

Software, consisting of programs, enables a computer to perform specific tasks, as opposed to its physical components (hardware) which can only do the tasks they are mechanically designed for.

5.1 Categories of Software

Two general categories of software are showed in Figure I-5-1.

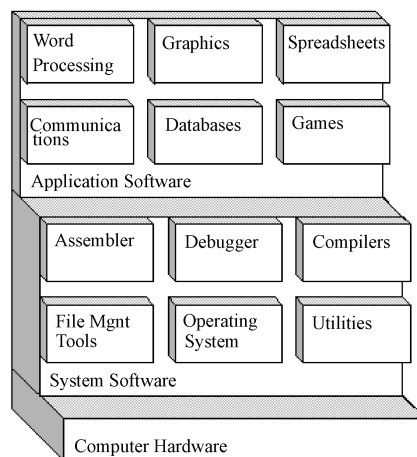


Figure I-5-1 Categories of Software

Other examples of software include:

- Programming languages define the syntax and semantics of computer programs. For example, Pascal, C, C++, VB/VB.NET, C#, Java, etc.
- Middleware controls and co-ordinates distributed systems. Middleware is computer software that connects software components or some people and their applications. The software consists of a set of services that allows multiple processes running on one or more machines to interact. This technology evolved to provide for interoperability in support of the move to coherent distributed architectures, which are most often used to

support and simplify complex distributed applications. It includes web servers, application servers, and similar tools that support application development and delivery. Middleware is especially integral to modern information technology based on XML, SOAP, Web services, and service-oriented architecture.

- Testware is software for testing hardware or a software package. Generally speaking, Testware is a sub-set of software with a special purpose, that is, for software testing, especially for software testing automation. Testware is produced by both verification and validation testing methods.
- Firmware is low-level software often stored on electrically programmable memory devices. Firmware is given its name because it is treated like hardware and run (“executed”) by other software programs. Typical examples of devices containing firmware range from end-user products such as remote controls or calculators, through computer parts and devices like hard disks, keyboards, or memory cards, all the way to scientific instrumentation and industrial robotics. Also more complex consumer devices, such as mobile phones, digital cameras, etc., contain firmware to enable the device’s basic operation as well as implementing higher-level functions.
- Device drivers control parts of computers such as disk drives, printers, CD drives, or computer monitors. A driver typically communicates with the device through the computer bus or communications subsystem to which the hardware connects. When a calling program invokes a routine in the driver, the driver issues commands to the device. Once the device sends data back to the driver, the driver may invoke routines in the original calling program. Drivers are hardware-dependent and operating-system-specific. They usually provide the interrupt handling required for any necessary asynchronous time-dependent hardware interface.

System software is a set of programs that interacts directly with hardware. System software is a generic term referring to any computer software which manages and controls the hardware so that application software can perform a task. System software serves as the interface between the user, the application software, and the computer’s hardware.

The most basic types of system software are:

- The computer BIOS and device firmware, which provide basic functionality to operate and control the hardware connected to or built into the computer;
- The operating system (prominent examples being Microsoft Windows, Mac OS X and Linux), which allows the parts of a computer to work together by performing tasks like transferring data between memory and disks or rendering output onto a display device. It also provides a platform to run high-level system software and application software;
- Utility programs, which help to analyze, configure, optimize and maintain the computer.

Application software, also called end-user programs, is a subclass of computer software that

employs the capabilities of a computer directly to a task that the user wishes to perform. This should be contrasted with system software which is involved in integrating a computer's various capabilities, but typically does not directly apply them in the performance of tasks that benefit the user.

E.g., database programs (Access), word processors (Word), and spreadsheets (Excel).

5.2 System Software

5.2.1 BIOS

BIOS (Basic Input/Output System) is the built-in software that determines what a computer can do without accessing programs from a disk. On PCs, the BIOS contains all the code required to control the keyboard, display screen, disk drives, serial communications, and a number of miscellaneous functions. The BIOS is typically placed in a ROM chip that comes with the computer (it is often called a ROM BIOS). This ensures that the BIOS will always be available and will not be damaged by disk failures. It also makes it possible for a computer to boot itself. In short, BIOS:

- Stored in the ROM chip;
- Contains the computer's startup instructions (ref. Figure I-5-2).



Figure I-5-2 CMOS Setup Utility on a PC

5.2.2 Operating System

Operating system (OS) is the most important program that runs on a computer. Every general-purpose computer must have an operating system to run other programs. Operating systems (ref. Figure I-5-3) perform basic tasks, such as recognizing input from the keyboard, sending output to the display screen, keeping track of files and directories on the disk, and controlling peripheral devices such as disk drives and printers.

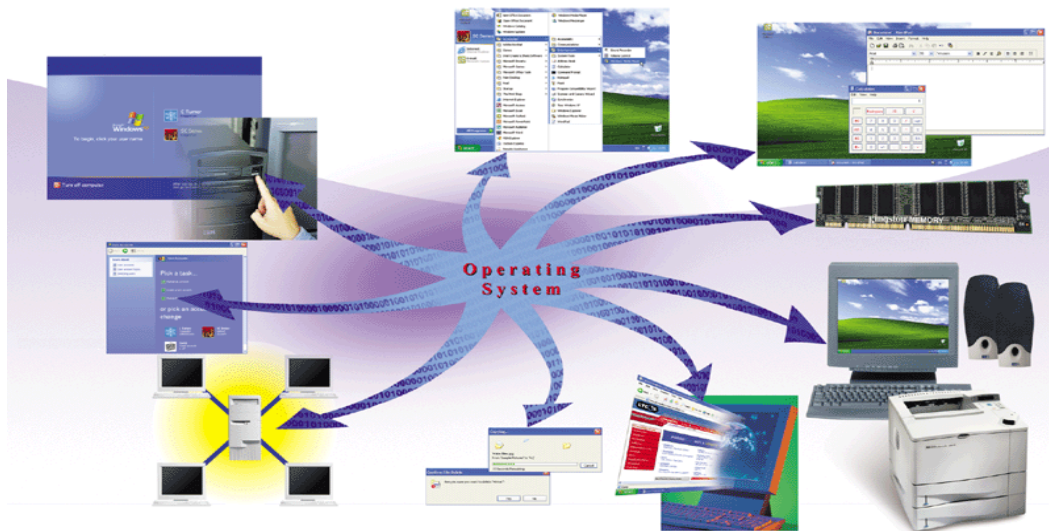


Figure I-5-3 Functions of OS

5.2.2.1 Booting-Up

Booting up (also known as “booting”) is a bootstrapping process that starts operating systems when the user turns on a computer system. A boot sequence is the initial set of operations that the computer performs when power is switched on. The bootloader typically loads the main operating system for the computer. The main steps in booting up are:

- (1) Turn on the computer. Power supply sends electrical signals to all parts of the computer;
- (2) The processor chip looks for the BIOS in the ROM chip;
- (3) BIOS executes POST (power on self test), checks components, e.g., keyboard, drives etc.;
- (4) POST results are compared to data in CMOS chip;
- (5) BIOS searches for the first available boot disk;
- (6) Load MBR (Master Boot Record) from the first available boot disk;
- (7) Load bootstrap program from the active partition;
- (8) Finally, the OS is loaded into memory (RAM).

5.2.2.2 Categories of OS

(1) users/tasks.

- A single user/single tasking operating system:

~allows only one user to run one program at a time.

E.g., DOS.

- A single user/multitasking operating system:

~allows a single user to work on two or more programs that reside in memory at the same

time.

E.g., Windows 3.X, Windows 95/98.

- A multiuser operating system:

~supports two or more simultaneous users. All mainframes and minicomputers are multi-user systems, but most personal computers and workstations are not. Another term for multi-user is time sharing.

E.g., Unix, Linux (multitasking).

- A multiprocessing operating system:

~supports two or more Central Processing Units (CPUs) running programs at the same time within a single computer system. Multiprocessing sometimes refers to the execution of multiple concurrent software processes in a system as opposed to a single process at any one instant. Multiprocessing involves the coordinated processing of programs by more than one processor.

However, the term multiprogramming is more appropriate to describe this concept, which is implemented mostly in software, whereas multiprocessing is more appropriate to describe the use of multiple hardware CPUs. A system can be both multiprocessing and multiprogramming, only one of the two, or neither of the two.

E.g., Windows 2000/2003, Windows XP/Vista, Windows 7/8/8.1/10, Microsoft server operating systems (Windows NT 3.1/3.5/3.51/4.0, Windows 2000/2003/2008/2012/2016 Server and Windows Home Server), UNIX, Linux.

(2) stand-alone/network/embedded.

- stand-alone:

A stand-alone operating system is a complete operating system that works on a desktop computer, notebook computer, or mobile computing device and that also works in conjunction with a network operating system. Some stand-alone operating systems are called client operating systems because they also work in conjunction with a network operating system. Client operating systems can operate with or without a network. Other stand-alone operating systems include networking capabilities, allowing the home and small business user to set up a small network.

E.g., DOS, Windows XP, Windows 10, Mac OS X, OS/2 Warp Client, UNIX, and Linux.

- Network:

A network operating system (NOS) is an operating system that is designed specifically to control a network and its message (e.g. packet) traffic and queues, controls access by multiple users to network resources such as files, and provides for certain administrative functions, including security.

E.g., NetWare, Windows 2003/XP, Vista, Windows 7/8/10, OS/2 Warp Server for e-business, UNIX, Linux, and Solaris.

- Embedded:

An embedded operating system is an operating system for embedded computer systems.

These operating systems are designed to be very compact and efficient, forsaking many functions that non-embedded computer operating systems provide, and which may not be used by the specialized applications they run. They are frequently also real-time operating systems. The operating systems on most PDAs and small devices are embedded operating system, reside on a ROM chip. Popular embedded operating systems include Pocket PC (P/PC, PPC), Windows Mobile 2003/2003 SE, Windows Mobile 5/6, Windows Phone CE/7/8, Windows 10 Mobile, Palm OS, Symbian OS, Windows IoT (formerly Windows Embedded), VxWorks, Android, and iOS (originally iPhone OS).

5.2.2.3 Command Based vs. Graphical OS

1) Command Based OS

- Command based has limited vocabulary of commands;
- Unusable if do not know commands.

E.g., `copy A:\afile.txt C:\dir\newname.txt`.

2) Graphical OS

- Graphical presents commands via pull-down menus;
- Generally commands are faster for expert user.

5.2.2.4 Graphical OS Manipulation

1) Start Menu / Start Screen

The Start menu is a user interface element used in Microsoft Windows since Windows 95 and in some other operating systems. It provides a central launching point for computer programs and performing other tasks.

Traditionally, the Start menu provided a customizable nested list of programs for the user to launch, as well as a list of most recently opened documents, a way to find files and get help, and access to the system settings. Later enhancements via Windows Desktop Update included access to special folders like “My Documents” and “Favorites” (browser bookmarks). Windows XP’s Start menu was expanded to encompass various My Documents folders (including My Music and My Pictures), and transplanted other items like My Computer and My Network Places from the Windows desktop. Until Windows Vista, the Start menu was constantly expanded across the screen as the user navigated through its cascading sub-menus.

Windows 10 (ref. Figure I-5-4) re-introduced the Start menu in a revised form. It uses a two column design similar to Windows 7’s version, except that the right side is populated by tiles, similarly to Windows 8’s Start screen. Applications can be pinned to the right half, and their respective tiles can be resized and grouped into user-specified categories. The left column displays a vertical list, containing frequently-used applications, and links to the “All apps” menu, File Explorer, Settings, and Power options. Some of these links, and additional links to folders such as Downloads, Pictures, and Music, can be added through Settings’ “Choose which folders

appear on Start” page. The Start menu can be resized, or be placed in a full-screen display resembling the Windows 8/8.1 Start screen (although scrolling vertically instead of horizontally). The Start menu also enters this state when “Tablet mode” is enabled.

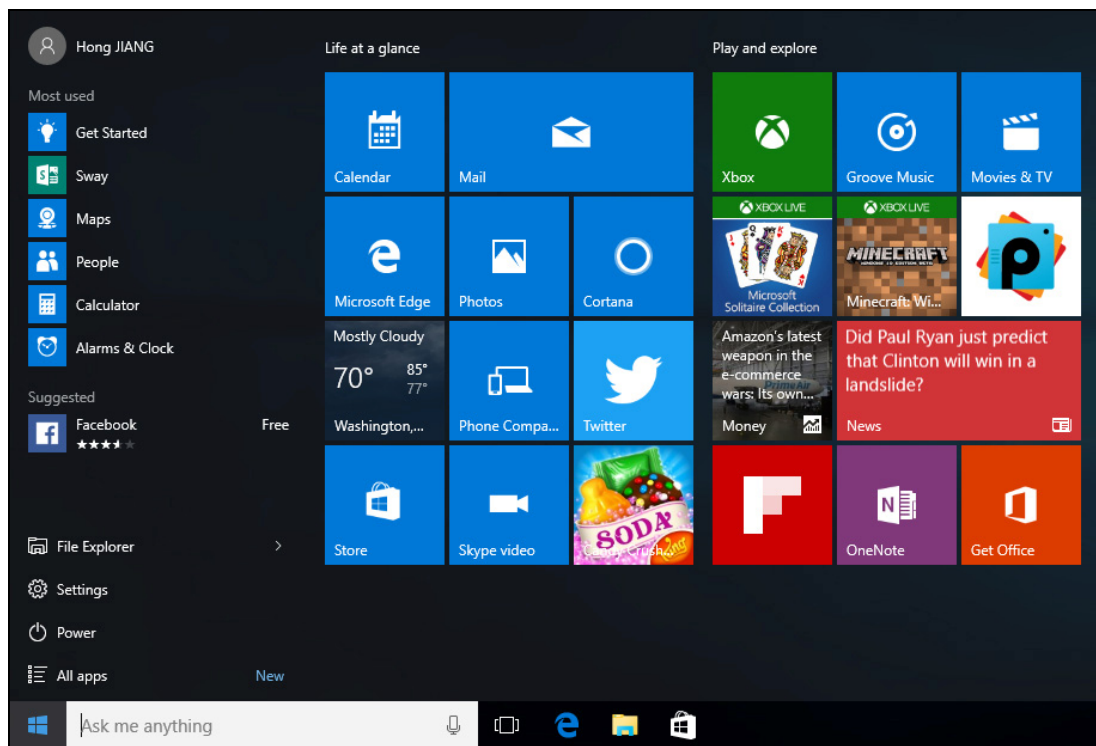


Figure I-5-4 The Start menu in Windows 10

2) File Manager/ File Browser /File Explorer

A file manager or file browser is a computer program that provides a graphical user interface to manage files and folders. It is also the component of the operating system that presents many user interface items on the monitor such as the taskbar and desktop.

The most common operations performed on files or groups of files include creating, opening (e.g. viewing, playing, editing or printing), renaming, moving or copying, deleting and searching for files, as well as modifying file attributes, properties and file permissions. Folders and files may be displayed in a hierarchical tree based on their directory structure. Some file managers contain features inspired by web browsers, including forward and back navigational buttons.

Some file managers provide network connectivity via protocols, such as FTP, NFS, SMB or WebDAV. This is achieved by allowing the user to browse for a file server (connecting and accessing the server's file system like a local file system) or by providing its own full client implementations for file server protocols.

The file manager is renamed File Explorer from Windows 8 and Windows Server 2012

onwards, and introduces new features such as a redesigned interface incorporating a ribbon toolbar (ref. Figure I-5-5), and a redesigned file operation dialog that displays more detailed progress and allows for file operations to be paused and resumed.

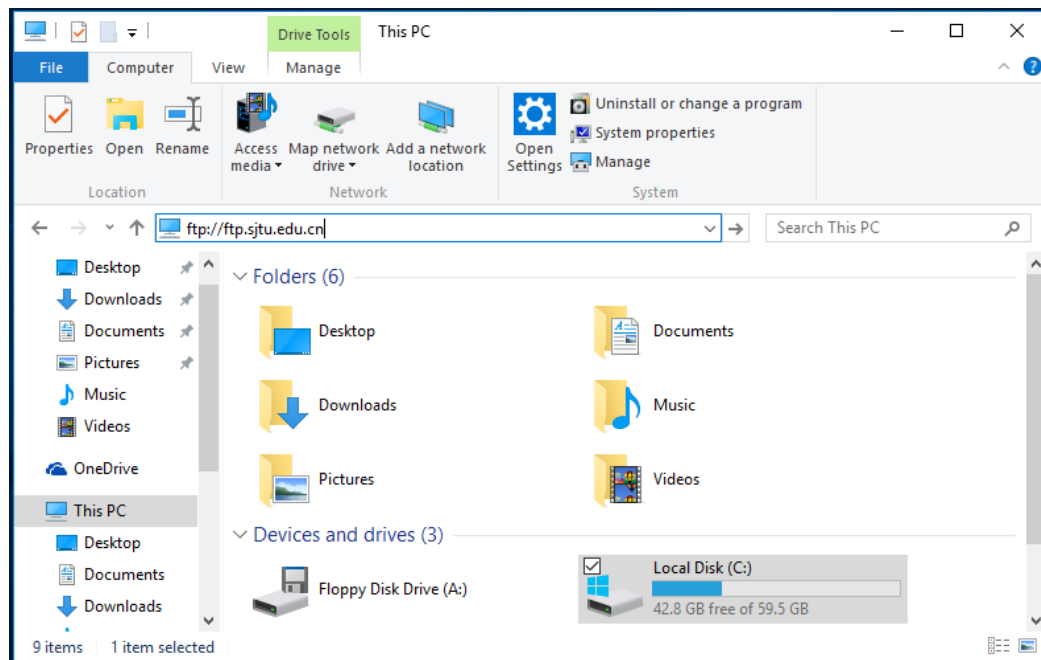


Figure I-5-5 File Explorer in Windows 10

3) Control Panel/Settings app

The Control Panel (ref. Figure I-5-6) is a part of the Microsoft Windows graphical user interface which allows users to view and manipulate basic system settings and controls via applets, such as adding, removing and setting hardware (Printer, Fax, Monitor, Microphone, Speaker, Mouse, Keyboard, Disk Drive, Hub, Sensor, etc.), adding and removing software, controlling user accounts, and changing accessibility options. Additional applets can be provided by third party software.

The Control Panel has been an inherent part of the Microsoft Windows operating system since Windows 2.0, with many of the current applets being added in later versions. Beginning with Windows 95, the Control Panel is implemented as a special folder, i.e. the folder does not physically exist, but only contains shortcuts to various applets such as Add or Remove Programs and Internet Options.

In recent versions of Windows, the Control Panel has two views, Classic View and Category View, and it is possible to switch between these through an option that appears on either the left side or top of the window.

Many of the individual Control Panel applets can be accessed in other ways. For instance, Display Properties can be accessed by right-clicking on an empty area of the desktop and choosing Properties.

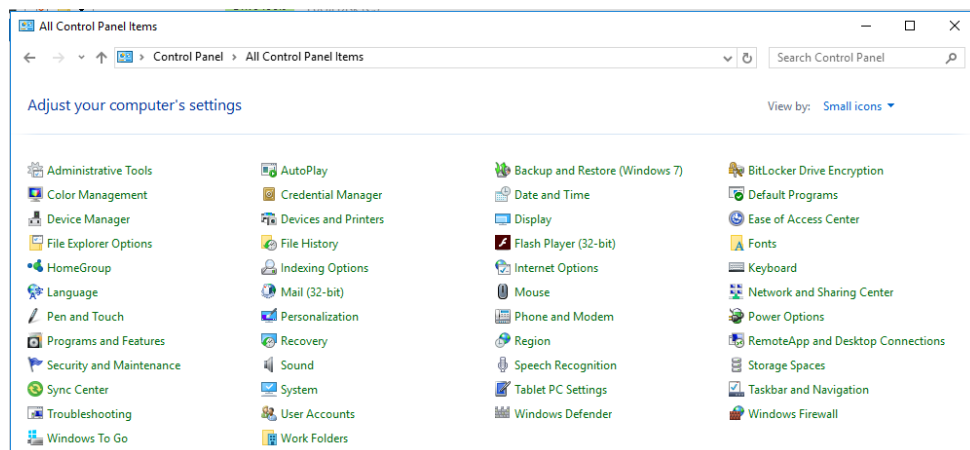


Figure I-5-6 Control Panel in Windows 10 (Viewed by Small Icons)

The classic view consists of shortcuts to the various control panel applets, usually without any description (other than the name). The categories are seen if the user use “Details” view.

The category view consists of categories, which when clicked on display the control panel applets related to the category.

The Control Panel can be accessed quickly by typing control in the Run dialog box (pressing Win+R to open the Run dialog box).

In Windows 10, Control Panel is partially deprecated in favor of Settings app (ref. Figure I-5-7), which was originally introduced on Windows 8 as “PC Settings” to provide a touchscreen-optimized settings area using its Metro-style app platform. The Settings app has been significantly enhanced and now offers most of the traditional Control Panel features. It includes every feature a casual or nontechnical PC user will need.

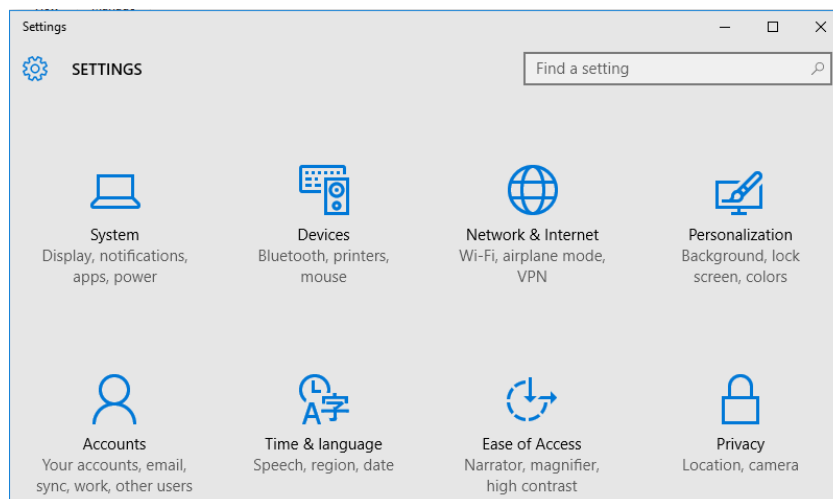


Figure I-5-7 Settings app in Windows 10

5.2.2.5 Some New Features in Windows 10

1) Cortana Personal Assistant

Cortana (ref. Figure I-5-8) is an intelligent personal assistant created by Microsoft for Windows 10, Windows 10 Mobile, Windows Phone 8.1 (where it now supersedes Bing Mobile), Microsoft Band, Xbox One, iOS and Android. It has been launched as a key ingredient of Microsoft's planned "makeover" of the future operating systems for Windows Phone and Windows.

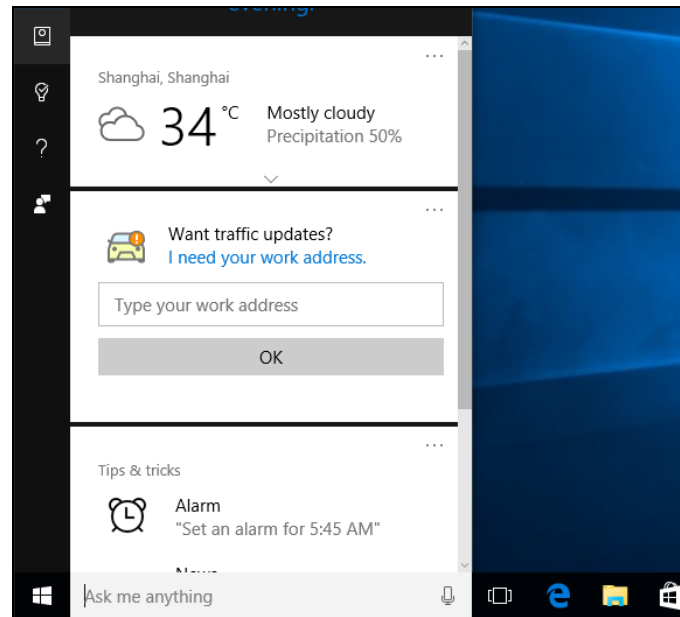


Figure I-5-8 Cortana in Windows 10

Cortana is currently available in English, French, German, Italian, Spanish, Chinese, and Japanese language editions, depending on the software platform and region in which it is used. Cortana mainly competes against assistants such as Apple's Siri and Google's Google Now.

Cortana can set reminders, recognize natural voice without the requirement for keyboard input, and answer questions using information from the Bing search engine (e.g., current weather and traffic conditions, sports scores, biographies). If Firefox is the default browser, Cortana uses the Firefox default search engine instead of Bing. Searches will only be made with Microsoft Bing search engine and all links will open with Microsoft Edge. Windows 8.1's universal Bing SmartSearch features are incorporated into Cortana, which replaces the previous Bing Search app which was activated when a user presses the "Search" button on their device. Cortana includes a music recognition service. Cortana can simulate rolling dice and flipping a coin. It integrates with the Microsoft Band watch band for Windows Phone devices if connected via Bluetooth, it can make reminders and phone notifications. Because Skype is a Microsoft product, Cortana provides smooth activation of Skype video calls from voice commands.

2) Windows Edge

Microsoft Edge is the default web browser developed by Microsoft and included in Windows 10, Windows 10 Mobile, and Xbox One consoles, replacing Internet Explorer as the default web browser on all device classes. Microsoft Edge adds new features such as integration with Cortana digital assistant, annotation tools, and a reading mode.

Microsoft Edge integrates with Microsoft's online platforms: it integrates with the Cortana digital assistant to provide voice control, search functionality, and dynamic, personalized information related to searches within the address bar. Users can make annotations to web pages that can be stored to and shared with OneDrive. It also integrates with the "Reading List" function and provides a "Reading Mode" that strips unnecessary formatting from pages to improve their legibility.

Microsoft has integrated Cortana into numerous products such as Microsoft Edge, the browser bundled with Windows 10. Microsoft's Cortana assistant is deeply integrated into its Edge browser. Cortana can find opening-hours when on restaurant sites, show retail coupons for websites, or show weather information in the address bar.

5.2.3 Utility Programs

A utility program is a kind of system software designed to help analyze, configure, optimize and maintain the computer. A single piece of utility program is usually called a utility (abbr. util) or tool.

5.2.3.1 Windows Utilities

Windows utilities can be accessed from the Start → Windows Administrative Tools command, ref. Figure I-5-9.

1) Disk Cleanup

Disk cleanup (ref. Figure I-5-10) is a computer program tool included in Microsoft Windows designed to help free up space on the computer user's hard drive. It searches and analyzes the hard drive, and then it determines which files on a hard drive may no longer be needed and deletes those files.

There are a number of different types of file categories that disk cleanup targets when it performs the initial disk analysis:

- Downloaded program files;
- Temporary Internet files;
- System queued Windows error reporting files;
- Delivery optimization files;
- Recycle bin;
- Temporary files;
- Thumbnails.

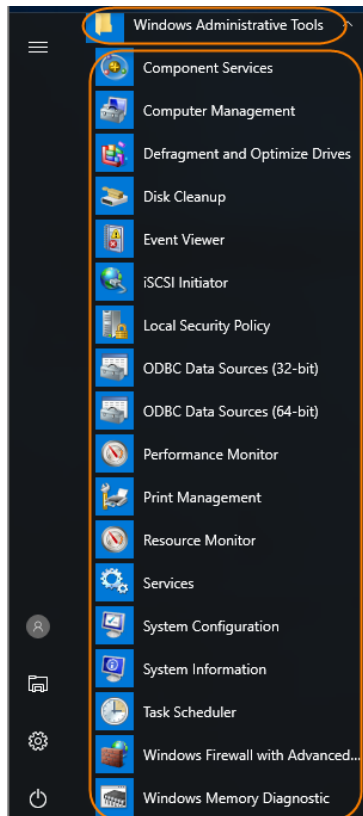


Figure I-5-9 System Tools

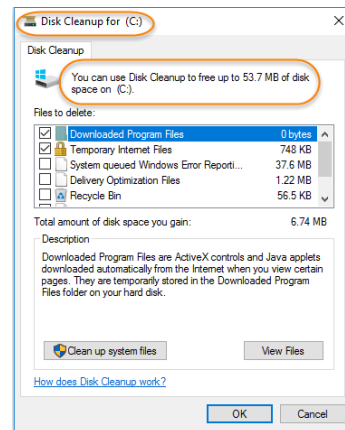


Figure I-5-10 Disk Cleanup

2) Defragment and Optimize Drives

One of the best ways you can improve your PC's performance is by optimizing the drive. Windows includes features to help optimize the different types of drives that PCs use today. No matter which type of drive your PC uses, Windows automatically chooses the optimization that's right for your drive.

By default, Optimize Drives, previously called Disk Defragmenter, runs automatically on a weekly schedule at the time set in automatic maintenance. But you can also optimize drives on your PC manually, ref. Figure I-5-11.

5.2.3.2 Utility Suites

1) Personal Computer Maintenance

Personal computer maintenance utility is a utility program that identifies and fixes operating system problems, detects and repairs disk problems, and includes the capability of improving a computer's performance.

E.g., Norton Utilities.

2) Antivirus programs

Antivirus (or anti-virus) program is a utility program that searches a hard disk for viruses

and removes any that are found. Most antivirus programs include an auto-update feature that enables the program to download profiles of new viruses so that it can check for the new viruses as soon as they are discovered.

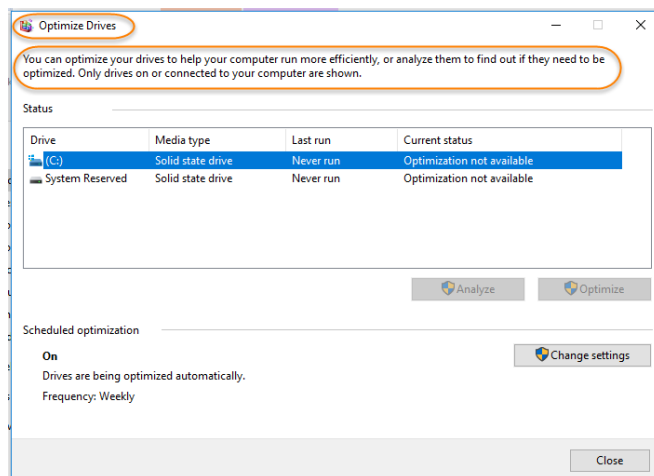


Figure I-5-11 Defragment and Optimize Drives

E.g., Norton Antivirus, McAfee VirusScan.

3) File Compression

File compression utility is a utility program that shrinks the size of a file(s), so the file takes up less storage space than the original file.

E.g., WinRar, WinZip, PKZIP.

5.3 Programming Languages

A programming language (Figure I-5-12) is a vocabulary and set of grammatical rules for instructing a computer to perform specific tasks. Each language has a unique set of keywords (words that it understands) and a special syntax for organizing program instructions. Languages improve in parallel with hardware developments.

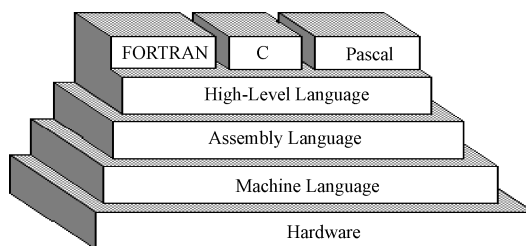


Figure I-5-12 Hierarchy of Programming Languages

Types of programming languages are low-level and high-level.

- Low-level programming languages are machine dependent. A machine-dependent language runs on only one particular type of computer. These programs are not easily portable to other types of computers. Each instruction in a low-level language usually equates to a single machine instruction.
- High-level programming languages are machine independent. A machine-independent language can run on many different types of computers and operating systems.

Machine languages and assembly languages are low-level languages. Procedural languages, non-procedural languages, object-oriented programming languages, and visual programming languages are high-level languages.

5.3.1 Machine language

- 1st. Generation Language.
- Written in the language of the computer—0's and 1's.
- The language to which all other generations of languages must be converted.
- Each different type of CPU has its own unique machine language.

5.3.2 Assembly language

- 2nd. Generation Language.
- Uses simple abbreviations or symbols to represent a number of machine language instructions.
- Specific to a certain physical (or virtual) computer architecture. This is in contrast to most high-level programming languages, which are ideally portable.
- Codes are converted to machine language by a special program called an assembler.

E.g., SUB STA LDA.

5.3.3 Procedural language

- High level language.
- 3rd. Generation Language (3GL).
- More English like.
- Allows concentration on logic of solving problem rather than controlling computer.
- Compiler program converts instructions into machine language, stored as an EXEcutable file.

E.g., BASIC, COBOL, FORTRAN, Ada, Pascal, C.

PRINT("Hello, My Friends!")

5.3.4 Non-procedural language

- Very high-level language.
- 4th. Generation Language (4GL).
- Programmer tells computer what to do but not how to do it.
- Programmer can be 10 times more productive than in 3rd. generation language.

E.g., database (FoxBASE, FoxPro) query language SQL (Structured Query Language),

Prolog, LISP.

SQL statements: select ID, name, score

from StudentFile

where score >= 90 and gender = "female"

will output the following results:

ID	name	score
1011011	Mary	96
1011005	Jenny	91
1011020	Melissa	98

5.3.5 Object-oriented programming (OOP) language

- Uses “objects”—data structures consisting of data fields and methods together with their interactions—to design applications and computer programs.
- Uses event-driven program to check for and respond to events.
- Examples of events: press a key, click a button, etc.
- Programming techniques: data abstraction, encapsulation, modularity, polymorphism, and inheritance.

E.g., C++ (pronounced SEE-plus-plus), Java, C# (pronounced SEE-sharp).

5.3.6 Visual programming language

- 5th. Generation Language.
- Provide a visual or graphical interface for creating source codes.
- Developers drag and drop objects to build programs.

E.g., VB/VB.NET, Delphi, PowerBuilder.

5.3.7 Execution of programming languages

Regardless of what language is used, the program need eventually be converted into machine language so that the computer can understand and execute it. There are two ways to do

this:

- **Compile the program:** to transform a program written in a high-level programming language from source code into object code. Programmers write programs in a form called source code. Source code must go through several steps before it becomes an executable program. The first step is to pass the source code through a compiler, which translates the high-level language instructions into object code. The final step in producing an executable program—after object code is produced by the compiler, it is passed through a linker. The linker combines modules and gives real values to all symbolic addresses, thereby producing machine code. A compiler is likely to perform many or all of the following operations: lexical analysis, preprocessing, parsing, semantic analysis, code generation, and code optimization.
- **Interpret the program:** to translate high-level instructions into an intermediate form, which it then executes. In contrast, a compiler translates high-level instructions directly into machine language. Compiled programs generally run faster than interpreted programs. The advantage of an interpreter, however, is that it does not need to go through the compilation stage during which machine instructions are generated. This process can be time-consuming if the program is long. The interpreter, on the other hand, can immediately execute high-level programs. For this reason, interpreters are sometimes used during the development of a program, when a programmer wants to add small sections at a time and test them quickly. In addition, interpreters are often used in education because they allow students to program interactively.

Many languages have been implemented using both compilers and interpreters, such as Java and C#.