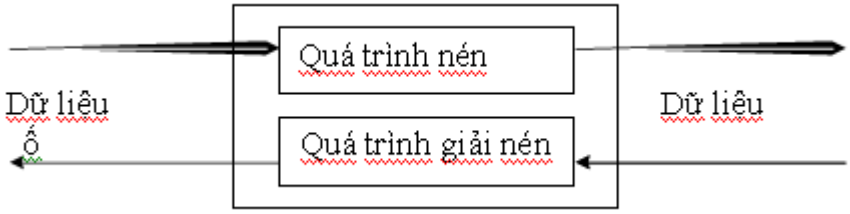
4.2.1 Nén dữ liệu:

Dữ liệu gốc qua bộ mã hóa, bộ mã hóa này làm giảm dung lượng thông tin "dư thừa" trong dữ liệu gốc và làm cho lượng thông tin thu được sau khi nén thường nhỏ hơn dữ liệu gốc rất nhiều. Mục đích là tiết kiệm được bộ nhớ, giảm đi những chi phí trong việc lưu trữ dữ liệu và chi phí thời gian để truyền dữ liệu đi xa trong truyền thông nhưng vẫn cho phép chúng ta khôi phục lại dữ liệu ban đầu mà đảm bảo được chất lượng của dữ liệu.

4.2.2 Giải nén dữ liệu:

Dữ liệu nén đi qua bộ giải mã dữ liệu, bộ giải mã sẽ thực hiện giải nén để thu được dữ liệu gốc ban đầu. Việc giải nén này thường dựa vào các thông tin đi kèm theo dữ liệu nén, tùy thuộc vào kiểu nén hay phương pháp nén mà dữ liệu giải nén được có hoàn toàn giống với dữ liệu gốc ban đầu hay không.

Ta có sơ đồ nén và giải nén sau:



4.2.3 Mục đích:

Thu nhỏ dũ liệu cần thiết tiết kiệm tài nguyên lưu trữ và tiết kiệm thời gian, thông lượng đường truyền khi cần truyền dũ liệu trên mạng WAN, LAN. Ngoài ra việc nén dữ liệu cũng là một cách mã hoá dữ liệu nhằm mục đích bảo mật dữ liệu.

4.2.4 Phương thức:

Khi nói đến phương thức nén dữ liệu ta cần hiểu là có hai công nghệ hoặc là hai loại thuật toán nén dữ liệu là nén mất mát thông tin và nén không mất mát thông tin.

4.2.4.1 Nén không mất mát thông tin:

Khi ta có một dữ liệu gọi là X và nhờ thuật toán nén không mất mát thông tin ta thu được dữ liệu Xc và sau đó nhờ việc giải nén dữ liêu Xc ta thu được Y và dữ liệu X và dữ liệu Y là giống nhau hoàn toàn. Những thuật toán nén dạng này như RLE, Huffman, Lempel-Ziv ... Những dữ liệu dạng văn bản, ký tụ thường sử dụng công nghệ nén này. Ví dụ như giá trị 65 là trong ASSCI ‘A’ khi sau khi nén và giải nén nếu sử dụng công nghệ mất mát thông tin sai lệch mã 65 thành 66 dẫn đến ‘B’ do vậy không thể chấp nhận được.

4.2.4.2 Nén mất mát thông tin:

Còn khi ta sử dụng công nghệ nén dữ liệu có mất mát thông tin thì X và Y theo ví dụ trên là không giống nhau hoàn toàn và việc chuẩn xác về dữ liệu sau khi giải nén với dữ liệu ban đầu có thê tin cậy được còn tuy thuộc vào thuật toán mà chúng ta sử dung và kích thước dữ liệu dùng để thao tác sử lý. Những thuật toán này thường sử dụng nén tín hiệu (tệp dữ liệu dạng nhị phân như dữ liệu tệp ảnh, video, audio).

Có rất nhiều phương pháp nén và giải nén đã ra đời và không ngừng cải tiến để ngày càng hoàn thiện, đem lại hiệu quả nén cao và chất lượng giải nén tốt nhất. Trong đề tài này em nghiên cứu về phương pháp nén và giải nén

mã loạt dài Run Length Coding (RLE). Qua đề tài này sẽ cho biết hiệu quả của việc nén và giải nén như thế nào.

4.3. Phương pháp mã hóa độ dài loạt RLE:

Mã hoá theo độ dài loạt RLE (Run Length Encoding) là một phương pháp nén dữ liệu dựa trên sự cắt bớt các dư thừa về không gian. Loạt ở đây được định nghĩa là dãy các phần tử dữ liệu liên tiếp có cùng chung một giá trị.

Nguyên tắc:

Nguyên tắc của phương pháp này là phát hiện một loạt các phần tử của dữ liệu lặp đi lặp lại liên tiếp, ví dụ:110000000000000011 .Ta thấy phần tử có giá trị 0 xuất hiện nhiều lần liên tiếp thay vì phải lưu trữ toàn bộ các phần tử có giá trị 0 ta chỉ cần lưu trữ chúng bằng cách sử dụng các cặp (độ dài loạt, giá trị).

Loại dư thừa đơn giản nhất trong một tập tin là các đường chạy dài gồm các kí tự lặp lại, điều này thường thấy trong các tập tin đồ hoạ bitmap, các vùng dữ liệu hằng của các tập tin chương trình, một số tập tin văn bản...

Ví dụ, xét chuỗi sau:

AAAABBBAABBBBBCCCCCCCCDABCBAAABBBBCCCD

Chuỗi này có thể được mã hoá một cách cô đọng hơn bằng cách thay thế chuỗi kí tự lặp lại bằng một thể hiện duy nhất của kí tự lặp lại cùng với một biến đếm số lần kí tự đó được lặp lại. Ta muốn nói rằng chuỗi này gồm bốn chữ A theo sau bởi ba chữ B rồi lại theo sau bởi hai chữ A, rồi lại theo sau bởi năm chữ B... Việc nén một chuỗi theo phương pháp này được gọi là mã hoá độ dài loạt. Khi có những loạt dài, việc tiết kiệm có thể là đáng kể. Có nhiều cách để thực hiện ý tưởng này, tuỳ thuộc vào các đặc trưng của ứng dụng (các loạt chạy có khuynh hướng tương đối dài hay không ? Có bao nhiêu bit được dùng để mã hoá các kí tự đang được mã ?).

Nếu ta biết rằng chuỗi của chúng ta chỉ chứa các chữ cái, thì ta có thể mã hoá biến đếm một cách đơn giản bằng cách xen kẻ các con số với các chữ cái. Vì vậy chuỗi kí tự trên được mã hoá lại như sau:

#A4#B3#A2#B5#C8#D1#A1#B1#C1#B1#A3#B4#C3#D1

Ở đây "#A4" có nghĩa là "bốn chữ A"... Chú ý là không đáng để mã hoá các loạt chạy có độ dài 1 hoặc 2 vì cần đến hai kí tự để mã hoá.

Đối với các tập tin nhị phân một phiên bản được tinh chế của phương pháp này được dùng để thu được sự tiết kiệm đáng kể. Ý tưởng ở đây là lưu lại các độ dài loạt, tận dụng sự kiện các loạt chạy thay đổi giữa 0 và 1 để tránh phải lưu chính các số 0 và 1 đó. Điều này giả định rằng có một vài loạt chạy ngắn (Ta tiết kiệm các bit trên một loạt chạy chỉ khi độ dài của đường chạy là lớn hơn số bit cần để biểu diễn chính nó trong dạng nhị phân), nhưng khó có phương pháp mã hoá độ dài loạt nào hoạt động thật tốt trừ phi hầu hết các loạt chạy đều dài.

Việc mã hoá độ dài loạt cần đến các biểu diễn riêng biệt cho tập tin và cho bản đã được mã hoá của nó, vì vậy nó không thể dùng cho mọi tập tin, điều này có thể hoàn toàn bất lợi, ví dụ, phương pháp nén tập tin kí tự đã được đề nghị ở trên sẽ không dùng được đối với các chuỗi kí tự có chứa số. Nếu những kí tự khác được sử dụng để mã hoá các số đếm, thì nó sẽ không làm việc với các chuỗi chứa các kí tự đó. Giả sử ta phải mã hoá bất kì kí tự nào từ một bảng chữ cái cố định bằng cách chỉ dùng các kí tự từ bảng chữ cái đó. Để minh hoạ, giả sử ta phải mã hoá bất kì một chuỗi nào từ một chữ cái đó, ta sẽ giả định rằng ta chỉ có 26 chữ cái trong bảng chữ cái (và cả khoảng trống) để làm việc.

Để có thể dùng vài chữ cái để biểu diễn các số và các kí tự khác biểu diễn các phần tử của chuỗi sẽ được mã hoá, ta phải chọn một kí tự được gọi là kí tự "Escape". Mỗi một sự xuất hiện của kí tự đó báo hiệu rằng hai chữ cái tiếp theo sẽ tạo thành một cặp (số đếm, kí tự) với các số đếm được biểu diễn bằng cách dùng kí tự thứ i của bảng chữ cái để biểu diễn số i. Vì vậy, chuỗi ví dụ của chúng ta sẽ được biểu diễn như sau với Q được xem là các kí tự "Escape"

QDABBBAABQHCDABCBAAAQDBCCCD

Tổ hợp của kí tự "Escape", số đếm và một kí tự lặp lại được gọi là một dãy Escape. Chú ý rằng không đáng để mã hoá các đường chạy có chiều dài ít

hơn bốn kí tự, vì ít nhất là cần đến ba kí tự để mã hoá bất kì một loạt chạy nào.

Trong trường hợp bản thân kí tự "Escape" xuất hiện trong dãy kí tự cần mã hoá ta sử dụng một dãy "Escape" với số đếm là 0 (kí tự space) để biểu diễn kí tự "Escape". Như vậy trong trường hợp kí tự "Escape" xuất hiện nhiều thì có thể làm cho tập tin nén phình to hơn trước.

Các loạt chạy dài có thể được cắt ra để mã hoá bằng nhiều dãy Escape, ví dụ, một loạt chạy gồm 51 chữ A sẽ được mã hoá như QZAQYA bằng cách dùng trên.
Phương pháp mã hoá độ dài loạt thường được áp dụng cho các tập tin đồ hoạ bitmap vì ở đó thường có các mảng lớn cùng màu được biểu diễn dưới dạng bitmap là các chuỗi bit có đường chạy dài. Trên thực tế, nó được dùng trong các tập tin .PCX, .RLE.

4.4 Mô tả thuật toán:

4.4.1 Thuật toán giải nén:

```
Lấy 2 byte
While ( phẩn tử mảng cuối cùng)
{
    if( nếu 2 byte này có giá trị bằng nhau)
    {
        Đếm có bao nhiêu byte liền kề sau đó có cùng giá trị
        Đẩy 2 byte trên ra mảng kết quả
        Đẩy giá trị đếm được vào byte tiếp theo ra mảng kết quả
        Cập nhật trở mảng nguồn
        Lấy 2 byte tiếp theo
    }
    Else
    {
        Đẩy byte trước vào mảng kết quả
        Biến lưu byte trước giá trị gần byte sau đó
        Lấy tiếp 1 byte nữa làm byte liền sau đó
```

```
                Cật nhật trỏ mảng nguồn
            }
        }
```

4.4.2 Thuật toán giải nén:

Lấy 1 byte, đẩy byte vừa lấy ra vào mảng kết quả. Lấy tiếp 1 byte kế sau.

```
While (phần tử mảng cuối cùng)
{
    If(2 byte này có giá trị bằng nhau)
    {
        Lấy 1 byte kế tiếp, nó là giá trị đếm phần tử trùng "đếm"
        Đẩy byte "kế sau" vào mảng kết quả
        Đẩy đếm byte tiếp theo vào mảng kết quả giá trị "kế sau"
        Cập nhật trỏ mảng nguồn
        Lấy 2 byte tiếp theo
    }
    Else
    {
        Đẩy byte trước vào mảng kết quả
        Biến lưu byte trước gán giá trị byte đó
        Lấy tiếp 1 byte nữa làm byte liền kế sau đó
        Cập nhật trỏ mảng nguồn.
    }
}
```

4.5 Modul thiết kế bài toán:

4.5.1 *Chương trình nén:*

```
int RleEncodeFile(char *inFile, char *outFile)
{
   FILE *fpIn;
   FILE *fpOut;
   int currChar, prevChar;
```

```
unsigned char count;
if ((fpIn = fopen(inFile, "rb")) == NULL)
{
    perror(inFile);
    return FALSE;
}
if (outFile == NULL)
{
    fpOut = stdout;
}
else
{
    if ((fpOut = fopen(outFile, "wb")) == NULL)
    {
       fclose(fpIn);
       perror(outFile);
       return FALSE;
    }
}
prevChar = EOF;
count = 0;
while ((currChar = fgetc(fpIn)) != EOF)
{
    fputc(currChar, fpOut);
    if (currChar == prevChar)
    {
       count = 0;
       while ((currChar = fgetc(fpIn)) != EOF)
       {
            if (currChar == prevChar)
            {
```

```
                count++;
                if (count == UCHAR_MAX)
                {
                    fputc(count, fpOut);
                    prevChar = EOF;
                    break;
                }
            }
            else
            {
                fputc(count, fpOut);
                fputc(currChar, fpOut);
                prevChar = currChar;
                break;
            }
        }
    }
    else
    {
        prevChar = currChar;
    }
    if (currChar == EOF)
    {
        fputc(count, fpOut);
        break;
    }
  }
  fclose(fpOut);
  fclose(fpIn);
  return TRUE;
}
```

4.5.2 Chương trình giải nén:

```
int RleDecodeFile(char *inFile, char *outFile)
{
   FILE *fpIn;
   FILE *fpOut;
   int currChar, prevChar;
   unsigned char count;
   if ((fpIn = fopen(inFile, "rb")) == NULL)
   {
        perror(inFile);
        return FALSE;
   }
   if (outFile == NULL)
   {
        fpOut = stdout;
   }
   else
   {
        if ((fpOut = fopen(outFile, "wb")) == NULL)
        {
           fclose(fpIn);
           perror(outFile);
           return FALSE;
        }
   }
   prevChar = EOF;
   while ((currChar = fgetc(fpIn)) != EOF)
   {
        fputc(currChar, fpOut);
        if (currChar == prevChar)
```

```
        {
            count = fgetc(fpIn);
            while (count > 0)
            {
                fputc(currChar, fpOut);
                count--;
            }
            prevChar = EOF;
        }
        else
        {
            prevChar = currChar;
        }
    }
    fclose(fpOut);
    fclose(fpIn);
    return TRUE;
}
```

4.6 Chương trình: phụ lục II

4.7 Chạy chương trình: