|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | |  | BỘ GIÁO DỤC VÀ ĐÀO TẠO  TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM |   **ĐỒ ÁN MÔN HỌC**  **BẢO MẬT THÔNG TIN**  Ngành: **CÔNG NGHỆ THÔNG TIN**  Môn Học: **BẢO MẬT THÔNG TIN**  Sinh viên thực hiện :  Hoàng Vũ 1811062571 18DTHB3  Lê Hoàng Nam 1811061517 18DTHB3  Nguyễn Quang Hưng 1811062450 18DTHB3  TP. Hồ Chí Minh, 2020 |
|  |

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

TP.HCM, ngày …. tháng …. năm 2020

Giảng viên hướng dẫn

**LỜI NÓI ĐẦU**

Ngày nay công nghệ thông tin là một trong những động lực quan trọng nhất thúc đẩy sự phát triển vượt bậc của nền kinh tế nước ta. Đồng thời tạo ra bầu không khí hoàn toàn mới trong mọi lĩnh vực khác nhau cùng với những lợi thế và thách thức mới. Công nghệ được xem như một gã khổng lồ bao quát hết tất cả mọi thứ từ công việc, dạy học, giải trí, thể thao, thậm chí làm việc nhà và các vật dụng thiết yếu trong gia đình càng lúc càng cải tiến, không ngừng phát triển nhầm đáp ứng những nhu cầu ngày càng cao của người sử dụng, dần dần trở thành một thế giới 4.0.

Khi nhìn vào định nghĩa mã hóa thông tin, chúng ta có thể thấy một trong những lĩnh vực rất cần thiết để áp dụng mã hóa trong quá khứ: trao đổi liên lạc giữa các bên trong chiến tranh. Hai đồng minh cần phải trao đổi thông tin mà đảm bảo kẻ thù không thể đọc được, vì vậy việc mã hóa dữ liệu khi trao đổi là một trong những yếu tố quyết định trong việc truyền tin mà không thể tạo được một kênh kết nối an toàn và bí mật.

Xuất phát từ những lý do trên, cùng với sự hướng dẫn của giảng viên, chúng em thực hiện 1 số chức năng như mã hóa mật khẩu, phân quyền account, theo dõi người dùng … để vận dụng những kiến thức đã được học vào thực tiễn, xây dựng một hệ thống bảo mật cho người dùng.

Chúng em xin chân thành cảm ơn!

**Sinh viên thực hiện**

**Hoàng Vũ**

**Lê Hoàng Nam**

**Nguyễn Quang Hưng**

**LỜI CẢM ƠN**

Sau quá trình học tập , nghiên cứu và sự nỗ lực của bản thân đặc biệt là sự hướng dẫn tận tình của Giảng Viên chúngem đã hoàn thành đề tài đúng tiến độ.

Đồng cảm ơn sâu sắc đến tất cả các quý thầy cô trường *Đại học Công nghệ Thông tin Tp.HCM* , các quý thầy cô khoa *Công nghệ Thông tin* đã tạo điều kiện hỗ trợ chúng em về cơ sở vật chất cũng như truyền đạt cho chúng em những kiến thức , những cái hay trong *Công Nghệ Thông tin.*

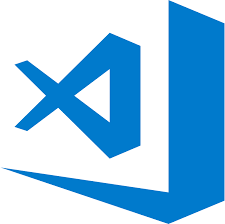
Mặc dù đã cố gắng hoàn thành đề tài trong phạm vi và khả năng cho phép nhưng chắc chắn sẽ không tránh khỏi những thiếu sót. Chúng em rất mong nhận được sự thông cảm, góp ý và tận tình chỉ bảo của quý thầy cô và các bạn.

**Chương 1: TỔNG QUAN**

**1.1. Giới thiệu đề tài**

***1.1. GIỚI THIỆU VỀ CÔNG CỤ THỰC HIỆN ĐỀ TÀI***

***1.1.1. Tổng quan về Visual Studio***



**Microsoft Visual Studio** là một môi trường phát triển tích hợp (IDE) từ Microsoft. Nó được sử dụng để phát triển chương trình máy tính cho Microsoft Windows, cũng như các trang web, các ứng dụng web và các dịch vụ web. Visual Studio sử dụng nền tảng phát triển phần mềm của Microsoft như Windows API, Windows Forms, Windows Presentation Foundation, Windows Store và Microsoft Silverlight. Nó có thể sản xuất cả hai ngôn ngữ máy và mã số quản lý.

Visual Studio bao gồm một trình soạn thảo mã hỗ trợ IntelliSense cũng như cải tiến mã nguồn. Trình gỡ lỗi tích hợp hoạt động cả về trình gỡ lỗi mức độ mã nguồn và gỡ lỗi mức độ máy. Công cụ tích hợp khác bao gồm một mẫu thiết kế các hình thức xây dựng giao diện ứng dụng, thiết kế web, thiết kế lớp và thiết kế giản đồ cơ sở dữ liệu. Nó chấp nhận các plug-in nâng cao các chức năng ở hầu hết các cấp bao gồm thêm hỗ trợ cho các hệ thống quản lý phiên bản (như Subversion) và bổ sung thêm bộ công cụ mới như biên tập và thiết kế trực quan cho các miền ngôn ngữ cụ thể hoặc bộ công cụ dành cho các khía cạnh khác trong quy trình phát triển phần mềm.

Visual Studio hỗ trợ nhiều ngôn ngữ lập trình khác nhau và cho phép trình biên tập mã và gỡ lỗi để hỗ trợ (mức độ khác nhau) hầu như mọi ngôn ngữ lập trình. Các ngôn ngữ tích hợp gồm có C, C++ và C++/CLI (thông qua Visual C++), VB.NET (thông qua Visual Basic.NET), C# (thông qua Visual C#) và F# (như của Visual Studio 2010). Hỗ trợ cho các ngôn ngữ khác như J++/J#, Python và Ruby thông qua dịch vụ cài đặt riêng rẽ. Nó cũng hỗ trợ XML/XSLT, HTML/XHTML, JavaScript và CSS.

Microsoft cung cấp phiên bản "Express" (đối với phiên bản Visual Studio 2013 trở về trước) và "Community" (đối với bản Visual Studio 2015 trở về sau) là phiên bản miễn phí của Visual Studio.

***1.1.2. Tổng quan về hệ quản trị cơ sở dữ liệu SQL Server***



Có rất nhiều hệ quản trị cơ sở dữ liệu mạnh mẽ như: Oracle, My SQL… nhưng chúng em sẽ sử dụng SQL Server để thực hiện đề tài*“Xây dựng phần mềm quản lý điểm sinh viên”* này.

Hệ quản trị cơ sở dữ liệu Microsoft SQL Server (MSSQL) là một trong những hệ quản trị cơ sở dữ liệu thông dụng nhất hiện nay. Đây là hệ quản trị cơ sở dữ liệu thường được sử dụng trong các hệ thống trung bình, với ưu điểm có các công cụ quản lý mạnh mẽ giúp cho việc quản lý và bảo trì hệ thống dễ dàng, hỗ trợ nhiều phương pháp lưu trữ, phân vùng và đánh chỉ mục phục vụ cho việc tối ưu hóa hiệu năng.

SQL Server luôn được Microsoft cải tiến để nâng cao hiệu năng, tính sẵn sàng của hệ thống, khả năng mở rộng và bảo mật, cung cấp nhiều công cụ cho người phát triển ứng dụng được tích hợp với bộ Visual Studio do Microsoft cung cấp. SQL Server có 4 dịch vụ lớn là Database Engine, Intergration Service, Reporting service và Analysis Services.

DataBase Engine: được phát triển để thực thi tốt hơn với việc hỗ trợ cả dữ liệu có cấu trúc và dữ liệu phi cấu trúc (XML).

Khả năng sẵn sàng của hệ thống được nâng cao, hỗ trợ các chức năng: Database mirroring (cơ sở dữ liệu gương), failover clustering , snapshots và khôi phục dữ liệu nhanh.

Việc quản lý chỉ mục được thực hiện song song với việc hoạt động của hệ thống. Người dùng có thể thêm chỉ mục, xây dựng lại chỉ mục hay xóa một chỉ mục đi trong khi hệ thống vẫn được sử dụng.

Chức năng phân vùng dữ liệu được hỗ trợ: Người dùng có thể phân vùng các bảng và chỉ mục cũng như quản lý phân vùng dữ liệu một cách dễ dàng. Việc hỗ trợ phân vùng dữ liệu giúp nâng cao hiệu năng hoạt động của hệ thống.

Dịch vụ đồng bộ hóa dữ liệu được mở rộng với việc hỗ trợ mô hình đồng bộ hóa ngang hàng. Đây là dịch vụ giúp đồng bộ hóa dữ liệu giữa các máy chủ dữ liệu, giúp mở rộng khả năng của hệ thống.

Dịch vụ tích hợp (Integration Service) thiết kế lại cho phép người dùng tích hợp dữ liệu và phân tích dữ liệu từ nhiều nguồn khác nhau. Hỗ trợ việc quản lý chất lượng dữ liệu và làm sạch dữ liệu, một công việc quan trọng trong tiến trình ETL.

Dịch vụ phân tích dữ liệu (Analysis Service): cung cấp khung nhìn tích hợp và thống nhất về dữ liệu cho người dùng, hỗ trợ việc phân tích dữ liệu.

Công cụ khai phá dữ liệu (Data mining) được tích hợp hỗ trợ nhiều thuật toán khai phá dữ liệu, hỗ trợ cho việc phân tích, khai phá dữ liệu và xây dựng các hệ thống hỗ trợ ra quyết định cho người quản lý.

Dịch vụ xây dựng quản lý báo cáo (Reporting Service) dựa trên nền tảng quản trị doanh nghiệp thông minh và được quản lý qua dịch vụ web. Báo cáo có thể được xây dựng dễ dàng với ngôn ngữ truy vấn MDX. Thông qua các công cụ trên Business Intelligent, người dùng dễ dàng truy cập báo cáo và trích xuất ra nhiều định dạng khác nhau thông qua trình duyệt web.

***1.1.3. Ngôn ngữ lập trình***

- C# (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000, trong đó người dẫn đầu là Anders Hejlsberg và Scott Wiltamuth.

- C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và nó được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java.

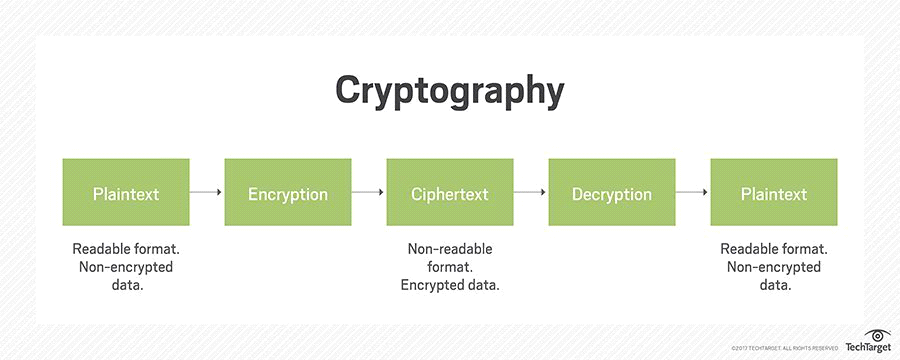
- C# được thiết kế cho Common Language Infrastructure (CLI), mà gồm Executable Code và Runtime Environment, cho phép chúng ta sử dụng các ngôn ngữ high-level đa dạng trên các nền tảng và cấu trúc máy tính khác nhau.

- C# với sự hỗ trợ mạnh mẽ của .NET Framework giúp cho việc tạo một ứng dụng Windows Forms hay WPF (Windows Presentation Foundation), . . . trở nên rất dễ dàng.

***1.2. GIỚI THIỆU VỀ MÃ HÓA THÔNG TIN***

Mã hóa thông tin là một hình thức biến đổi dữ liệu thành một dạng dữ liệu khác có ý nghĩa khác với dữ liệu trước khi bị biến đổi ban đầu, với mục đích chỉ cho phép một số người nhất định có thể đọc được dữ liệu ban đầu, thông qua việc giải mã dữ liệu sau khi biến đổi.

Hay nói cách khác, mã hóa là biến dữ liệu ban đầu A thành dữ liệu B, và việc đọc dữ liệu A sẽ thông qua việc giải mã dữ liệu B về A.



*Quá trình mã hóa và giải mã*

Định nghĩa khá phức tạp, nên chúng ta sẽ trả lời một số câu hỏi để làm rõ khái niệm mã hóa ở trên:

**Biến đổi dữ liệu là gì?**

Biến đổi dữ liệu là một quy tắc nào đó biến đổi một lượng dữ liệu này thành một lượng dữ liệu khác. Nếu nhìn theo khía cạnh toán học, thì biến đổi dữ liệu chính là một dạng hàm số y = f(x) với x là dữ liệu ban đầu, y là dữ liệu sau khi biến đổi từ dữ liệu x và f là hàm biến đổi.

Trong mật mã học, khi nghiên cứu về mã hóa thông tin, dữ liệu ban đầu được gọi là Plaintext (kí hiệu là P), dữ liệu sau khi mã hóa được gọi là Ciphertext (kí hiệu là C), hàm biến đổi/mã hóa được gọi là phương pháp mã hóa và được kí hiệu là E (Encryption).

**1.2. Đối tượng và phạm vi nghiên cứu**

Với các mặt hạn chế của hệ thống quản lý cũ của cửa hàng nên mục tiêu của nhóm em là xây dựng một hệ thống phần mềm trợ giúp hầu hết các công đoạn và nghiệp vụ quản lý. Đề tài “Xây dựng website bán điện thoại di động” nhằm đạt được các mục tiêu cụ thể như sau:

o **Không chồng chéo**: hệ thống mới cần rà soát và điều chỉnh lại chu trình  
nghiệp vụ cũ của cửa hàng, sao cho loại bỏ hoàn toàn việc chồng chéo  
như đã xảy ra trước đây.

o **Chính xác**: việc nhập, tính toán các giá bán và mua cần được tự động hóa  
một cách tối đa tránh các sai sót làm mất thời gian kiểm tra và có thể gây  
ra các thiệt hại cho cửa hàng.

o **Xử lý nhanh chóng**: Việc ghi phiếu và tính toán tổng tiền phải được làm một cách nhanh chóng và chính xác.

o **Tìm kiếm dễ dàng**: Việc tra cứu, tìm kiếm các thông tin liên quan đến  
hàng hóa, nhà cung cấp, nhân viên, khách hàng và các giao dịch mua bán  
cần phải được thực hiện một cách dễ dàng.

o **Ổn định và chiếm ít tài nguyên máy** : W cần có sự ổn định để  
tránh sai xót về dữ liệu để tránh làm tổn hại cho cửa hàng. Đồng thời cũng  
chiếm ít tài nguyên máy để hệ thống chạy được nhanh và chạy được trên  
các máy có cấu hình thấp.

**1.3. Mục tiêu đề tài**

Xây dựng website quảng bá và bán sản phẩm điện thoại di động của các hãng điện thoại đến khách hàng một cách nhanh nhất, chính xác và thuận tiện nhất.

Website sau khi xây dựng sẽ đáp ứng được những yêu cầu cơ bản sau :

Giúp cho khách hàng :

- Xem sản phẩm và thông tin sản phẩm.

- Tìm kiếm sản phẩm theo các tiêu như : Tên sản phẩm , giá, sản phẩm mới, sản phẩm khuyến mại, hãng sản xuất, cấu hình máy tùy chọn

Giúp cho người quản trị :

- Quản lý phòng và dịch vụ (thêm, sửa, xóa sản phẩm ….)

- Quản lý tài khoản (thêm, sửa, xóa hãng sản xuất ….)

- Quản lý các hóa đơn

Quản lý hoạt động của công ty

Quyết định giá chính thức cho từng phòng và dịch vụ

Từ đó nắm bắt được tình hình của khách, tình hình để có kế hoạch định hướng, phát triển nhằm đáp ứng yêu cầu khách hàng và tiến triển cho khách sạn.

Quản lý bán sản phẩm.

Quản lý khách hàng: Bộ phận này chịu trách nhiệm lưu trữ toàn bộ thông tin khách hàng cung cấp.

Lập đơn đặt hàng: Bộ phận này căn cứ vào nhu cầu đặt hàng của khách để lập đơn đặt hàng sau đó chuyển cho khách hàng và chuyển cho bộ phận quản trị mạng.

Lập hoá đơn: Bộ phận này dựa vào đơn đặt hàng để xuất hoá đơn cho khách hàng.

Thống kê tình hình khách sạn : Báo cáo tình hình cụ thể phòng nào trống để ban quản trị có kế hoạch phù hợp.

Quản lý và xử lý đơn hàng:

- Đánh dấu tình trạng phòng

- Xoá hóa đơn

- Kiểm tra độ chính xác của đơn hàng.

**1.4. Mục đích chức năng**

*1.4.1 Chức năng người dùng*

|  |  |  |
| --- | --- | --- |
| STT | Chức năng | Mô tả |
| 1 | Xem phòng | Người xem có thể xem thông tin về phòng bao gồm: số phòng, tình trạng, giá, phân loại |
| 2 | Bình luận | Người xem có thể bình luận về sản phẩm, nội dung và thông tin người bình luận cần lưu trong database. |
| 3 | Đặt phòng | Chức năng cho phép quản duyệt phòng trên và nhấn nút đặt phòng đã chọn. |
| 4 | Dịch vụ | Cho phép đặt dịch vụ trên menu |
| 7 | Tạo mới tài khoản | Cho phép tạo mơi tài khoản |

Chi tiết chức năng phía người dùng

Khi quản trị truy cập app, ngay tại trang chủ khách sẽ được giới thiệu tình trạng phòng đã đặt hay chưa.

Các dịch vụ được trình bày rõ ràng với mô tả tóm tắt, giá, Khi kích vào tên, dịch vụ bất kỳ khách hàng sẽ có thể đặt dịch vụ, mô tả tóm tắt và chi tiết dịch vụ sẽ được trình bày rõ ràng và đầy đủ.

Mỗi khi tìm được dịch vụ phù hợp, quản có thể click vào nút “đặt dịch vụ” của sản phẩm đó, dịch vụ đó sẽ tự động thêm vào danh sách với số lượng là 1.

Khi đã chọn đủ sản phẩm để mua, khách click vào chức năng giỏ hàng, danh sách tên các sản phẩm cùng số lượng, đơn giá sẽ được trình bày chi tiết. Tại đây khách hàng có thể thay đổi số lượng, thêm bớt hoặc xóa bỏ sản phẩm. Khi đồng ý mua khách hàng click vào ô “thanh toán”. Một hóa đơn hàng hóa, chi tiết về tên sản phẩm, số lượng, đơn giá, thành tiền và tổng tiền trên hóa đơn sẽ được hiển thị. Bằng cách điền đẩy đủ các

Đăng ký tài khoản khách hàng và đăng nhập

Để thuận tiện cho quá trình mua bán sản phẩm và để sử dụng được một số tính năng nâng cao của hệ thống,ban quản trị có thể thêm mới tài khoản và đặt quyền hạn cho tài khoản đó

Khi đã có tài khoản, quản trị có thể truy cập hệ thống để đặt phòng, dịch vụ, xem các thông tin về lược sử đặt của khách hàng và sử dụng các tính năng nâng cao khác của hệ thống.

*Cập nhật thông tin nhà sản xuất, loại sản phẩm và sản phẩm*

Danh sách nhà sản xuất và loại sản phẩm, sản phẩm được lưu trữ và cập nhật mỗi khi có thay đổi hoặc bổ sung.

**CHƯƠNG 2. CƠ CHẾ HOẠT ĐỘNG**

***2.1. Đôi nét về mật mã Vigenere***

Mật mã Vigenère là một phương pháp mã hóa chữ văn bản tiếng Anh, lần đầu tiên được mô tả bởi Giovan Battista Bellaso vào năm 1553. Phương pháp mã hóa mật mã Vigenère dễ hiểu và dễ thực hiện, nhưng chỉ đến năm 1863 với nhiều nỗ lực suốt ba thế kỷ, Friedrich Kasiski mới xuất bản một phương pháp chung để giải mã mật mã Vigenère.

Mật mã Vigenère là tập hợp các quy tắc thay thế chữ cái đơn trong bảng chữ cái tiếng Anh qua việc sử dụng 26 mật mã Caesar với các bước dịch chuyển từ 0 đến 25 tương ứng từ chữ ‘a’ đến chữ ‘z’. Cụ thể, bản mã Vigenère được lập theo công thức sau:

*ci = (pi + ki) mod 26, i=1,…,l*

trong đó, *C = {c1…cl}* là bản mã, *P={p1…pl}* là bản gốc, *K = {k1…kl}* là dãy khóa và l là độ dài bản rõ. Tương tự, bản gốc *P* có thể được tính nếu biết khóa và bản mã theo công thức:

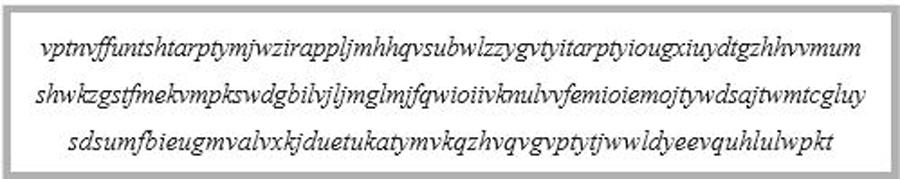
*pj = (cj – kj) mod 26, j = 1,…,l*

Mật mã Vigenère không thể phá vỡ trong trường hợp sử dụng các khóa đủ dài. Nhưng với các khóa ngắn hoặc nếu nhà thám mã có đủ nhiều bản mã so với độ dài khóa thì khá dễ để phá vỡ. Việc thám mật mã Vigenère thường tiến hành theo hai bước là: xác định độ dài chu kỳ của khóa trước, sau đó tìm khóa cụ thể.

***Tìm chu kỳ khóa của mật mã Vigenère***

Đầu tiên cần lưu ý là chu kỳ của khóa tìm được có thể không đúng với thực tế được sử dụng. Nếu bản mã đủ dài thì có thể là chính xác, các phương pháp được cung cấp ở đây là gần đúng.

Mật mã Vigenère áp dụng các mật mã Caesar khác nhau cho các chữ cái liên tiếp. Ví dụ một bản mã Vigenère như sau:



*Hình 1. Bản mã hóa sử dụng mật mã Vigenère*

Mật mã Caesar là một dạng của mật mã thay thế, theo đó mỗi ký tự trong bản rõ được thay thế bằng một ký tự cách nó một đoạn trong bảng chữ cái để tạo thành bản mã. Giả sử với khóa là 3 (dịch 3 vị trí trong bảng chữ cái), thì chữ ‘a’ sẽ được thay bằng chữ ‘d’, chữ ‘b’ sẽ được thay bằng ‘e’ và cứ thế đến hết bản rõ. Phương pháp này được đặt tên là Caesar, vị Hoàng đế đã sử dụng loại mật mã này thường xuyên trong công việc.

Nếu mật mã Vigenère sử dụng khóa có chu kỳ 3 là 'PUB', thì chữ cái rõ đầu tiên được mã hóa bằng mật mã Caesar với khóa là 16 (P là chữ cái thứ 16 của bảng chữ cái), chữ cái thứ hai được mã với khóa là 21 (chữ cái U) và chữ cái thứ ba được mã với khóa là 2 (chữ cái B). Chữ cái rõ thứ 4 được mã hóa quay lại bằng chữ khóa thứ nhất (khóa 16). Kết quả là, các chữ cái ở các vị trí 1,4,7,10,... đều được mã hóa bằng cùng một mật mã Caesar với chữ khóa là P. Các chữ cái ở các vị trí 2,5,8,11,... và 3,6,9,12,... được mã hóa bằng mật mã Caesar với khóa tương ứng là chữ U và B.

Như vậy, trình tự chính xác sẽ phụ thuộc vào chu kỳ của khóa mật mã, tức là độ dài khóa, như với ví dụ trên thì độ dài chu kỳ khóa là 3.

***Phương pháp tìm chu kỳ khóa theo sự lặp lại của nhóm chữ cái***

Để xác định chu kỳ của khóa mật mã Vigenère, phương pháp Kasiski xem xét sự lặp lại của các nhóm chữ cái như Hình 2.

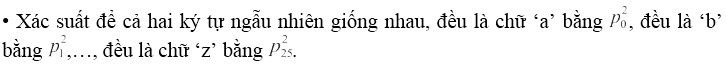
*Hình 2. Sự lặp lại của nhóm chữ cái*

Đoạn lặp lại loạt ***VHVS*** gồm 18 ký tự, gợi ý rằng độ dài khóa có thể là 18, 9, 6, 3, 2. Còn đoạn lặp lại loạt **QUCE** là 30 ký tự, gợi ý độ dài khóa là 30, 15, 10, 6, 5, 3, 2. Kết hợp lại, độ dài khóa có thể là 6, 3 hoặc 2.

***Chỉ số trùng hợp (Index of coincidence - I.C. còn được ký hiệu là****Ic()****)***

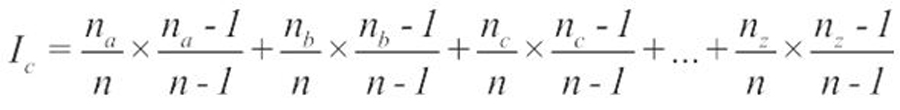
Nếu trong bản mã không có sự lặp lại của một loạt chữ cái nào, người thám mã sẽ sử dụng đến chỉ số trùng hợp Ic.

Giả sử X là một chuỗi ký tự trong tiếng Anh, ký hiệu xác suất xuất hiện của các chữ a, b,…, z lần lượt là p0, p1,…, p25. Khi đó:



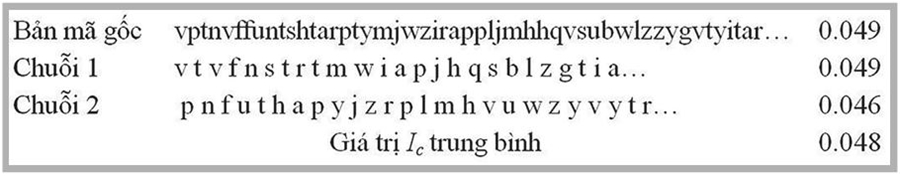
• Khi đó:

Ic (x) = ∑  = 0.0822+0.0152+…+0.0012 = 0.065

Chỉ số trùng hợp đôi khi được gọi là tỷ lệ lặp lại. Nếu bản mã cụ thể có độ dài n, na là tần số xuất hiện của chữ ‘a’, nb là tần số xuất hiện của chữ ‘b’…, thì chỉ số trùng hợp gần đúng được tính theo công thức sau: 

Chỉ số trùng hợp (Ic) là một kỹ thuật thống kê giúp xác định một đoạn văn bản có đáp ứng quy luật ngôn ngữ của tiếng Anh. Một tính chất quan trọng của kỹ thuật là giá trị Ic không thay đổi nếu áp dụng mật mã thay thế đơn cho văn bản. Điều này là do *Ic* dựa trên tần số xuất hiện của chữ cái và mật mã thay thế đơn không làm thay đổi tần số của bộ chữ cái riêng lẻ. Với văn bản tiếng Anh sẽ có giá trị *I*c làm tròn là 0.06, nếu các ký tự có phân phối đồng đều thì *Ic* gần hơn với 0,03 - 0,04.

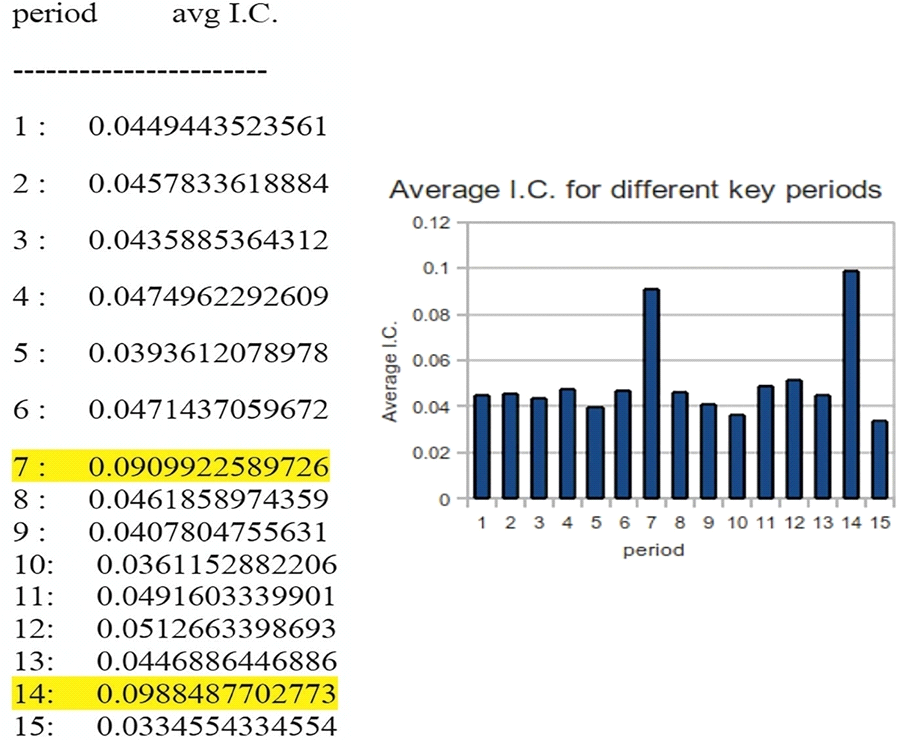
Phương pháp dùng chỉ số trùng hợp để xác định chu kỳ của khóa mật mã Vigenère thực hiện như sau. Trước tiên, giả sử độ dài khóa là 2, thực hành trích xuất hai chuỗi tại các vị trí 1, 3, 5, 7,... và 2, 4, 6, 8,... từ bản mã như Hình 1 (lưu ý rằng *Ic* được tính bằng cách sử dụng toàn bộ chuỗi mã, không chỉ là phần được hiển thị).



Tương tự với trường hợp độ dài khóa là 3 sẽ có 3 chuỗi, tương ứng với các giá trị *Ic* như sau:

Như vậy, Ic trung bình đối với trường hợp chu kỳ 2 là khoảng 0,048 và đối với trường hợp chu kỳ 3 là khoảng 0,047.

Quy trình này sẽ được lặp lại cho tất cả các độ dài khóa muốn kiểm tra. Ví dụ tiếp tục tính với chu kỳ khóa lên đến 15 sẽ có các giá trị trung bình I.C (avg I.C.) tương ứng như Hình 3.



*Hình 3. Các giá trị trung bình I.C.*

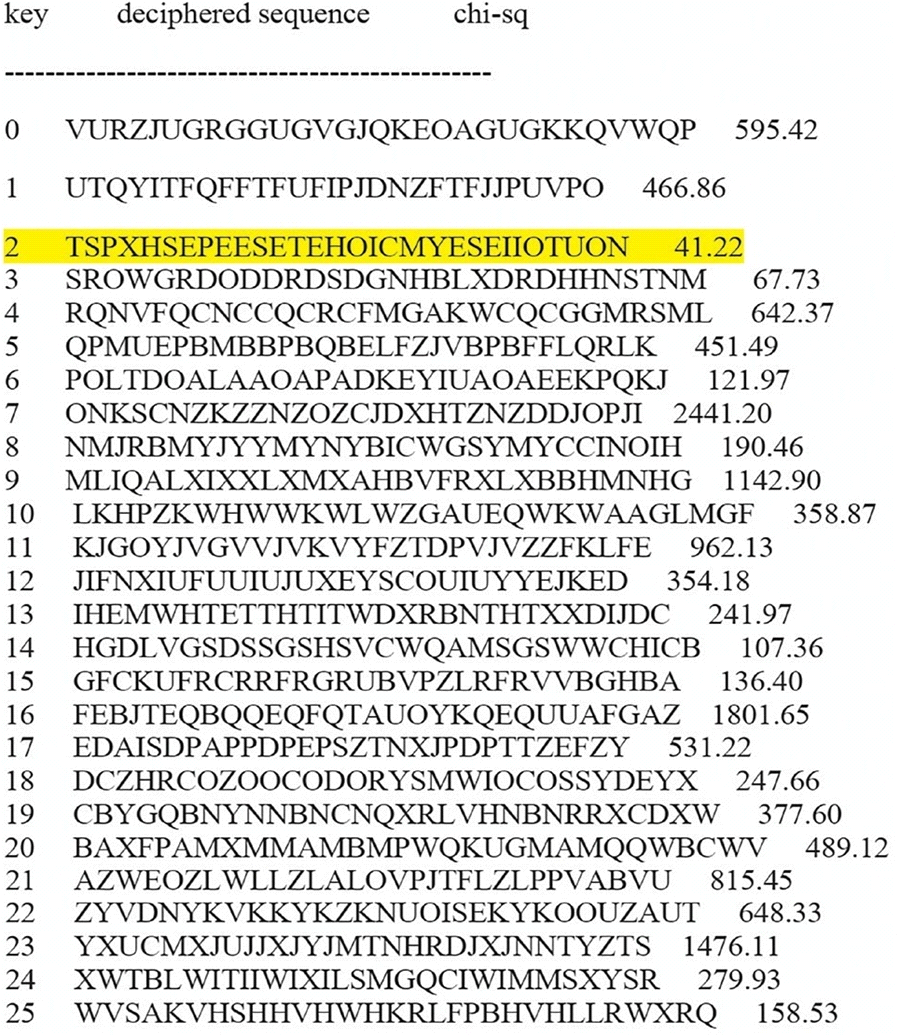
Theo cột giá trị, hoặc theo biểu đồ, có 2 giá trị trung bình I.C cao đột biến đã gợi ý rằng khóa mật mã có thể có độ dài 7 hoặc 14. Cả hai xác suất này phải được tiếp tục kiểm tra.

***Tìm khóa mật mã theo kỹ thuật thống kê khi bình phương***

Ví dụ thám mã với khóa chu kỳ 7 (sử dụng 7 mật mã Caesar) cho bản mã ở Hình 1, việc tìm khóa khá dễ dàng. Thám mã sẽ so sánh giá trị thống kê Khi bình phương của dãy phá mã với giá trị phân phối tần số xuất hiện chữ cái tiếng Anh.

Lập   chuỗi   chữ   cái   lấy   từ   các   vị   trí   1, 8, 15, 22,… của bản mã ở Hình 1 (vurzjugrggugvgjqkeoagugkkqvwqp…). Đây là chuỗi được mã hóa với cùng một mật mã Caesar.

Giải mã chuỗi này với cả 26 mật mã Caesar có thể, lập bảng so sánh phân phối tần số của văn bản được giải mã với phân phối tần số tiếng Anh cho mỗi khóa. Tương ứng, sẽ thu được 26 giá trị thống kê Khi bình phương. Khóa chính xác sẽ tương ứng với văn bản được giải mã với thống kê Khi bình phương thấp nhất. Kết quả cụ thể như Hình 4 đã tìm được chữ cái khóa đầu tiên, theo đó giá trị Khi bình phương nhỏ nhất là 41.22, tương ứng với khóa là 2 (chữ cái ‘c’).



**Chương 2: PHÂN TÍCH HỆ THỐNG**

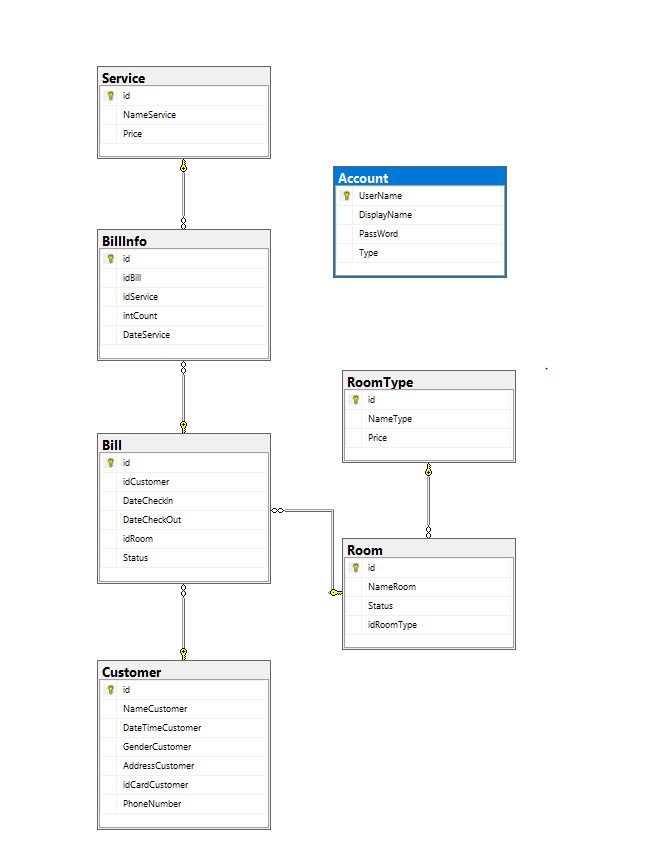
**2.1. Tổng quan về cơ sở dữ liệu**

*2.1.1. Cở sở dữ liệu là gì?*

2.1.1.1. Khái niệm

Cơ sở dữ liệu ( CSDL ) là một tập hợp có sắp xếp các thông tin, dữ liệu về một vấn đề nào đó, nhằm giúp cho người sử dụng dễ dàng tra cứu, sử dụng và cập nhập.

Đặc điểm chủ yếu của CSDL là cách tổ choc, sắp xếp thông tin, các dữ liệu có liên quan với nhau sẽ được lưu trong các tập tin hay trong các bảng. Nói cách khác nó là tập hợp các dữ liệu có quan hệ với nhau.



Cơ sở dữ liệu

2.1.1.2. Sự cần thiết của cơ sở dữ liệu

Ngày nay nhu cầu tích luỹ lưu trữ và xử lý dữ liệu đã có mặt trên mọi lĩnh vực, trong mọi hoạt động của con người. Nhưng thông tin ngày càng lớn và phức tạp, buộc con người phải sắp xếp các thông tin sao cho có khoa học, vì vậy đòi hỏi họ phải sử dụng CSDL.

2.1.1.3. Hệ quản trị cơ sở dữ liệu ( Database Management System )

Là một công cụ cho phép quản lý và tương tác với cơ sở dữ liệu như cho phép xử lý, thay đổi, truy xuất cơ sở dữ liệu. Theo nghĩa này hệ quản trị cơ sở dữ liệu có nhiệm vụ rất quan trọng như là một bộ phận diễn dịch với ngôn ngữ bậc cao nhằm giúp người sử dụng có thể dùng được hệ thống mà ít nhiều không cần quan tâm đến thuật toán chi tiết hoặc biểu diễn dữ liệu trong máy tính. Hầu hết các hệ quản trị cơ sở dữ liệu đều thực hiện các chức năng sau:

- Lưu trữ dữ liệu

- Tạo và duy trì cấu trúc dữ liệu

- Hỗ trợ bảo mật và riêng tư.

- Cho xem và xử lý các dữ liệu lưu trữ

- Cung cấp một cơ chế chỉ mục ( index ) hiệu quả để truy cập nhanh các dữ liệu lựa chọn.

- Cung cấp tính nhất quán giữa các bản ghi khác nhau.

- Bảo vệ dữ liệu khỏi mất mát bằng các quá trình sao lưu ( backup ) và phục hồi dữ liệu ( recovery ).

Trong hệ quản trị cơ sở dữ liệu quan hệ dữ liệu được tổ chức thành các bảng, các bảng bao gồm các trường và các trường chứa các bản ghi. Mỗi trường tương ứng với một mục dữ liệu, hai hay nhiều bảng có thể liên kết nếu chúng có một hay nhiều trường chung.

**2.2. Giới thiệu ngôn ngữ lập trình c#**

*2.2.1. Tổng quan ngôn ngữ C#*

**C#** là một ngôn ngữ lập trình hiện đại được phát triển bởi Microsoft và được phê duyệt bởi European Computer Manufacturers Association (ECMA) và International Standards Organization (ISO). **C#** được phát triển bởi Anders Hejlsberg và nhóm của ông trong việc phát triển .Net Framework. **C#** được thiết kế cho các ngôn ngữ chung cơ sở hạ tầng (Common Language Infrastructure – CLI), trong đó bao gồm các mã (Executable Code) và môi trường thực thi (Runtime Environment) cho phép sử dụng các ngôn ngữ cấp cao khác nhau trên đa nền tảng máy tính và kiến trúc khác nhau.

**Ngôn ngữ ra đời cùng với .NET**

* Kết hợp **C++** và Java.
* Hướng đối tượng.
* Hướng thành phần.
* Mạnh mẽ (robust) và bền vững (durable).
* Mọi thứ trong **C#** đều Object oriented.
* Kể cả kiểu dữ liệu cơ bản.
* Chỉ cho phép đơn kế thừa.
* Dùng interface để khắc phục.
* Lớp **Object** là cha của tất cả các lớp.
* Mọi lớp đều dẫn xuất từ Object.
* Cho phép chia chương trình thành các thành phần nhỏ độc lập nhau.
* Mỗi lớp gói gọn trong một file, không cần file header như **C/C++**.
* Bổ sung khái niệm namespace để gom nhóm các lớp.
* Bổ sung khái niệm “*property*” cho các lớp.
* Khái niệm delegate & event.

**Vai trò C# trong .NET Framework**

* .NET runtime sẽ phổ biến và được cài trong máy client:
* Việc cài đặt App **C#** như là tái phân phối các thành phần .NET
* Nhiều App thương mại sẽ được cài đặt bằng **C#**.
* **C#** tạo cơ hội cho tổ chức xây dựng các App Client/Server n-tier.
* Kết nối ADO.NET cho phép truy cập nhanh chóng & dễ dàng với SQL Server, Oracle…
* Cách tổ chức .NET cho phép hạn chế những vấn đề phiên bản:
* Loại bỏ “*DLL Hell* ”…
* ASP.NET viết bằng **C#**:
* GUI thông minh.
* Chạy nhanh hơn (đặc tính của **.**NET)
* Mã ASP.NET ko còn là mới hỗn độn.
* Khả năng bẫy lỗi tốt, hỗ trợ mạnh trong quá trình xây dựng App Web.

**Quá trình dịch CT C#**

* Mã nguồn **C#** (tập tin \*.cs) được biên dịch qua MSIL:
* MSIL: tập tin .exe hoặc .dll.
* MSIL được CLR thông dịch qua mã máy:
* Dùng kỹ thuật JIT (just-in-time) để tăng tốc độ.

**Tài liệu tham khảo**

S. S. Omran A. S. Al-Khalid D. M. Al-Saady, A Cryptanalytic Attack on Vigenère Cipher Using Genetic Algorithm, 2011 IEEE Conference on Open Systems (ICOS2011), September 25 - 28, 2011, Langkawi, Malaysia

Mehmet E. Dalkilic and Cengiz Gungor, An Interactive Cryptanalysis Algorithm for the Vigenère Cipher, Ege University, International Computer Institute Bornova 35100 Izmir, TURKEY

