

# LẬP TRÌNH TRỰC QUAN

GVGD: TS. Nguyễn Thị Xuân Hương  
[huongntx@uit.edu.vn](mailto:huongntx@uit.edu.vn)

# Mục tiêu

- Nắm vững kiến thức, phương pháp lập trình trên môi trường windows và các công nghệ có liên quan
- Nắm được các kỹ thuật lập trình cao cấp với nhiều tính năng mới cùng ngôn ngữ C#
- Hiểu rõ được lợi ích của các công cụ/ngôn ngữ lập trình có liên quan trên môi trường Windows
- Xây dựng được ứng dụng trực quan

# Nội dung

1. Tổng quan - Các khái niệm cơ bản
2. Tổng quan – Ngôn ngữ lập trình C#
3. Lập trình giao diện đồ họa
4. Xử lý sự kiện
5. Thư viện đồ họa GDI
6. Multi-Document Interface (MDI)
7. Đa tiến trình
8. Multimedia

# Tài liệu tham khảo

- Giáo trình lập trình trên Windows, Vũ Thanh Nguyên, Phạm Thi Vương, Trần Anh Dũng, Huỳnh Ngọc Tín, NXB ĐHQG TPHCM, 2010
- Tài liệu hướng dẫn thực hành Lập trình môi trường Windows, Phạm Thị Vương, Lê Minh Trí, Nguyễn Hoàng Vũ, NXB ĐHQG TPHCM, 2014
- Windows System Programming, 4th Edition, Johnson (John) M. Hart, AddisonWesley, 2010.
- Programming Microsoft Windows with C#, Charles Petzold, Microsoft Press, 2001
- Programming Microsoft Windows Forms, Charles Petzold, Microsoft Press, 2005

# Đánh giá môn học

- Bài tập/chuyên cần: **20%**
- Thực hành: **30%**
- Đồ án: **50%**
  - Làm nhóm 2-3 sv
  - Báo cáo tiến độ: tuần 8
  - Báo cáo final: tuần cuối cùng (tuần 15-16)

# Danh sách đồ án tham khảo

1. Xây dựng chương trình Explorer (tương tự Window Explorer)
2. Xây dựng chương trình Wordpad
3. Xây dựng chương trình Paint (Windows)
4. Xây dựng chương trình hỗ trợ dạy và minh họa cho các cấu trúc dữ liệu: Danh sách liên kết, stack, queue, cây nhị phân,...
5. Xây dựng chương trình hỗ trợ dạy và minh họa cho các thuật toán sắp xếp
6. Xây dựng chương trình hỗ trợ dạy và minh họa cho cấu trúc dữ liệu: Cây đẻ đen
7. Xây dựng phần mềm hỗ trợ giải bài tập khảo sát hàm số

# Danh sách đồ án tham khảo

1. Xây dựng phần mềm hỗ trợ giải bài tập hình học giải tích
2. Xây dựng phần mềm thí nghiệm quang hình học
3. Xây dựng phần mềm thí nghiệm mạch điện
4. Xây dựng trò chơi cờ tướng
5. Xây dựng trò chơi cờ vua
6. Xây dựng trò chơi rắn săn mồi
7. Xây dựng trò chơi Line
8. Xây dựng trò chơi xếp hình
9. Xây dựng trò chơi xếp gạch

# Danh sách đồ án tham khảo

- Nhóm các chương trình quản lý
  - 1. Phần mềm quản lý thư viện
  - 2. Phần mềm quản lý nhà sách
  - 3. Phần mềm quản lý khách sạn
  - 4. Phần mềm quản lý phòng mạch tư
  - 5. Phần mềm quản lý sổ tiết kiệm
  - 6. Phần mềm quản lý bán vé máy bay
  - 7. Phần mềm quản lý cây gia phả
  - 8. Phần mềm quản lý vật tư
  - 9. Phần mềm quản lý bệnh viện
  - 10. Phần mềm quản lý nhân sự cho công ty
  - 11. Phần mềm quản lý phân phối dược phẩm

# Các đề tài tự chọn

- Báo trước với GV để nhận góp ý

# Tạo nhóm và đăng kí nhóm

???

# **Chương 1:**

# **TỔNG QUAN LẬP TRÌNH**

# **MÔI TRƯỜNG WINDOWS**

# Nội dung chính



Lịch sử Windows



Đặc điểm môi trường Windows



Lập trình hướng sự kiện



.NET Framework



Visual Studio

# Nội dung chính

1

Lịch sử Windows

2

Đặc điểm môi trường Windows

3

Lập trình hướng sự kiện

4

.NET Framework

5

Visual Studio

# Câu hỏi thảo luận

1. Các phiên bản và năm ra đời của HĐH Windows
2. So sánh Windows và DOS
3. Các ưu điểm của ứng dụng windows
4. .Net Framework là gì? Ý nghĩa?

# Lịch sử Windows

	1990	1993	1994	1995	1996	1997
<b>Windows Desktop</b>	Windows 3x	Windows NT		Windows 95	Windows NT Workstation	
<b>Windows Server</b>		Windows NT Advanced Server 3.1	Windows NT Server 3.5	Windows NT Server 3.51	Windows NT Server 4.0	Windows NT Server Enterprise Edition
<b>Windows Cloud</b>						

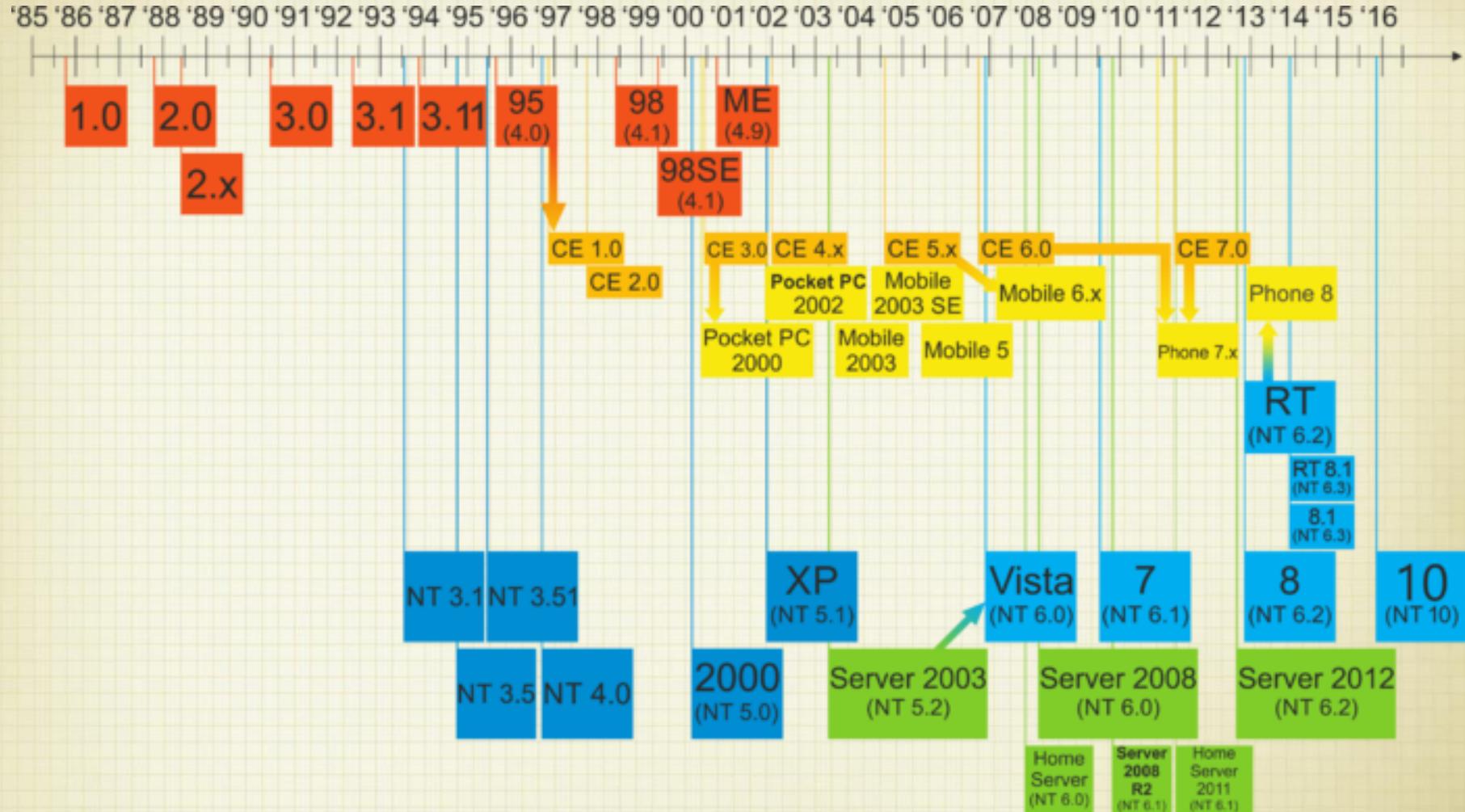
# Lịch sử Windows

	1998	2000	2001	2003	2006	2008	2009
<b>Windows Desktop</b>	Windows 98	1. Windows ME 2. Windows 2000 Professional	Windows XP		Windows Vista		Windows 7
<b>Windows Server</b>	Windows NT Server 4.0 Terminal Server Edition	Windows 2000 Server Family		1. Windows Server 2003 2. Windows Server Systems		Windows Server 2008	
<b>Windows Cloud</b>						Windows Azure	

# Lịch sử Windows

	2012	2013	2014	2015			
<b>Windows Desktop</b>	Windows 8	Windows 8.1		Windows 10			
<b>Windows Server</b>	Windows Server 2012 R2	Windows Server 2012 R2					
<b>Windows Cloud</b>			Microsoft Azure				

# Lịch sử Windows



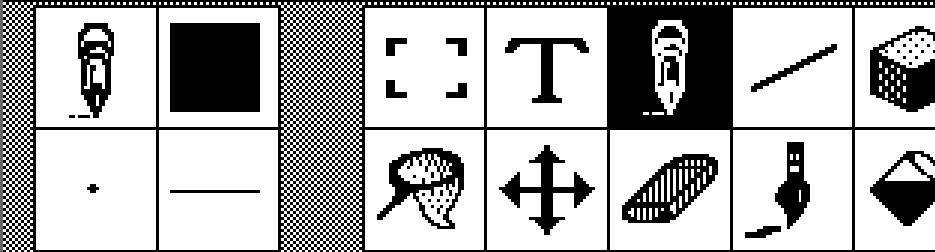
# Thị phần sử dụng

- 06/2018

Desktop OS	Net Applications	StatCounter
Windows XP	4.08%	2.45%
Windows Vista	0.30%	0.54%
Windows 7	41.74%	32.67%
Windows 8	1.08%	1.84%
Windows 8.1	4.95%	6.35%
Windows 10	35.71%	38.54%
All listed versions	87.86%	82.39%

Paint - (untitled)

File Edit Font FontSize Style  
Palette Options



Write - (Untitled)

File Edit Search Character  
Paragraph Document

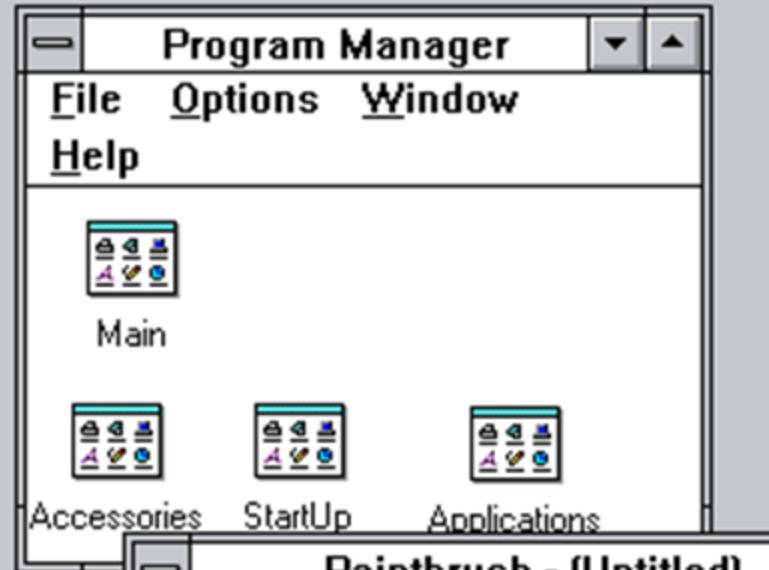
## Windows 1.01 (1985)

- Rudimentary toolbar
- MS-DOS Executive
- Each window has its own menu
- Tiled windows



Page 1





Write - (Untitled)

File Edit Find Character Paragraph Document Help

## Windows 3.0 (1990)

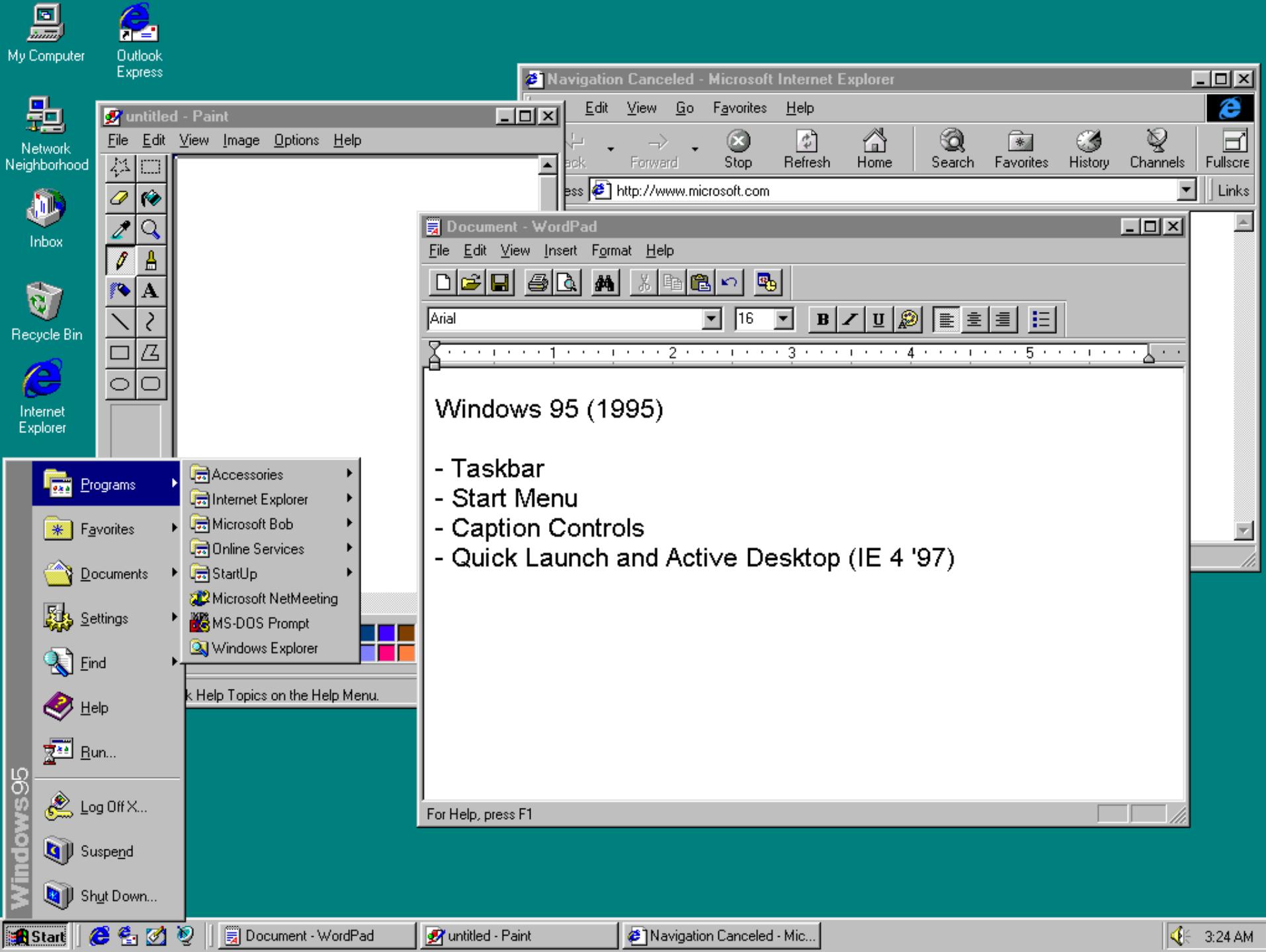
- Program Manager
- Minimize to the desktop
- Overlapping windows

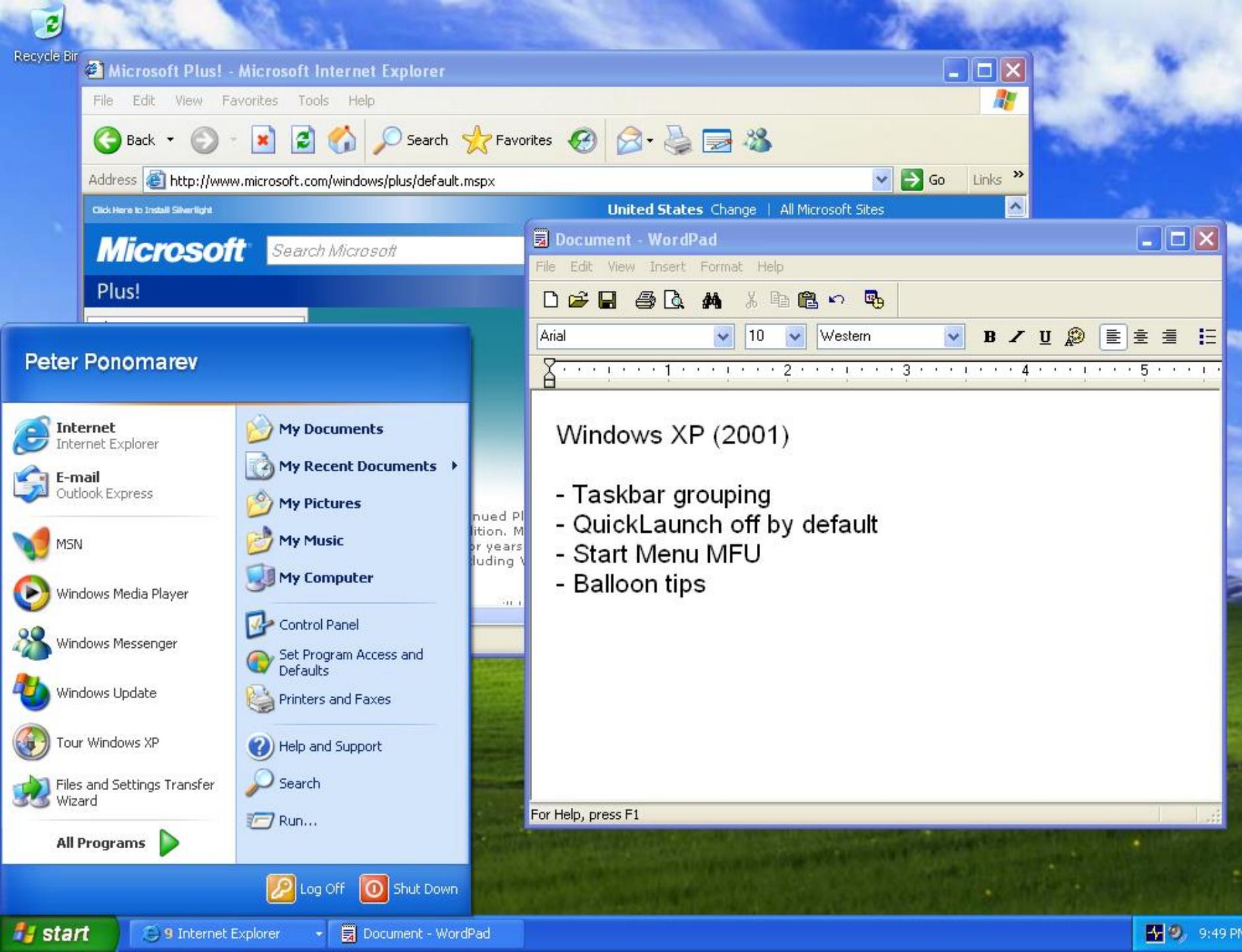
x

Page 1

This screenshot shows the Windows 3.0 Write application window. The menu bar includes File, Edit, Find, Character, Paragraph, Document, and Help. The main area displays the text "Windows 3.0 (1990)". A vertical scroll bar is on the right side of the window.



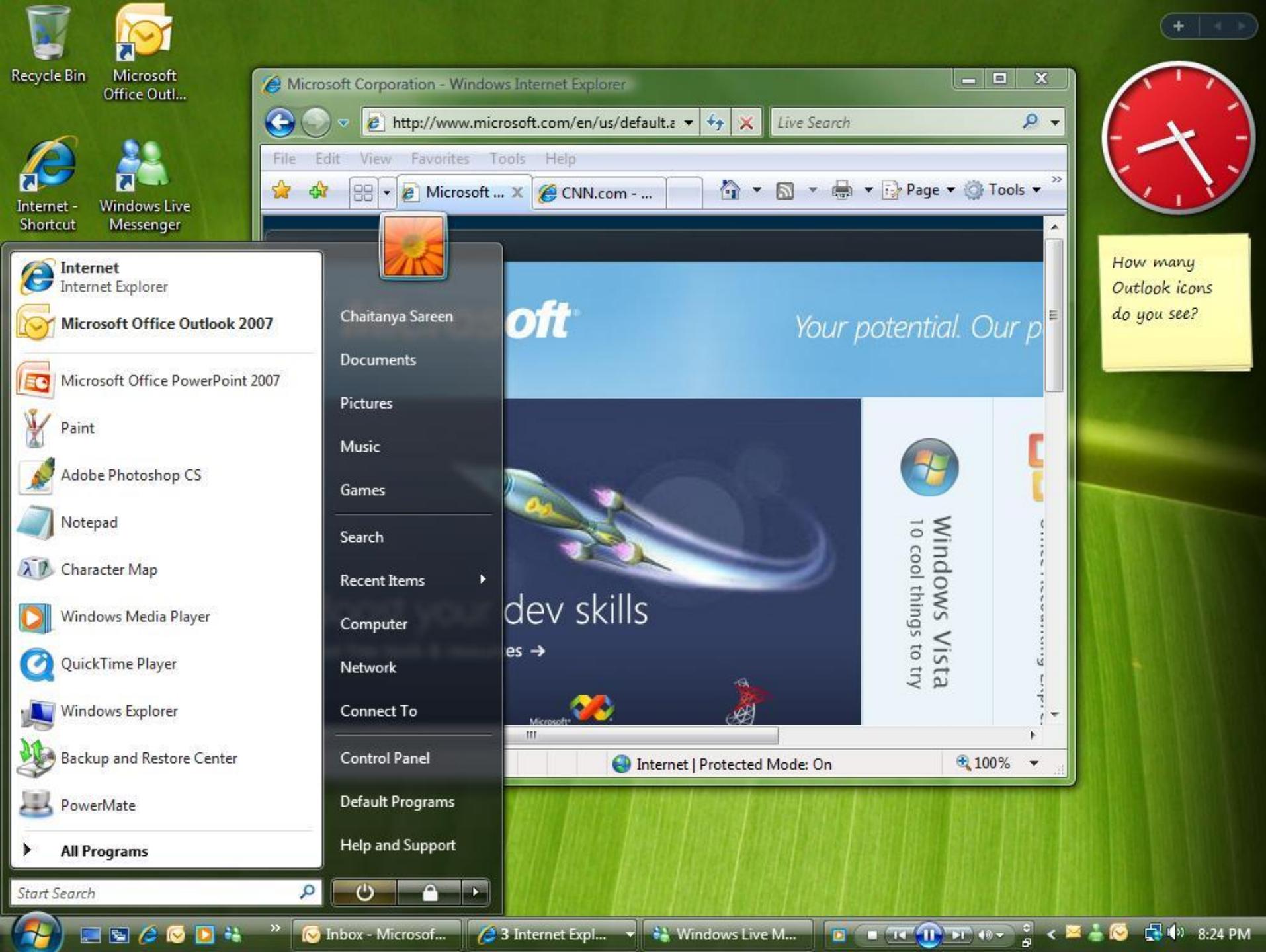




## Windows XP (2001)

- Taskbar grouping
- QuickLaunch off by default
- Start Menu MFU
- Balloon tips

For Help, press F1

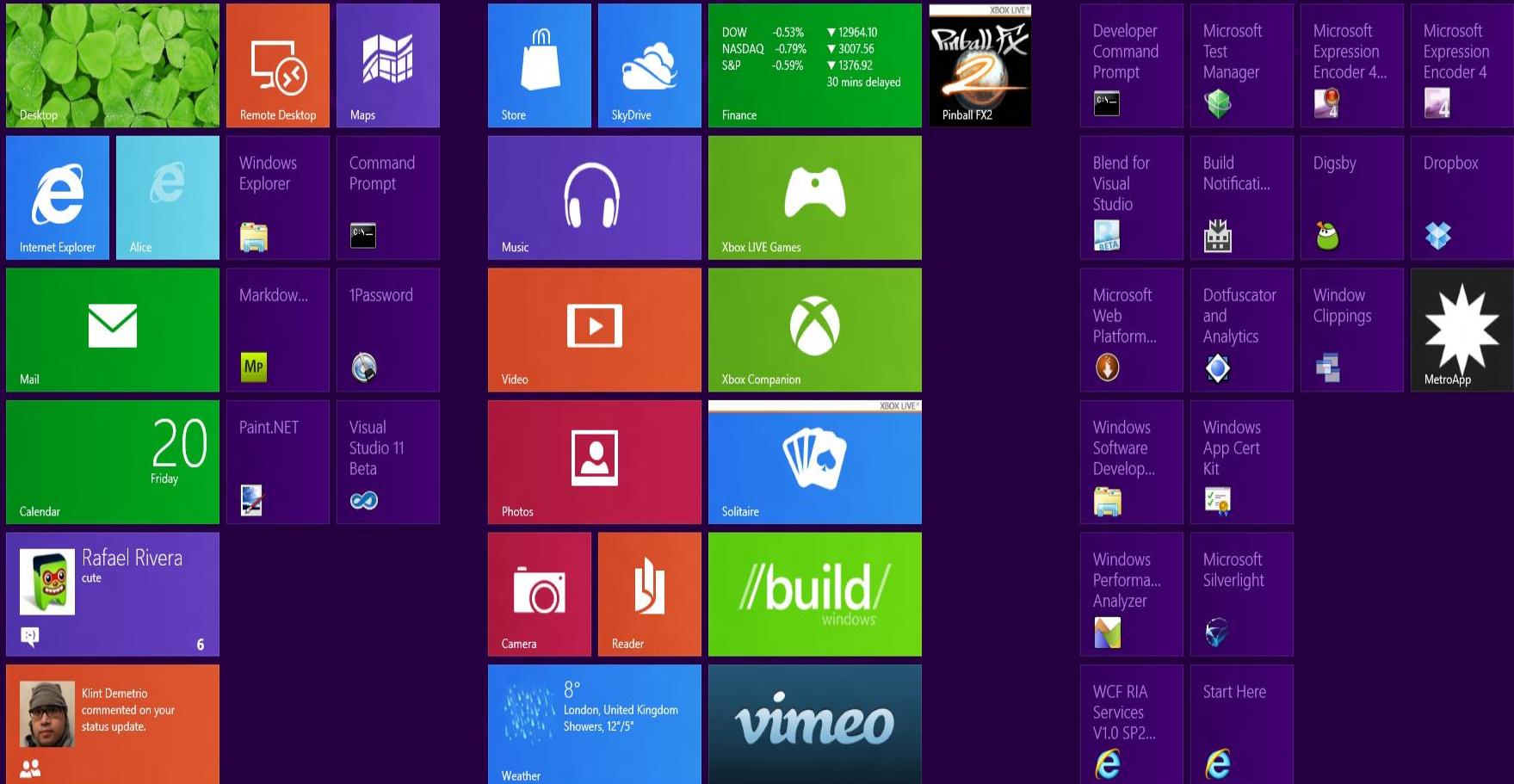


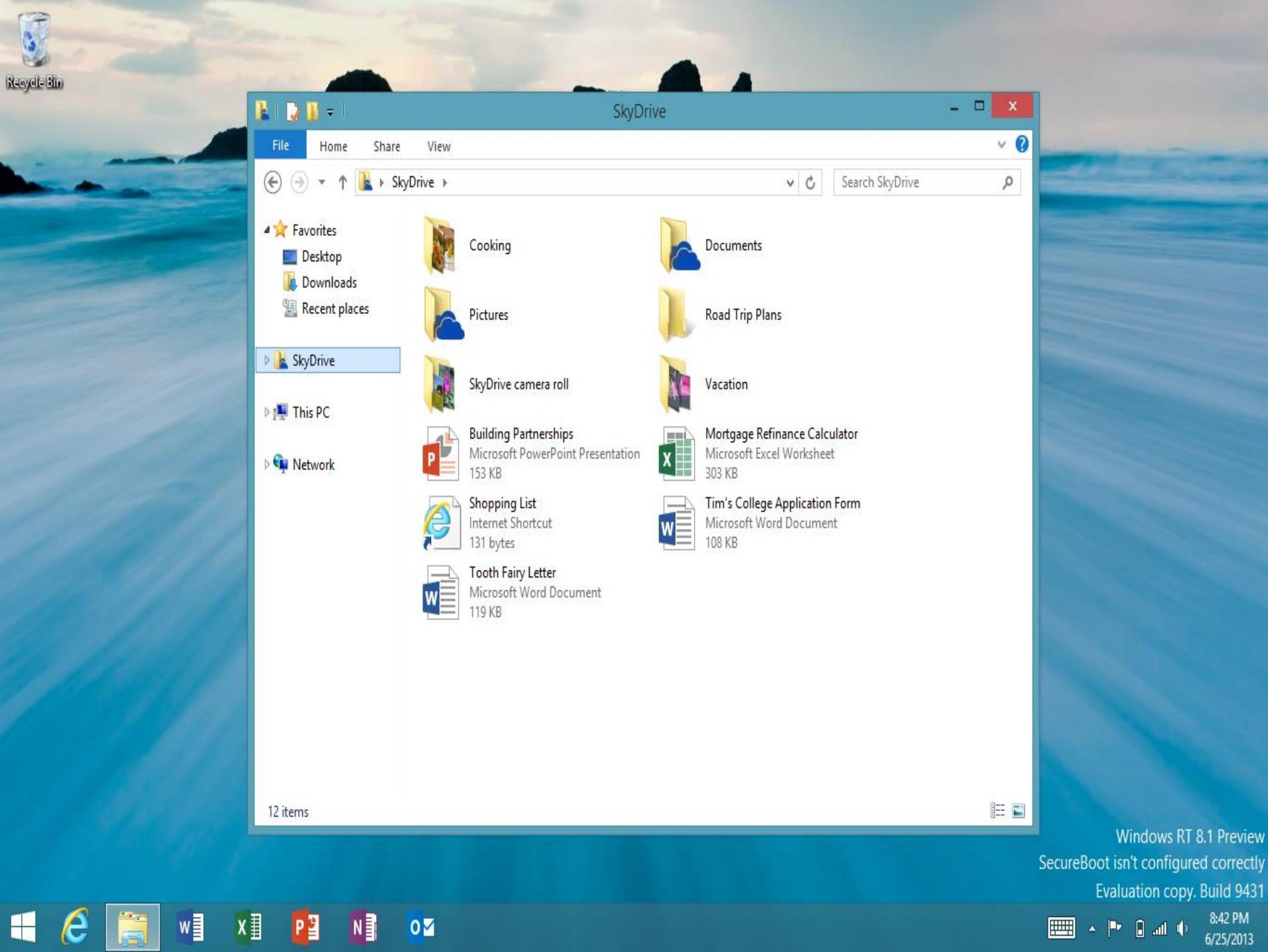
How many  
Outlook icons  
do you see?



# Start

DrPizza 





Windows RT 8.1 Preview

SecureBoot isn't configured correctly

Evaluation copy, Build 9431



8:42 PM

6/25/2013



Recycle Bin



Welcome to  
Tech Preview

Antonio Onio



Documents

Pictures

PC settings

File Explorer

Snipping Tool

Calculator

Paint

Sticky Notes

Games

All Apps



Store

72°  
Los Angeles  
Sunny  
81°/62°



Windows Technical Preview  
Evaluation copy. Build 9841



10:24 AM  
9/30/2014

# Nội dung chính

1

Lịch sử Windows

2

**Đặc điểm môi trường Windows**

3

Lập trình hướng sự kiện

4

.NET Framework

5

Visual Studio

# Môi trường Windows

- Hệ điều hành 32/64 bit
- Giao diện người dùng kiểu đồ họa (GUI)
- Đa nhiệm
- Quản lý bộ nhớ
- Tư tưởng hướng đối tượng
- Giao diện đồ họa độc lập thiết bị
- Kiến trúc hướng sự kiện, dựa vào thông điệp

# Giao diện người dùng kiểu đồ họa (GUI)

- Visual Interface
- WYSIWYG (What you see is what you get)
- Các ứng dụng Windows có giao diện thống nhất với cùng dáng vẻ:
  - một cửa sổ hình chữ nhật
  - thanh tiêu đề
  - menu, hộp hội thoại, thanh trượt

# Đa nhiệm

- Nhiều chương trình có thể được kích hoạt và chạy cùng một lúc
- Mỗi chương trình chiếm một phần tài nguyên của hệ thống và có một phần bộ nhớ thường trú
- Windows 32 bits hoạt động theo cơ chế preemptive. Mỗi chương trình có thể được tách thành các thread thực hiện cùng lúc

# Quản lý bộ nhớ

- Mã chương trình và dữ liệu có thể dịch chuyển trong bộ nhớ vật lý
- Khả năng sử dụng bộ nhớ phụ
- Các thư viện liên kết động (.DLL)

# Tư tưởng hướng đối tượng

- Cửa sổ ứng dụng chính và các cửa sổ con là các thể hiện (đối tượng) của lớp cửa sổ được đăng ký trước

# Giao diện đồ họa độc lập thiết bị

- Giao diện đồ họa của Window cho phép tạo các đối tượng đồ họa và định dạng văn bản cho cả màn hình và máy in
- Chương trình viết cho Windows không điều khiển trực tiếp thiết bị xuất như màn hình hay máy in mà thông qua một ngôn ngữ lập trình đồ họa gọi là Giao Diện Thiết Bị Đồ Họa (Graphics Device Interface: GDI)

# Kiến trúc hướng thông điệp (message driven)

- Windows và các ứng dụng Windows hoạt động theo cơ chế truyền, nhận thông điệp
- Các hoạt động của chương trình thay đổi tùy theo thông điệp mà nó nhận được,
- Thông điệp được gởi qua lại giữa ứng dụng và Windows, giữa các ứng dụng với nhau
- Cửa sổ ứng dụng tự động được vẽ lại mỗi khi có sự thay đổi kích thước hay vùng bị che. Điều này được thực hiện nhờ hệ điều hành gởi thông điệp cho chương trình

# Các loại ứng dụng cơ bản

- **Windows Application:** ứng dụng 32 bits, giao diện đồ họa
- **Win32 DLL:** ứng dụng 32 bits, dạng thư viện liên kết động (Dynamic-linked library)
- **Win32 LIB:** ứng dụng 32 bits, dạng thư viện liên kết tĩnh (Static-linked library)
- **Console Application:** ứng dụng 32 bits, với giao diện dạng DOS command line
- **Assembly Exe và Assembly DLL:** Ứng dụng và DLL chạy trên môi trường .NET (chứa mã trung gian)

# Windows Application dựa C

Năm	Ngôn ngữ	Công cụ
1985	C	Windows application programming interface (API)
1992	C++	Microsoft Foundation Class (MFC) Library
2001	C# hoặc C++.Net	Windows Form (1 phần của .NET Framework)
2006	C#, XAML	Windows Presentation Foundation (WPF)

# Các thư viện lập trình

- Windows API
- OWL - Object Windows Library
- VCL – Visual Component Library
- MFC - Microsoft Foundation Classes
- ATL - Active Template Library
- WTL - Windows Template Library
- BCL – Base Class Library
- STL – Standard Template Library
- TOM – Text Object Model
- ...

# Windows và DOS

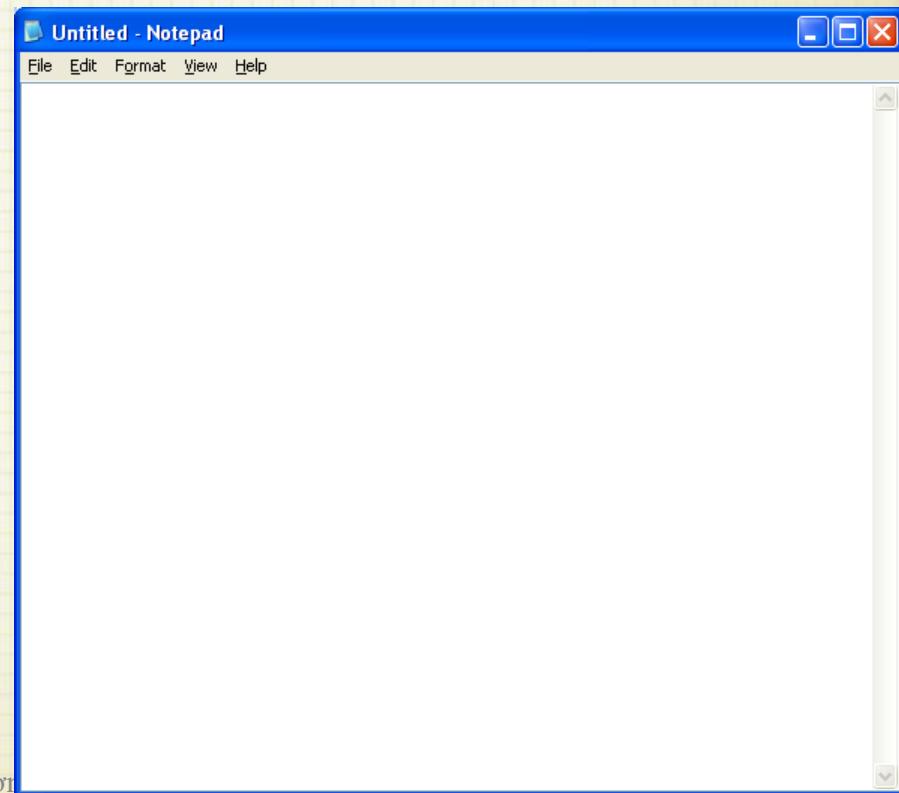
Windows	DOS
Lập trình sự kiện, dựa vào thông điệp (message)	Thực hiện tuần tự theo chỉ định
Multi-tasking	Single task
Multi-CPU	Single CPU
Tích hợp sẵn Multimedia	Phải dùng các thư viện Multimedia riêng
Hỗ trợ 32 bits hay hơn nữa	Ứng dụng 16 bits
DLL, OLE, DDE, COM, OpenGL, DirectX	Không có

# Giao diện đồ họa

- Các dạng ứng dụng giao diện đồ họa
  - SDI - Single Document Interface
  - MDI - Multiple Document Interface
  - Dialog

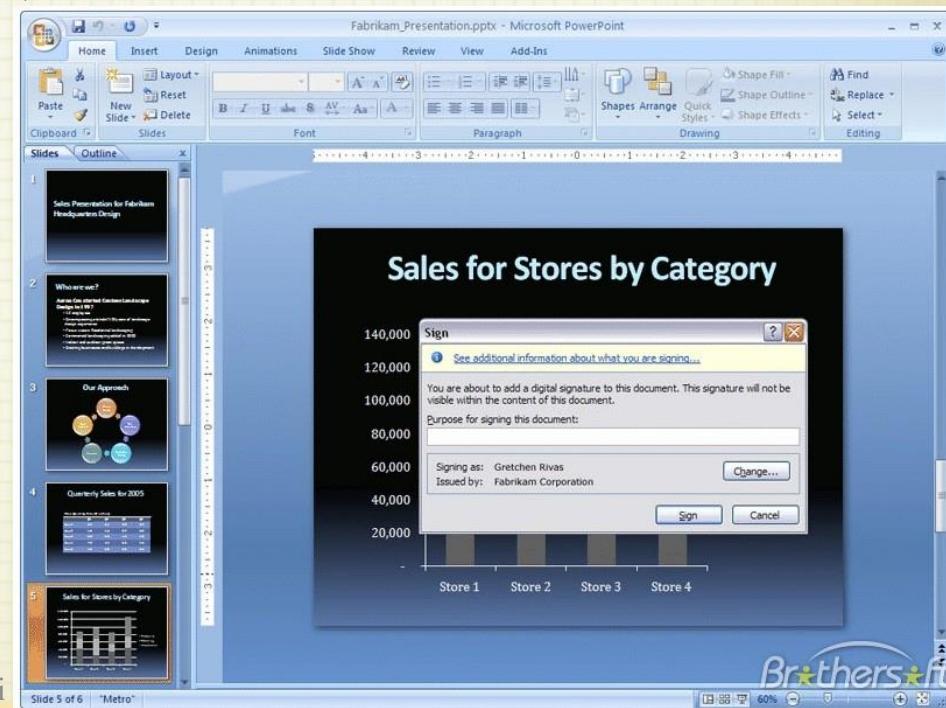
# SDI - Single Document Interface

- Một cửa sổ làm việc
- Cho phép thay đổi kích thước cửa sổ (Resizeable)
- Không có các cửa sổ con
- Ví dụ: Notepad, Paint,...



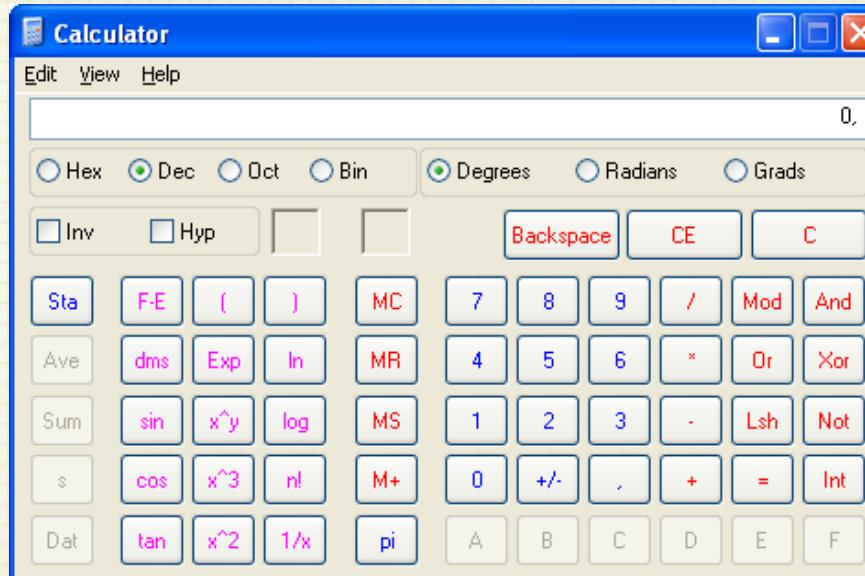
# MDI - Multiple Document Interface

- Một cửa sổ làm việc chính và nhiều cửa sổ con
- Cho phép thay đổi kích thước cửa sổ (Resizeable)
- Cho phép Maximize/Minimize/Close các cửa sổ con
- Ví dụ: Word, Excel, VC++,...



# Dialog

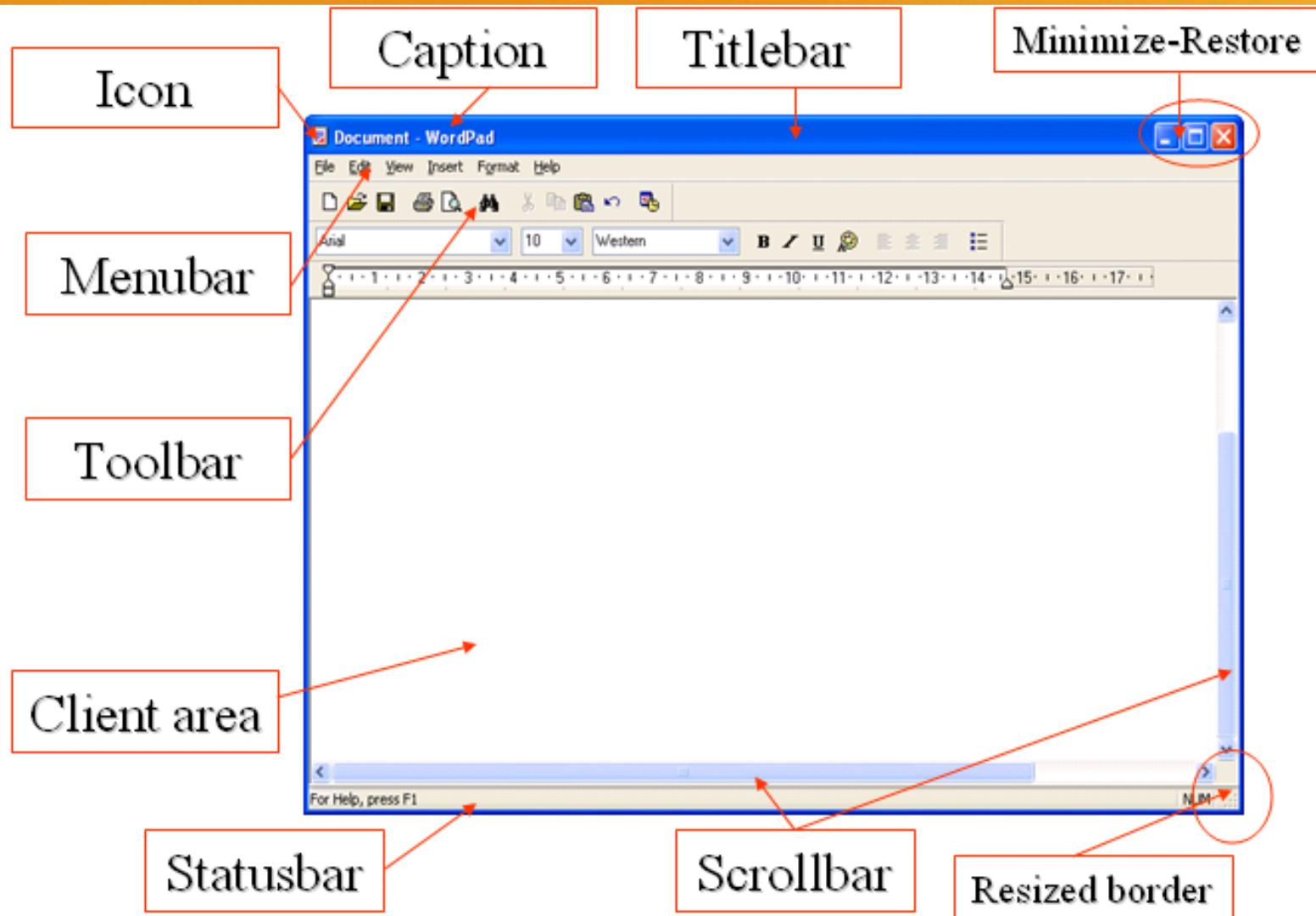
- Một cửa sổ làm việc
- Thường kích thước cố định
- Thường có các button, edit box, list box,...
- Ví dụ: Calculator, CD Player,..



# Cửa sổ - Window

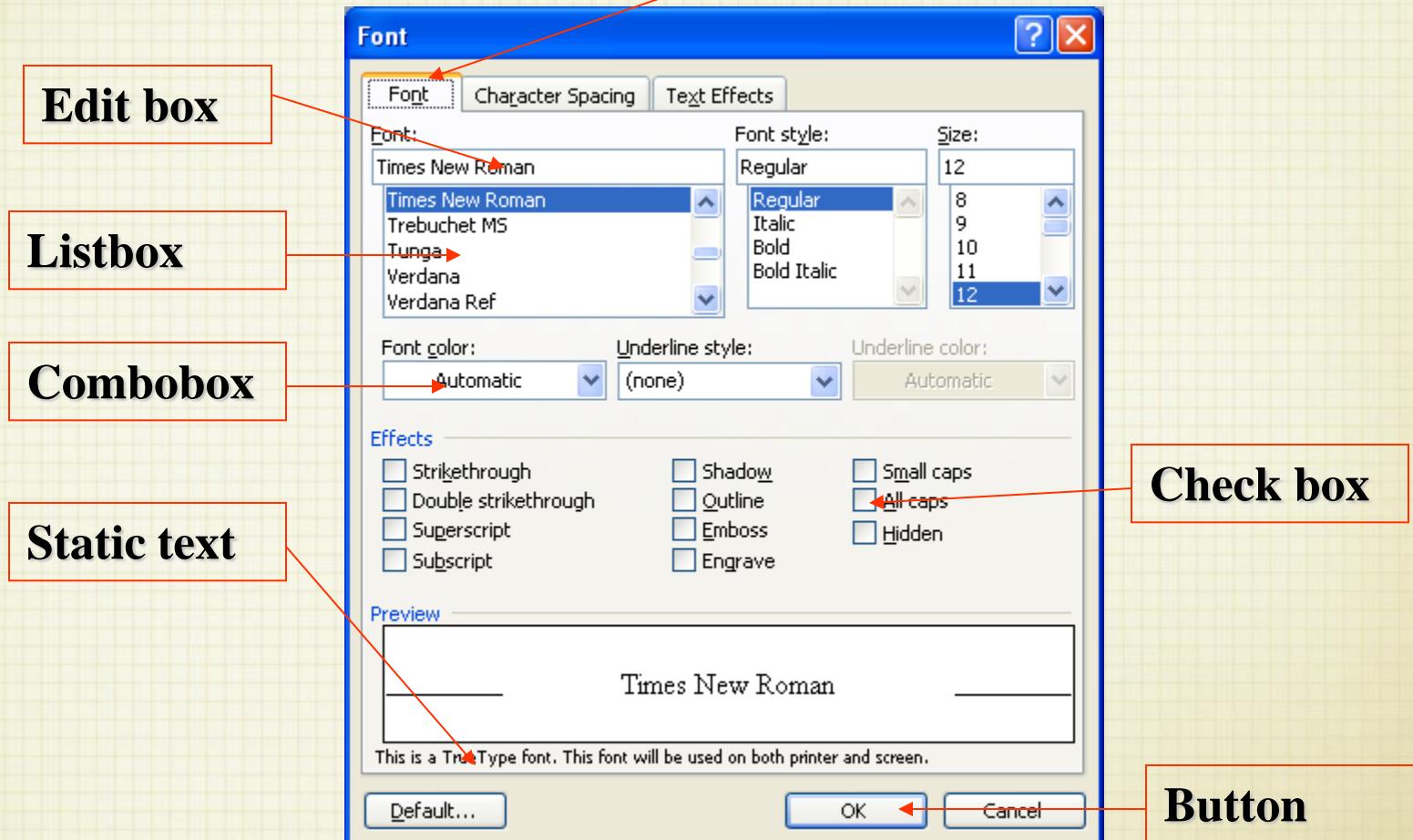
- Là một vùng chữ nhật trên màn hình dùng để hiển thị kết quả output và nhận các input từ người dùng
  - Công việc đầu tiên của 1 ứng dụng GUI là tạo một cửa sổ làm việc
- Mỗi cửa sổ đều có 1 cửa sổ cha (parent window), ngoại trừ cửa sổ nền Desktop
  - Tại mỗi thời điểm, chỉ có một cửa sổ nhận input từ người dùng (Active/Focused window)

# Các thành phần của một cửa sổ

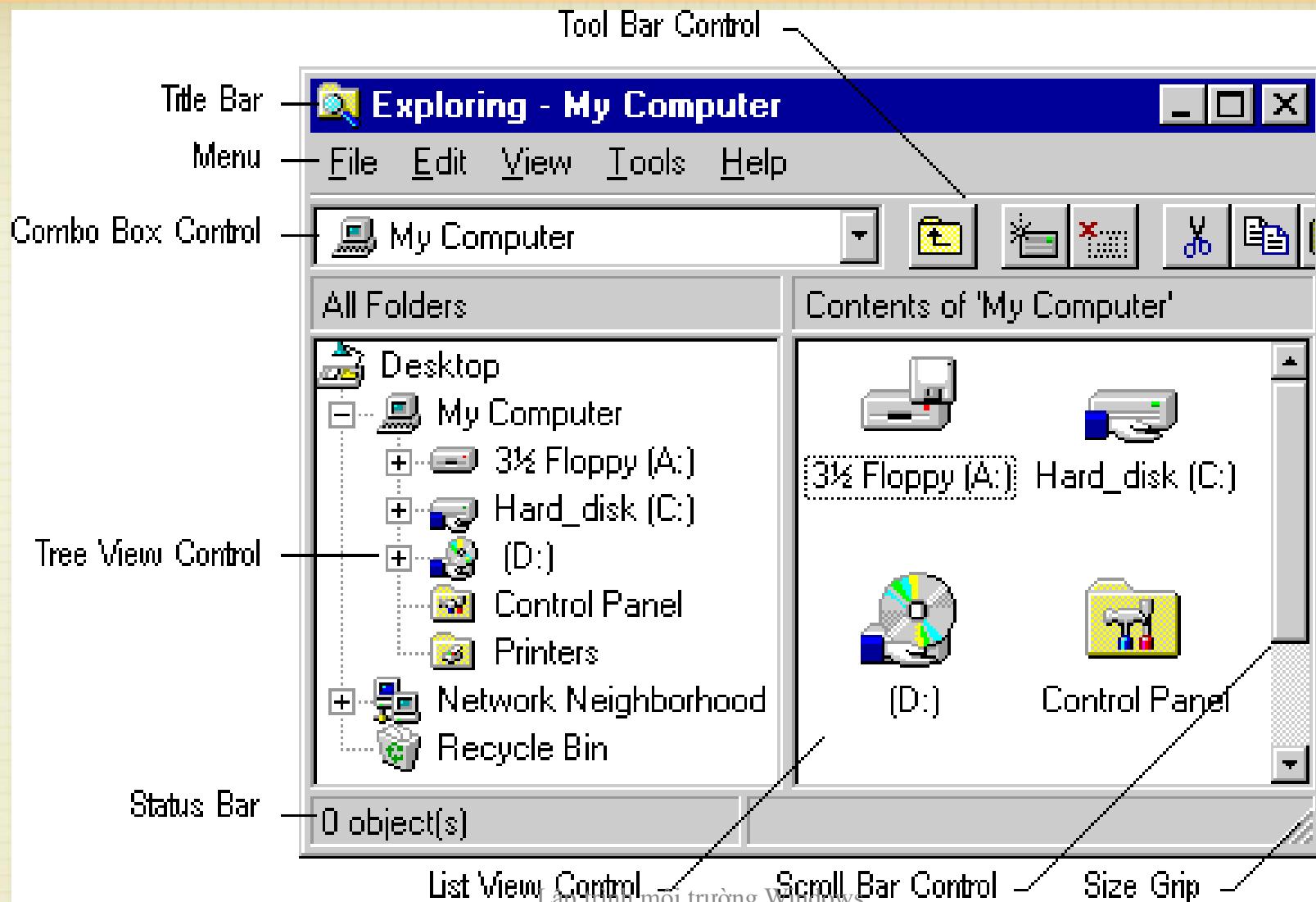


# Các control thông dụng

## Tab Control



# Các control thông dụng



# Nội dung chính



Lịch sử Windows



Đặc điểm môi trường Windows



**Lập trình hướng sự kiện**



.NET Framework

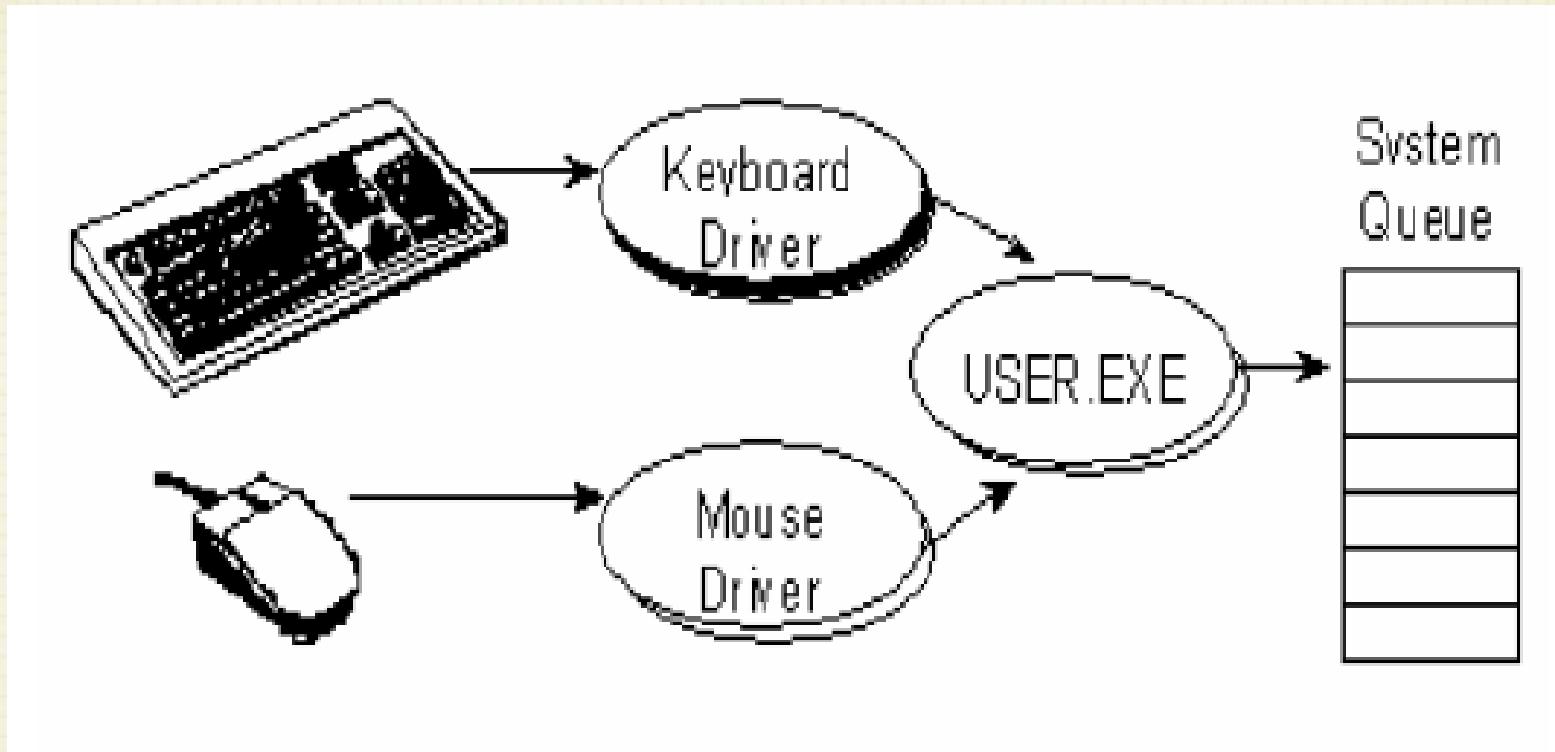


Visual Studio

# Lập trình hướng sự kiện

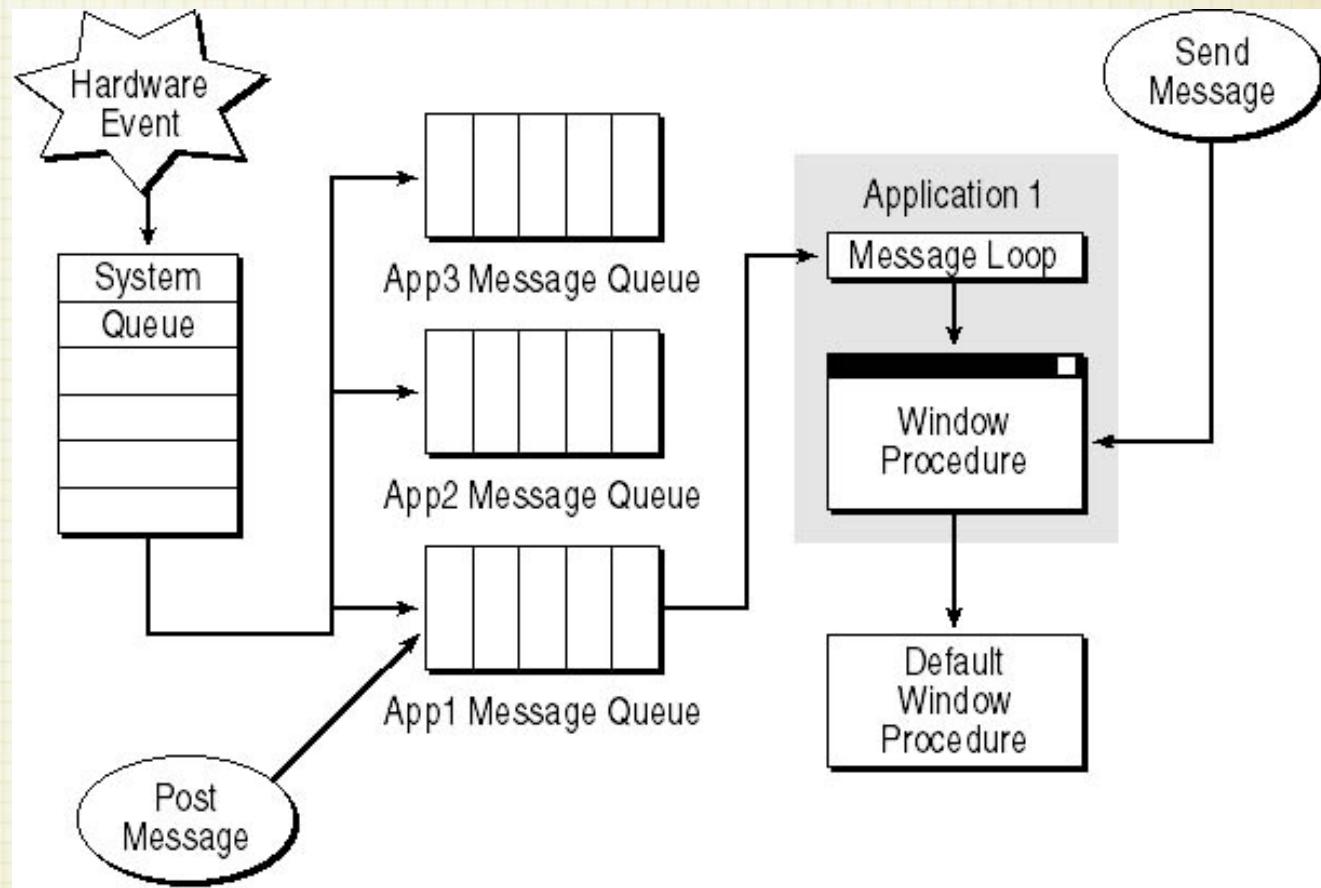
- Qui trình xử lý thông điệp
- Các loại thông điệp thường gặp
- Vòng lặp xử lý thông điệp

# Phát sinh các sự kiện và thông điệp



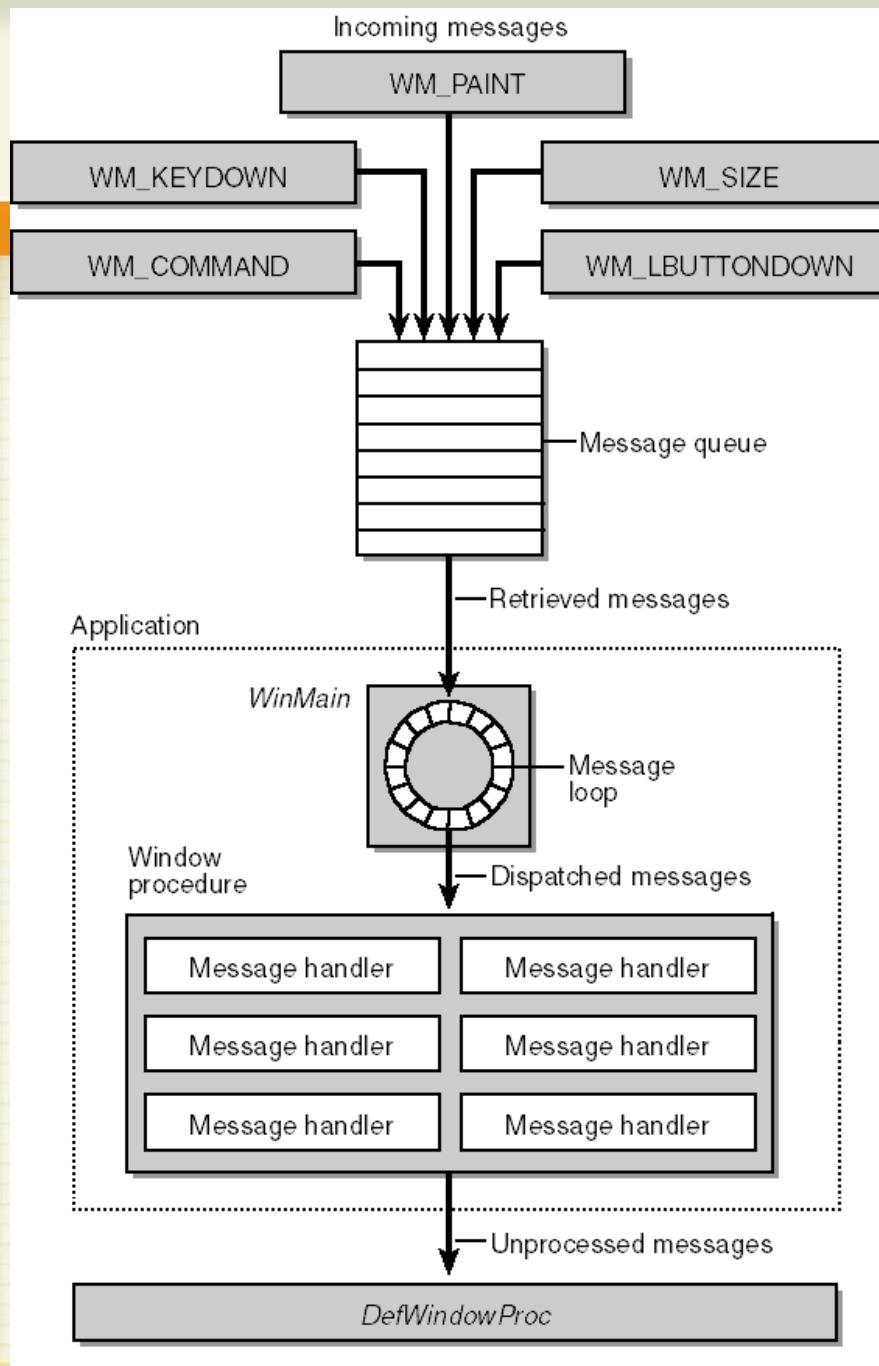
# Hàng đợi thông điệp

- Luồng điều khiển của chương trình được xác định dựa vào hành động của người dùng khi sử dụng chương trình



# Thủ tục cửa sổ

- Hệ điều hành ra lệnh cho ứng dụng nhờ thủ tục cửa sổ (Window Procedure)
- Hàm cửa sổ cho biết phản ứng của chương trình với các tác động bên ngoài như user input...
- Hệ điều hành Windows gọi thông điệp cho một chương trình ứng dụng bằng cách gọi hàm cửa sổ của ứng dụng đó, với tham số là thông điệp



# Windows Application trên C (Chương trình chính)

```
LRESULT CALLBACK WndProc (HWND, UINT, WPARAM, LPARAM);

int WINAPI WinMain (HINSTANCE hInstance, HINSTANCE hPrevInstance, PSTR szCmdLine, int iCmdShow)
{
    static TCHAR szAppName[] = TEXT ("...");

    HWND      hwnd;
    MSG       msg;
    WNDCLASS  wndclass;

    wndclass.style          = CS_HREDRAW | CS_VREDRAW;
    wndclass.lpfnWndProc   = WndProc;
    ...
    wndclass.lpszClassName = szAppName;

    if (!RegisterClass (&wndclass))
    {
        MessageBox (NULL, TEXT ("..."), ...);
        return 0;
    }
    hwnd = CreateWindow (szAppName,...,NULL);

    ShowWindow (hwnd, iCmdShow);
    UpdateWindow (hwnd);

    while (GetMessage (&msg, NULL, 0, 0))
    {
        TranslateMessage (&msg);
        DispatchMessage (&msg);
    }
    return msg.wParam;
}
```

Khai báo cửa sổ

Đăng ký  
cửa sổ

Vòng lặp bắt thông điệp

# Windows Application trên C (Hàm xử lý thông điệp)

```
LRESULT CALLBACK WndProc (HWND hwnd, UINT message, WPARAM wParam, LPARAM lParam)
{
    HDC      hdc ;
    PAINTSTRUCT ps ;
    RECT     rect ;

    switch (message)
    {
        case WM_CREATE:           → Thông điệp nhận được
            ...
            return 0 ;           → Xử lý thông điệp

        case WM_PAINT:
            ...
            return 0 ;

        case WM_DESTROY:
            ...
            return 0 ;

        ...
    }
    return DefWindowProc (hwnd, message, wParam, lParam) ;
}
```

# Một số thông điệp thường gặp

Thông điệp	Được gửi khi
<b>WM_CHAR</b>	Có một ký tự được nhập từ bàn phím
<b>WM_COMMAND</b>	Người sử dụng click một mục trên menu, button...
<b>WM_CREATE</b>	Một cửa sổ được tạo ra
<b>WM_DESTROY</b>	Một cửa sổ bị huỷ.
<b>WM_LBUTTONDOWN</b>	Phím trái chuột được nhấn
<b>WM_LBUTTONUP</b>	Phím trái chuột được thả ra
<b>WM_MOUSEMOVE</b>	Con trỏ chuột di chuyển trên màn hình
<b>WM_PAINT</b>	Cửa sổ cần vẽ lại
<b>WM_QUIT</b>	Ứng dụng sắp kết thúc
<b>WM_SIZE</b>	Cửa sổ có thay đổi kích thước

# Windows Application trên C# (Chương trình chính)

```
using System;
using System.Windows.Forms;
namespace NotepadForms
{
    public class Program
    {
        [STAThread]
        static void Main()
        {
            Form myForm = new MyForm(); → Khai báo cửa sổ
            Application.Run(new MyForm()); → Khởi động vòng lặp thông điệp
        }
    }
}
```

# Windows Application trên C# (Xử lý thông điệp)

```
public class MyForm : System.Windows.Forms.Form
{
    public MyForm()
    {
        this.Paint += new System.Windows.Forms.PaintEventHandler
            (this.MainForm_Paint);
    }

    private void MainForm_Paint (object sender,
                                System.Windows.Forms.PaintEventArgs evt)
    {
        ...
    }
}
```

**Gán sự kiện**

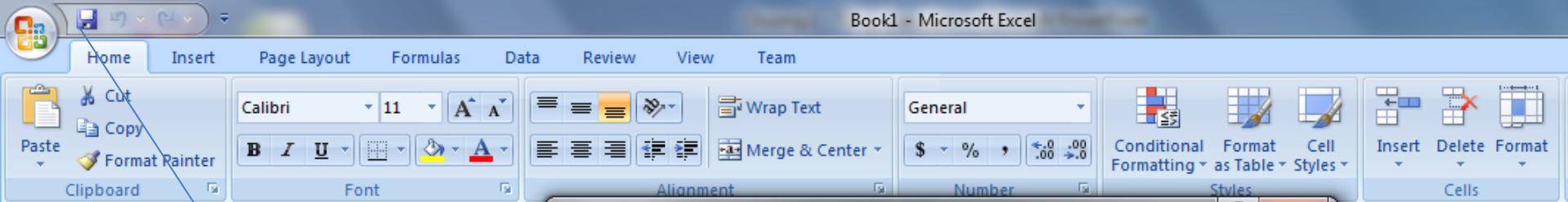
**Xử lý sự kiện**

# Tài nguyên - Resource

- Tài nguyên (Menu bar, dialog, bitmap, icon, cursor,...) của ứng dụng được định nghĩa bên ngoài và được thêm vào trong file thực thi của ứng dụng khi biên dịch
- Các loại tài nguyên của ứng dụng
  - Accelerator table: bảng mô tả phím tắt (hot-key)
  - Bitmap: ảnh bitmap
  - Caret: con trỏ văn bản
  - Cursor: con trỏ chuột
  - Dialog box: khung hộp thoại (Dialogbox Template)
  - Font: font chữ

# Tài nguyên - Resource

- Các loại tài nguyên của ứng dụng
  - Icon: biểu tượng
  - Menu: menu
  - String-table entry: bảng mô tả các chuỗi ký tự
  - Version information: bảng mô tả thông tin “phiên bản”



A1

Resource

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

Sheet1 Sheet2 Sheet3

9/4/2024

62

Lập trình môi trường Windows

The screenshot shows a Microsoft Excel window with the 'Format Cells' dialog box open over a worksheet. The dialog box is titled 'Format Cells' and has tabs for Number, Alignment, Font, Border, Fill, and Protection. The 'Font' tab is selected, displaying the 'Font' dropdown set to 'Calibri', 'Font style' set to 'Regular', and 'Size' set to '11'. Other options include 'Underline' (None), 'Effects' (Strikethrough, Superscript, Subscript), and a 'Color' picker. A preview window shows the text 'AaBbCcYyZz' in the chosen font. A note at the bottom states: 'This is a TrueType font. The same font will be used on both your printer and your screen.' At the bottom right of the dialog are 'OK' and 'Cancel' buttons. The background shows a grid of cells labeled A1 through E27. A red arrow points from the word 'Resource' in the main area to the 'Font' dropdown in the ribbon's Font group.

# Nội dung chính

1

Lịch sử Windows

2

Đặc điểm môi trường Windows

3

Lập trình hướng sự kiện

4

.NET Framework

5

Visual Studio

# Giới thiệu .NET

- .NET bao gồm một bộ đầy đủ các công cụ giúp tạo và cài đặt các ứng dụng, .NET bao gồm:
  - Sản phẩm của .NET
    - Visual studio.NET IDE, Expression Blend
    - Ngôn ngữ lập trình: C#, VB.NET, J#, XAML...
  - Dịch vụ web của .NET
  - .NET Framework

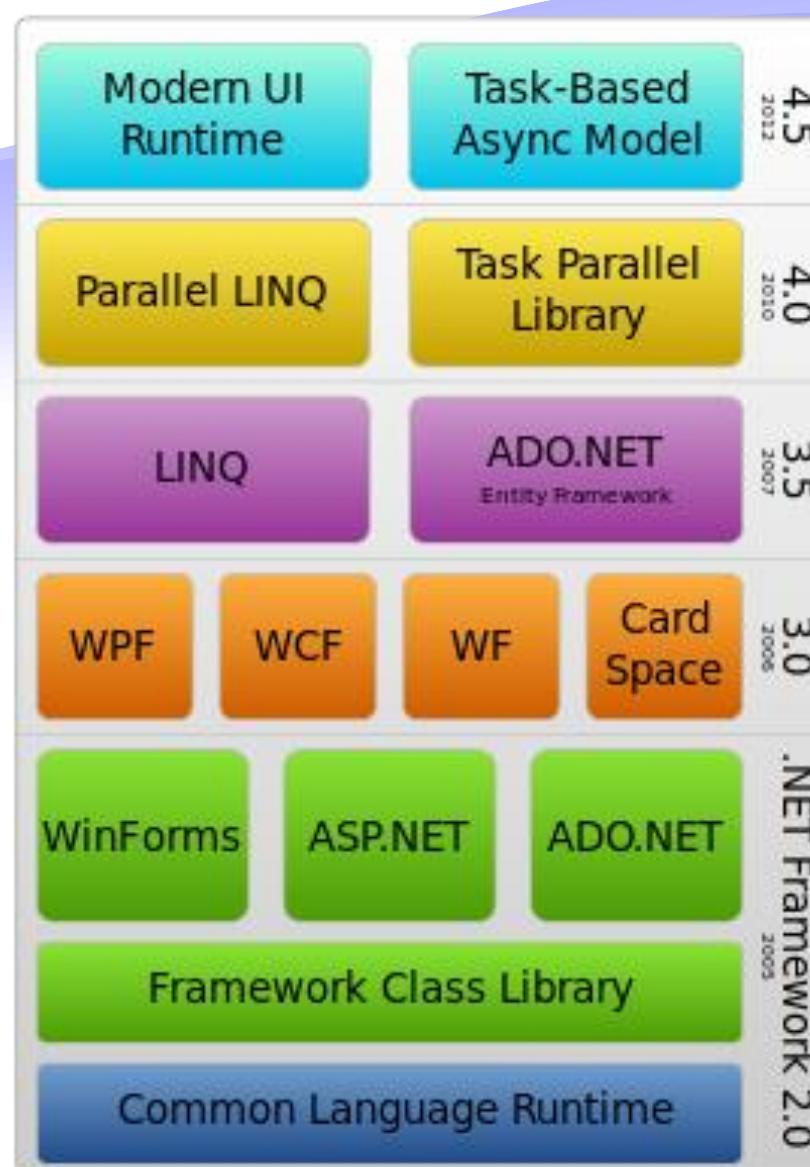
# .NET Framework

- Tương tác giữa các thành phần trong và ngoài .NET Framework
- Common Language Runtime Engine
- Độc lập ngôn ngữ
- Thư viện lớp cơ sở
- Quản lý phiên bản
- Bảo mật
- Tính di động (portable)

# Kiến trúc .NET Framework

- .NET framework bao gồm 4 thành phần chủ yếu
  - Common Language Runtime (CLR)
  - Các lớp cơ sở của .NET Framework
  - Data Management
  - Giao diện người dùng

# .NET Framework



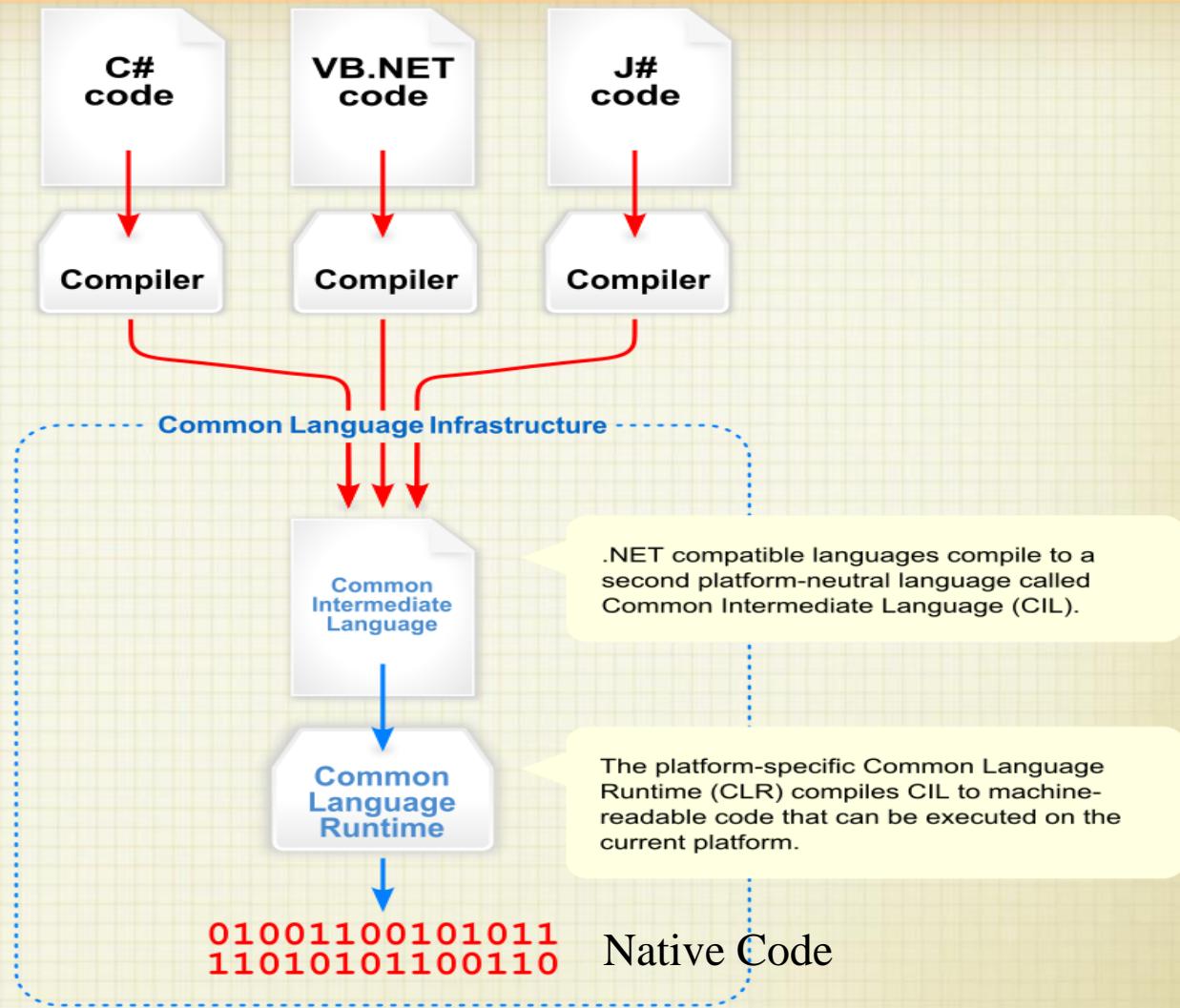
# CLR

- Common Language Runtime là môi trường để cho tất cả các ứng dụng viết trên .NET chạy
- CLR hỗ trợ nhiều ngôn ngữ và cung cấp các công cụ dùng chung cho các ngôn ngữ lập trình khác nhau, giúp cho việc tương tác qua lại giữa các ngôn ngữ lập trình khác nhau dễ dàng hơn

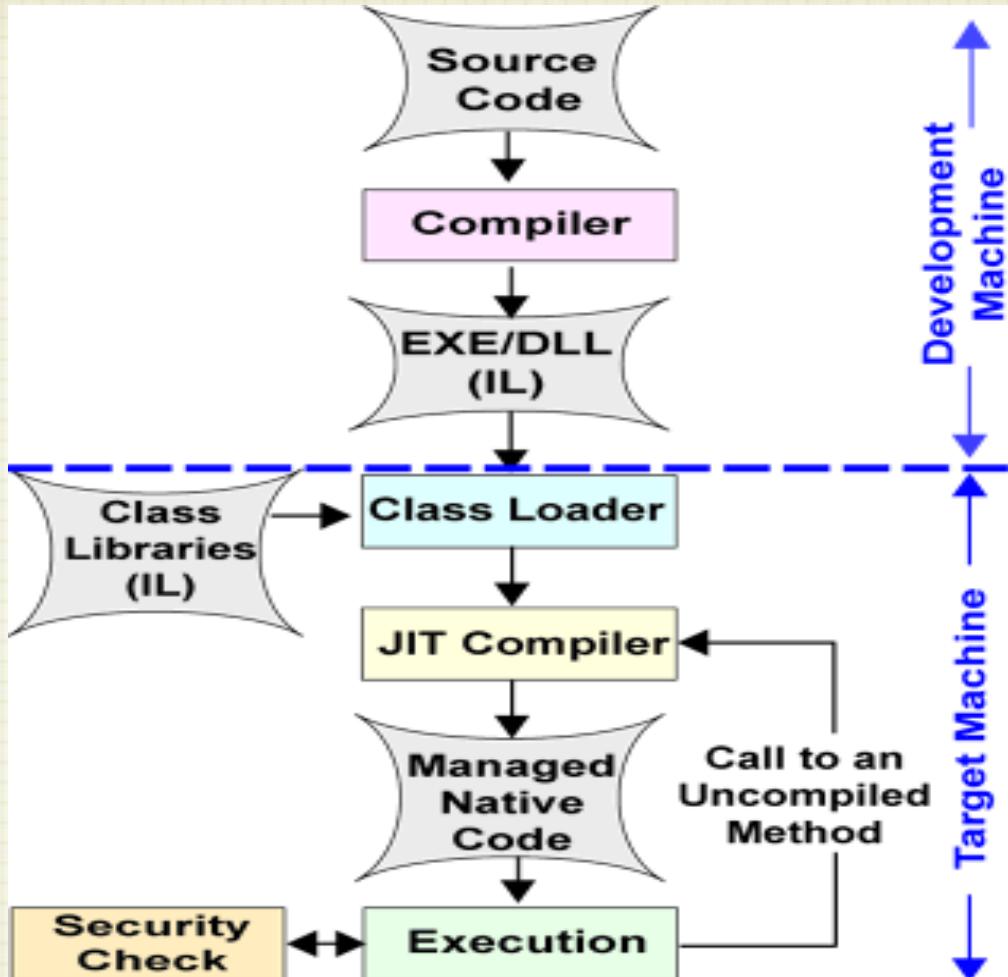
# Đặc điểm của CLR

- Tự động quản lý bộ nhớ
- Hệ thống kiểu dữ liệu dùng chung giữa các ngôn ngữ lập trình
- Tính tương tác giữa các ngôn ngữ
- Độc lập với cấu trúc phần cứng bên dưới
- Cơ chế bảo mật hệ thống
- An toàn kiểu dữ liệu

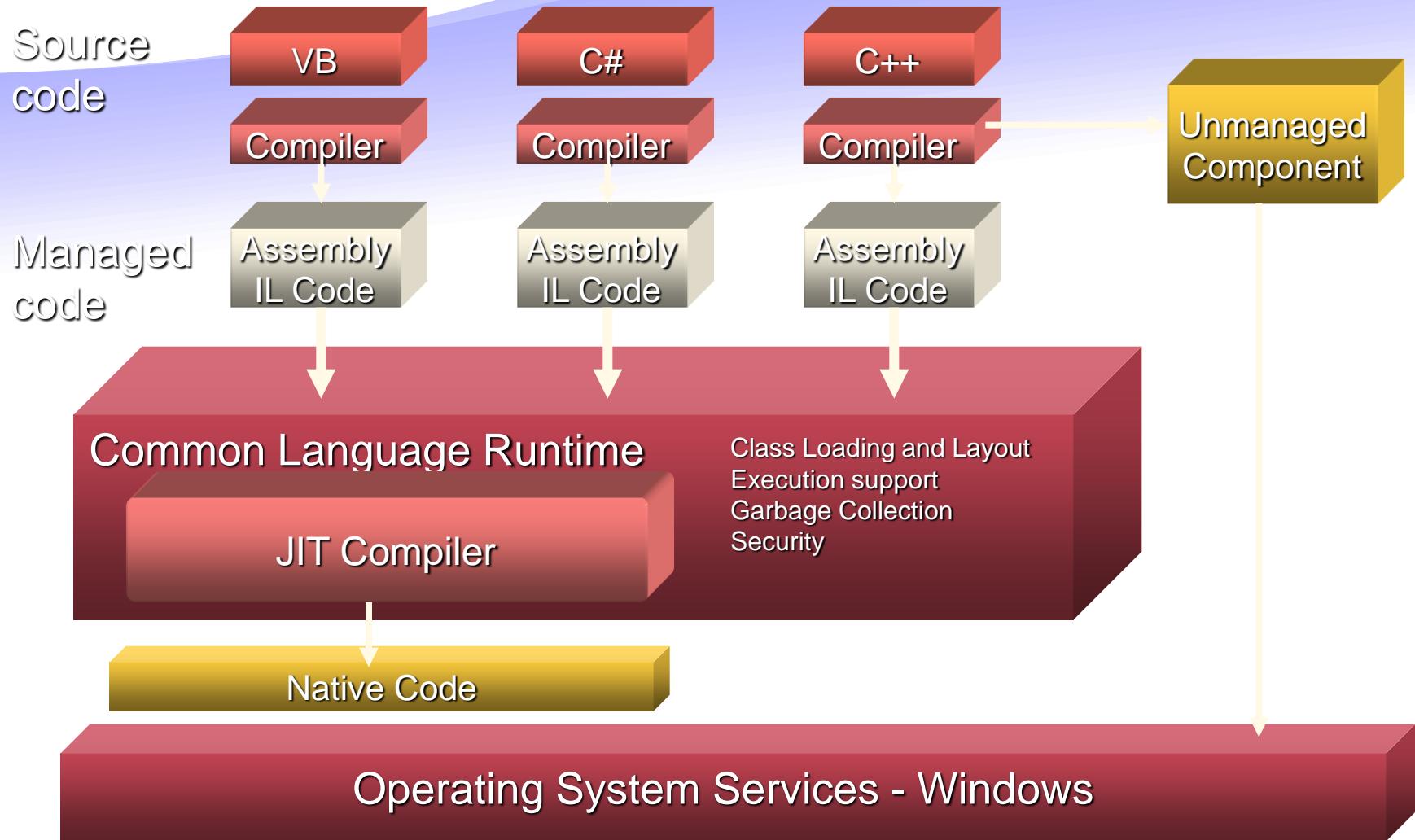
# CLR



# Quá trình biên dịch



# CLR Execution Model



# Các lớp cơ sở của .NET framework

- Được xây dựng theo phương pháp hướng đối tượng
- Cung cấp các lớp dùng để xử lý các vấn đề thường gặp phải khi phát triển ứng dụng
- Được dùng chung cho nhiều ngôn ngữ lập trình khác nhau
- Được tổ chức thành những namespace được lưu trữ trong assembly.

# Các lớp cơ sở của .NET framework

- Namespace dùng để nhóm các lớp và các interface có mối quan hệ luận lý với nhau.
- Các namespace có thể được dùng trong bất cứ ngôn ngữ nào tương thích với .NET.
- Namespace được sử dụng để hạn chế việc dùng cùng một tên cho 2 hay nhiều lớp có mục đích sử dụng khác nhau

# Các lớp cơ sở của .NET framework

- Assembly là một đơn vị phần mềm có chứa đầy đủ các thông tin về các lớp hiện thực, các cấu trúc và các interface để hiện thực ứng dụng
- Assembly lưu trữ các thông tin để mô tả chính nó, thông tin này gọi là meta data

# Các lớp cơ sở của .NET framework



# Các lớp cơ sở của .NET framework

- System.Drawing
- System.Data
- System.Windows.Forms
- System.Web.UI
- System.Web.Services
- .....

# Ưu điểm của .NET Framework

- Dùng chung cho nhiều ngôn ngữ lập trình
- Tạo ứng dụng độc lập với môi trường phần cứng bên dưới
- Cho phép viết ứng dụng bằng nhiều ngôn ngữ tương thích với .NET
- Tự động quản lý tài nguyên
- Dễ dàng triển khai ứng dụng

# Nội dung chính



Lịch sử Windows



Đặc điểm môi trường Windows



Lập trình hướng sự kiện



.NET Framework



Visual Studio

# Visual studio.NET IDE

- Visual Studio .NET Integrated Development Environment (IDE) cung cấp một giao diện dùng chung để phát triển nhiều loại dự án khác nhau.
- Cung cấp môi trường để thiết kế, viết code, kiểm tra và sửa lỗi khi phát triển ứng dụng
- Các thành phần của Visual Studio .NET IDE
  - Dự án và giải pháp
  - Giao diện của các thành phần

WindowsFormsApplication2 (Running) - Microsoft Visual Studio

FILE EDIT VIEW PROJECT BUILD DEBUG TEAM TOOLS TEST ANALYZE WINDOW HELP

Quick Launch (Ctrl+Q)

Anh Huynh

Process: [11052] WindowsFormsApplication2 Lifecycle Events Thread: Stack Frame:

Solution Explorer

Search Solution Explorer (Ctrl+Shift+F)

WindowsFormsApplication2

- Properties
- References
- lib
- App.config
- Form1.cs

  - Form1.Designer.cs
  - Form1.resx

- Form1
- Program.cs

Form1.cs [Design]

WindowsFormsApplication2 WindowsFormsApplication2.Form1 Form1()

```
using Aron.Sinoai.OfficeHelper;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApplication2
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
    }
}
```

Quản lý solution

Khung soạn thảo

Output

Show output from: Debug

```
'WindowsFormsApplication2.vshost.exe' (CLR v4.0.30319: WindowsFormsApplication2.vshost.exe): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC_32\System.Data\v4.0_4.0.0.0__b77a5c561934e089\System.Data.dll'
'WindowsFormsApplication2.vshost.exe' (CLR v4.0.30319: WindowsFormsApplication2.vshost.exe): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC_MSIL\System.Deployment\v4.0_4.0.0.0__b03f5f7f1d33970a\System.Deployment.dll'
'WindowsFormsApplication2.vshost.exe' (CLR v4.0.30319: WindowsFormsApplication2.vshost.exe): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC_MSIL\System.Xml\v4.0_4.0.0.0__b77a5c561934e089\System.Xml.dll'
The thread 0x2424 has exited with code 259 (0x103).
The thread 0x1594 has exited with code 259 (0x103).
'WindowsFormsApplication2.vshost.exe' (CLR v4.0.30319: WindowsFormsApplication2.vshost.exe): Loaded 'C:\Users\anhht\Downloads\Compressed\WindowsFormsApplication2\WindowsFormsApplication2.dll'
```

Xuất kết quả

Solution Explorer Team Explorer

Call Stack Breakpoints Command Window Immediate Window Output

Autos Locals Watch Threads

9/4/2024

Lập trình môi trường Windows

Ln12 Col35 Ch35

81

Ready

# Lịch sử Visual Studio

Product name	Codename	Internal version	Supported .NET Framework versions	Release date
Visual Studio	N/A	4.0	N/A	April 1995
Visual Studio 97	<i>Boston</i>	5.0	N/A	February 1997
Visual Studio 6.0	<i>Aspen</i>	6.0	N/A	June 1998
Visual Studio .NET (2002)	<i>Rainier</i>	7.0	1.0	February 13, 2002
Visual Studio .NET 2003	<i>Everett</i>	7.1	1.1	April 24, 2003
Visual Studio 2005	<i>Whidbey</i>	8.0	2.0, 3.0	November 7, 2005
Visual Studio 2008	<i>Orcas</i>	9.0	2.0, 3.0, 3.5	November 19, 2007
Visual Studio 2010	<i>Dev10/Rosario</i>	10.0	2.0, 3.0, 3.5, 4.0	April 12, 2010

# Lịch sử Visual Studio

Product name	Codename	Internal version	Supported .NET Framework versions	Release date
Visual Studio 2012	<i>Dev11</i>	11.0	2.0, 3.0, 3.5, 4.0, 4.5, 4.5.1, 4.5.2	September 12, 2012
Visual Studio 2013	<i>Dev12</i>	12.0	2.0, 3.0, 3.5, 4.0, 4.5, 4.5.1, 4.5.2	October 17, 2013
Visual Studio 2015	<i>Dev14</i>	14.0	2.0, 3.0, 3.5, 4.0, 4.5, 4.5.1, 4.5.2, 4.6	November 12, 2014
Visual Studio 2017	<i>Dev15</i>	15.0	2.0, 3.0, 3.5, 4.0, 4.5, 4.5.1, 4.5.2, 4.6, 4.6.1, 4.6.2, 4.7	March 7, 2017
Visual Studio 2019	<i>Dev16</i>	16.0	4.8, 4.7. 2, 4.7. 1, 4.7, 4.6. 2, 4.6. 1, 4.6, 4.5. 2, 4.5. 1, 4.5, 4.0.	April 2, 2019
Visual Studio 2022	<i>Dev17</i>	17.0	4.8.1, 4.8, 4.7.2, 4.7.1, 4.7, 4.6.2	November 8, 2021

# Tính năng các phiên bản

Product	Extensions	Projects templates	MSDN integration	Debugging	Profiling	Static analysis	IntelliTrace	Unit test	Code coverage	Coded UI test	Test impact analysis	Load testing	Lab management	Architecture and modeling	<a href="#">Windows Phone development</a>
Professional	Yes	Yes	Full or Essentials	Yes	Yes	No	No	Yes	No	No	No	No	No	No	Yes
Premium	Yes	Yes	Full	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	No	No	Read-only	Yes
Ultimate	Yes	Yes	Full	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Test Professional	No	No	Full	No	No	No	No	No	No	No	Yes	No	Yes	No	No

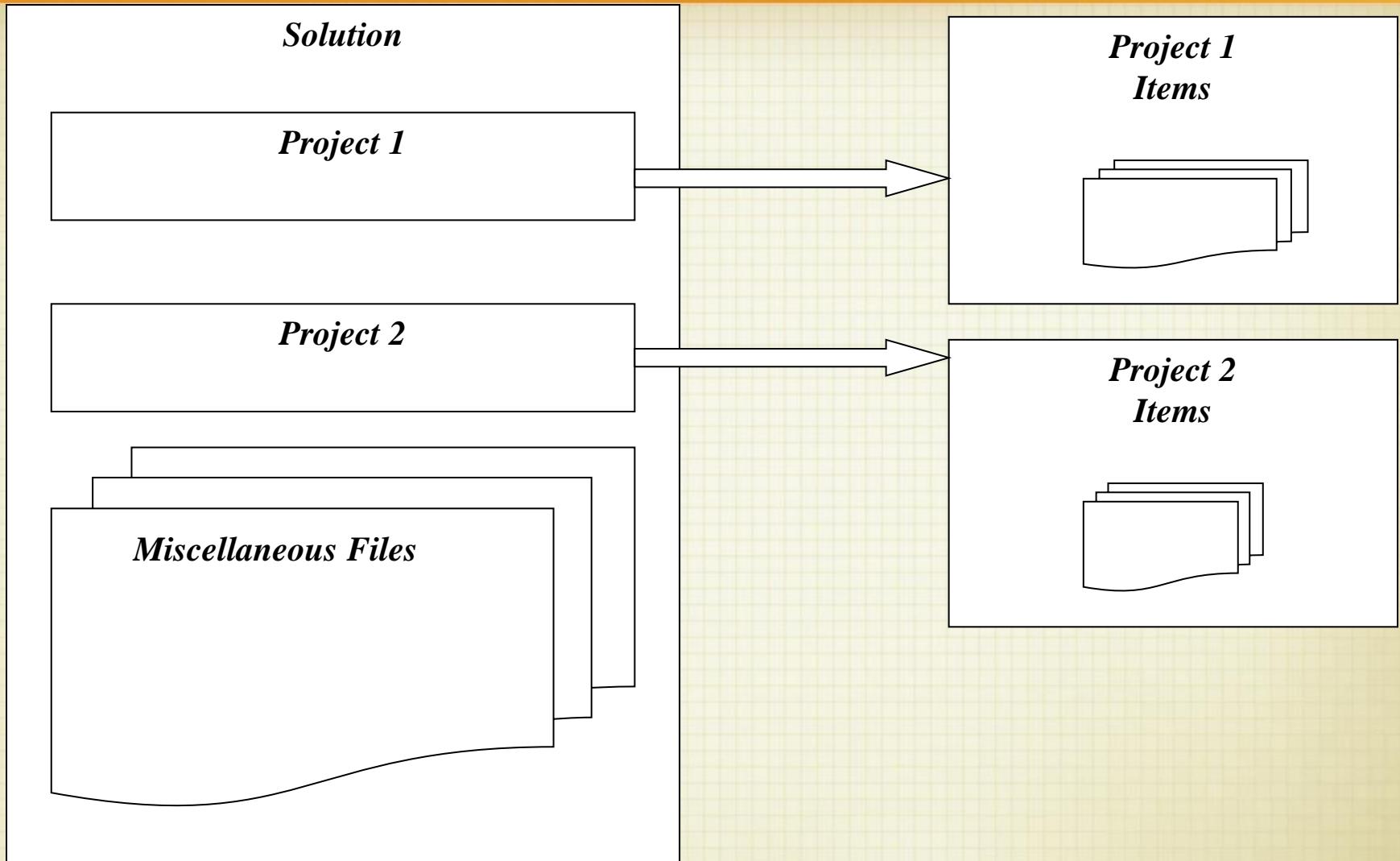
# Visual Studio 2022

- Develop cross-platform mobile and desktop apps with .NET MAUI. Build responsive Web UIs in C# with Blazor. Build, debug, and test .NET and C++ apps in Linux environments. Use hot reload capabilities across .NET and C++ apps. Edit running ASP.NET pages in the web designer view.
- Visual Studio offers AI-enhanced features like IntelliSense for context-aware code completions and IntelliCode, which leverages AI patterns from open-source code. Integrated GitHub Copilot provides AI-powered code completions, chat assistance, debugging suggestions, and unit test generation. These capabilities boost productivity and code quality, helping you code faster and smarter.

# Dự án và giải pháp

- Trong visual studio .NET ứng dụng được tạo thành từ nhiều phần tử ví dụ như file, thư mục
- Để tổ chức các phần tử trên .NET cung cấp dự án và giải pháp
- Dự án (project): cho phép xây dựng, dịch và sửa lỗi các thành phần tạo nên ứng dụng. Dự án được dịch ra thành file exe hoặc dll
- Giải pháp (Solution): Bao gồm một hay nhiều project

# Dự án và project



# Q & A

# Thank You !