Computer componets

Hardware: physical components that makes a computer system.

Software : computer programs and related data that provide instructions for telling computer hardware what to do and how to do it.

Software Types

1. System Software
2. Application Software

System Software

The system software is collection of programs designed to operate, control, and extend the processing capabilities of the computer itself.

System software are generally prepared by computer manufactures.

These software products comprise of programs written in low-level languages which interact with the hardware at a very basic level.

System software serves as the interface between hardware and the end users.

Some examples of system software are Operating System, Compilers, Interpreter, Assemblers etc.

Features of system software are as follows:

Close to system

Fast in speed

Difficult to design

Difficult to understand

Less interactive

Smaller in size

Difficult to manipulate

Generally written in low-level language

Application Software

Application software products are designed to satisfy a particular need of a particular environment.

All software applications prepared in the computer lab can come under the category of Application software.

Application software may consist of a single program, such as a Microsoft's notepad for writing and editing simple text.

It may also consist of a collection of programs, often called a software package, which

work together to accomplish a task, such as a spreadsheet package.
Examples of Application software are following:

Student Record Software
Inventory Management Software
Income Tax Software
Railways Reservation Software
Microsoft Word
Microsoft Excel
Microsoft PowerPoint

Features of application software are as follows:

Close to user
Easy to design
More interactive
Slow in speed
Generally written in high-level language
Easy to understand
Easy to manipulate and use
Bigger in size and requires large storage space

Assembler

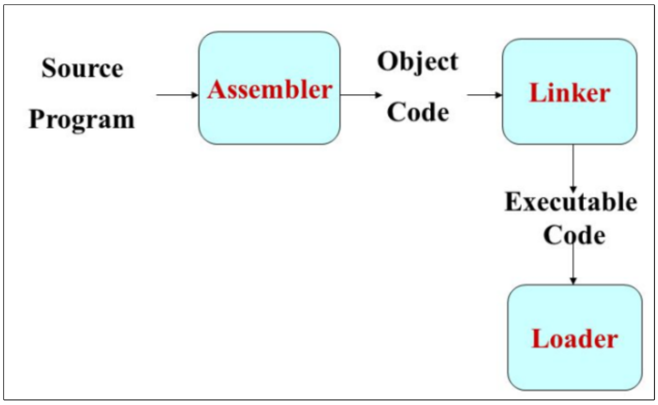
A program which translates an assembly language program into a machine language program is called an assembler.
It translating mnemonic operation codes to their machine language equivalents machine language equivalents.
Examples for mnemonic codes are AND, SUB, MUL, INC, JNE, CMP etc.
Classification of assembler
If an assembler which runs on a computer and produces the machine codes for the same computer then it is called self Assembler or Resident assembler.
If an assembler that runs on a computer and produces the machine codes for other computer then it is called Cross Assembler.

Assemblers are further divided into two types:

One Pass Assembler and Two Pass Assembler.

One pass assembler is the assembler which assigns the memory addresses to the variables and translates the source code into machine code in the first pass simultaneously.
A Two Pass Assembler is the assembler which reads the source code twice. In the first pass, it reads all the variables and assigns them memory addresses. In the second pass, it reads the source code and translates the code into object code.

Role of Assembler



Functions of Assembler

1.Fundamental functions
translating mnemonic operation codes to their machine language equivalents .
assigning machine addresses to symbolic labels

2.Assembler ‘s functions
Convert mnemonic operation codes operation to their machine language equivalents.
Convert symbolic operands to their equivalent machine addresses.
Build the machine instructions in the proper format.
Convert the data constants to internal machine representations.
Write the object program and the assembly listing .

Loader

Loader is a system software program that performs the loading function.
A loader takes the object code of a program as input and prepares it for execution.
Loading is the process of placing the program into memory for execution.
Loader is responsible for initiating the execution of the process.

Functions of Loader

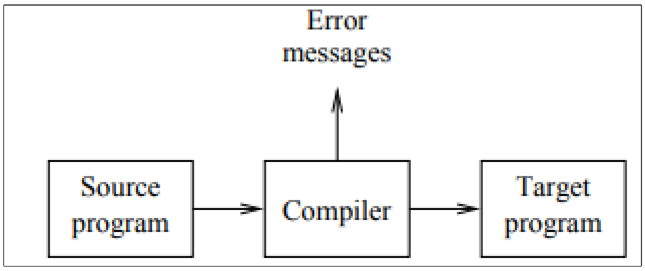
1. Allocation : Loader determines and allocate the required memory space for the program to execute properly
2. Linking : The loader analyze and resolve the symbolic references made in the object program
3. Relocation : The loader maps and relocates the address references to corresponding newly allocated memory space during the execution
4. Loading : Loader loads the machine code corresponding to the object program into the allocated memory space and makes the program ready to execute

Compiler

Compiler is a language translator which translate high level language into machine language.

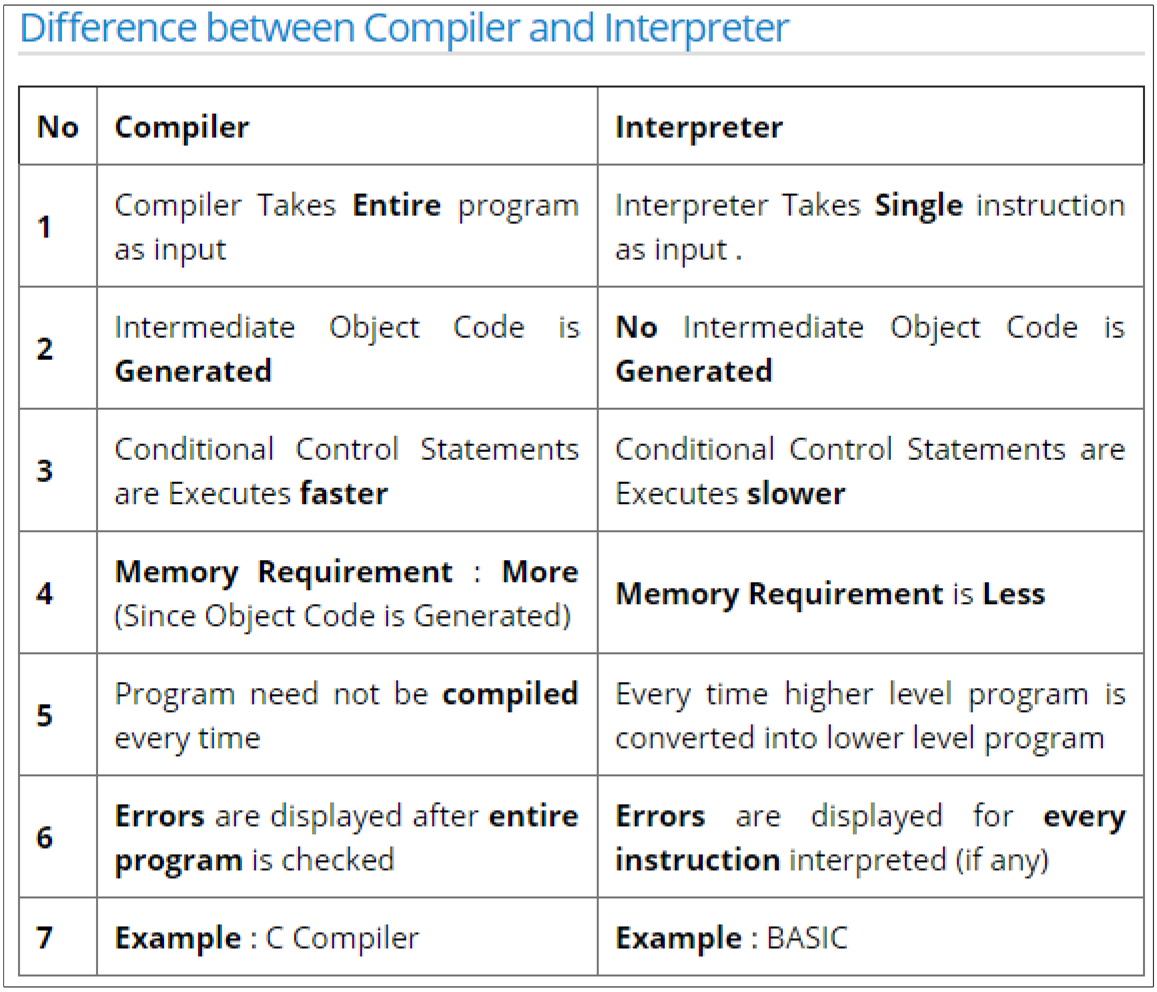
A compiler searches all the errors of a program and lists them.
If the program is error free then it converts the code of program into machine code and then the program can be executed by separate commands.

Role of Compiler



Functions

Compiler does the translation all at once to corresponding machine language
Interpreter
Interpreter is also a language translator which translate high level language into machine language.
An interpreter checks the errors of a program statement by statement.
Functions
It translate one instruction at a time and then execute that instruction immediately



Difference between Compiler and Interpreter

No	Compiler	Interpreter
1	Compiler Takes **Entire** program as input	Interpreter Takes **Single** instruction as input .
2	Intermediate Object Code is **Generated**	**No** Intermediate Object Code is **Generated**
3	Conditional Control Statements are Executes **faster**	Conditional Control Statements are Executes **slower**
4	**Memory Requirement : More** (Since Object Code is Generated)	**Memory Requirement** is **Less**
5	Program need not be **compiled** every time	Every time higher level program is converted into lower level program
6	**Errors** are displayed after **entire program** is checked	**Errors** are displayed for **every instruction** interpreted (if any)
7	**Example** : C Compiler	**Example** : BASIC

Operating System

An operating system is a program that manages a computer's hardware.
A program that acts as an intermediary between a user of a computer and the computer hardware.

USER->Operating System ->Hardware

Functions of OS

Following are some of important functions of an operating system.
Memory management
Processor management
Device management
File management
Security
Control over system performance
Job accounting
Error detecting aids

Coordination between other software and users

1) **Memory Management**

Memory management refers to management of primary memory or main memory.
Main memory is a large array of words or bytes where each word or byte has its own
address. Main memory provides a fast storage that can access directly by the CPU.
So for a program to be executed, it must be in main memory.

Operating system does the following:-

Keep tracks of primary memory ie, what part of it are in use by whom, what part are not in use.
In multiprogramming, OS decides which processs will get memory when and how much.
Allocates the memory when the process requests it to do so.
De-allocates the memory when the process no longer needs it or has been terminated.

2) **Processor Management**

In, multiprogrammming environment, OS decides which process gets the processor when and for how much time.This function is called process scheduling.

Operating system does the following activities for the processor management:-

Keeps tracks of processor and status of process. Program responsible for this task is known as traffic controller.
Allocates the processor(CPU) to a process.
De-allocates processor when processor when processor is no longer required.

3) **Device Management**

Operating system manages device communication via their respective drivers.

Operating system does the following activities for device management:-

Keep track of all devices. Program responsible for this task is known as the Input/Output(I/O) controller.
Decides which process gets the device when and for how much time
Allocates the device in the efficient way.
De-allocates devices.

4) **File Management**

A file system is normally organized into directories for easy navigation and usage. These directories may contain files and other directories.

Operating system does the following activities for file management:-

Keeps track of information, location, uses, status etc. The collective facilities are often known as file system.
Decides who gets the resources.

Allocates the resources.
De-allocate the resources.

Other Important Activities

Followingare some of the important activities that operating system does:-

Security: By means of password and similar other techniques, preventing unauthorized access to programs and data.
Control Over System Performance: Recording delays between request for a service and response from the system.
Job Accounting: Keeping track of time and resources used by various jobs and users.
Error Detecting Aids: Production of dumps, traces, error messages and other debugging and error detecting aids.
Coordination between Other Softwares and Users: Coordination and assignment of compliers, interpretors, assemblers and other software to the various users of the computer system

Goals of OS

1. Convenience : Os makes computer more convenient to users
2. Efficiency : Os allows the computer system resource to be used in an efficient way
3. Ability to Evolve : OS should be constructed in such a way that it should permit the effective development, testing and introduction of new system functions without interfering with services

Types of Operating Systems

Different types of OS are the following :

1. Batch Systems
2. Multiprogrammed batch Systems
3. Timesharing Systems
4. Multi processor Systems
5. Real Time Systems

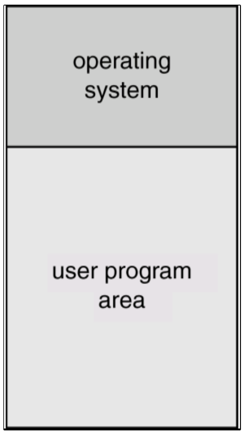
Batch Systems (Simple Batch Systems)

Batch is a sequence of user jobs

In simple batch system, user jobs of similar features are grouped and then run in the computer as groups
It is done with the help of software called Resident Monitor, the first OS
The common input devices used are card readers and tape driver
The common output device used are line printers, and card punches
The user submits the jobs in input devices.
The resident monitor loads the job in sequence and executes it. It automatically transfers

the control from one job to another. This is also called as Automatic Job Sequencing

Memory Layout for a Simple Batch System



To give the information about which program to be executed by the resident monitor,the control cards were introduced.

Eg :

$ FTN - Execute FORTRAN compiler
$LOAD - Execute object program
$ RUN- Execute user program

Data and program cards are distinguished from control cards by putting //(slash) instead of $

Resident monitor loads the program according to the order of the control cards

When a control card indicates the program to run, the monitor loads the program into memory and transfers control to it. When the program completes the control is transferred to monitor

Parts of resident monitor :

1. Control card interpreter – responsible for reading and carrying out instructions on the cards.
2. Loader – loads systems programs and applications programs into memory.
3. Device drivers – know special characteristics and properties for each of the system's I/O devices.

Advantages

No set up time. So CPU idle is reduced
Efficiency is high since monitor take care of all operations

Disadvantages

Allows no interaction
User has no control over immediate results of program

Multiprogrammed Batch Systems

Several jobs are kept in main memory at the same time, and the CPU is multiplexed among them.

OS picks the one job from memory and begins to execute it. If the job has to wait for some I/O resources, The OS simply picks some other jobs from the main memory and execute it. When it job has to wait, CPU is switched to some other jobs and so on.

When the first job finishes waiting it get CPU back.

The number of Jobs running simultaneously and Competing for CPU is known as Degree of Multiprogramming

The time loss in switching from one job to another is called as Context switch

OS Features Needed for Multiprogramming

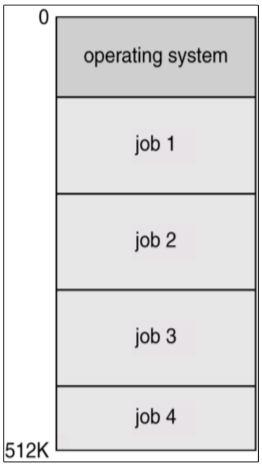
I/O routine supplied by the system.

Memory management – the system must allocate the memory to several jobs.

CPU scheduling – the system must choose among several jobs ready to run.

Allocation of devices.

Memory Layout for a Multiprogrammed Batch System



Time Sharing System

In timesharing systems, a single computer is used by a number of users.

The users can use the system simultaneously, through separate terminals.

The time sharing OS, divides the CPU time among all the users

If the system has 5 terminals, then the CPU time is divided into 5 equal slots, one for each user.

Each user is allocated a very short period of CPU time one by one in rotation.

So each user will have an impression that he has his own computer

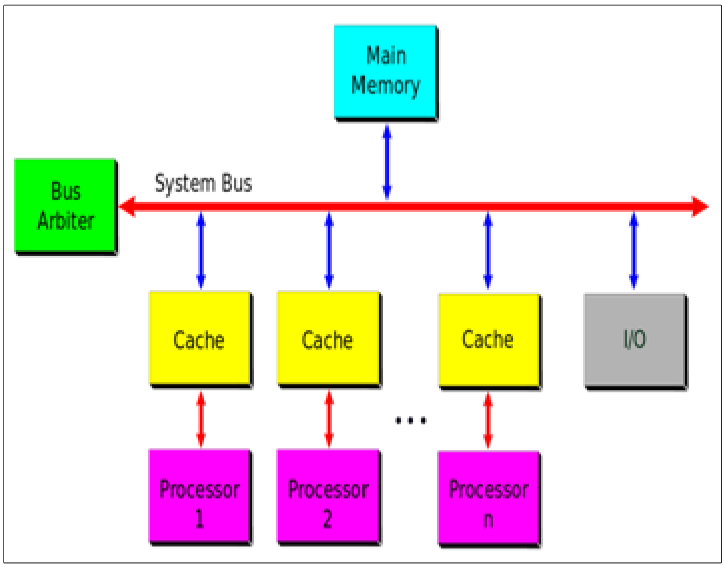
The time during which a user uses the CPU is known as Time Slice or Time Slot or Quantum

Eg: UNIX, LINUX

Multi Processor Systems

Also known as Parallel Systems or Tightly Coupled Systems.
It have more than one processor in close communication
It shares the computer bus, the clock and sometimes memory and peripheral devices
Tightly coupled system – If processors share memory and a clock; communication usually takes place through the shared memory
Two types of Multi processor systems
1. Symmetric Multiprocessing (SMP) :
Each processor runs an identical copy of the operating system. Many processes can run at once without performance deterioration.
Most modern operating systems support SMP

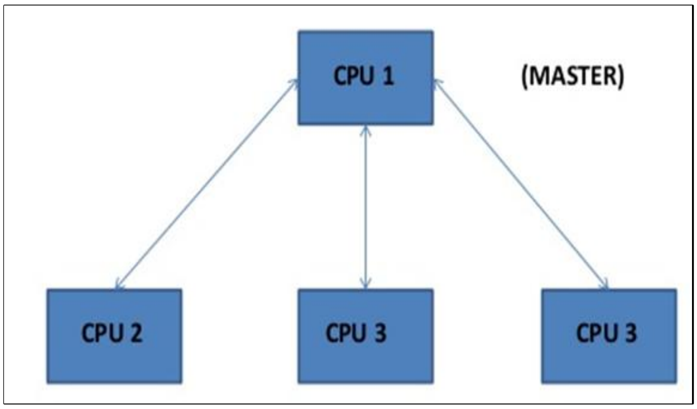
Symmetric Multiprocessing Architecture



Asymmetric multiprocessing :

Each processor is assigned a specific task; master processor schedules and allocates work to slave processors.
More common in extremely large systems
Different processor do different tasks, one may be master and it may control other CPU's.

Asymmetric Multiprocessing Architecture

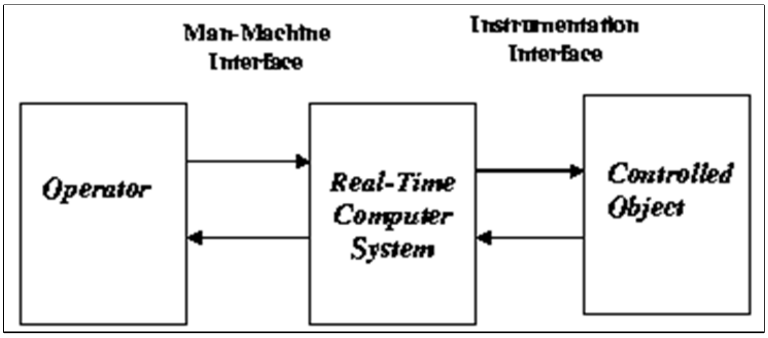


Advantages

Increased throughput : By increasing the number of processors, we will get more work done in less time
Economy of scale : Save money, since they share devices, peripherals
Increased reliability : If functions can be distributed among several processors, then the failure of one processor will not halt the system, only slow it down. This ability to continue providing service proportional to the level of surviving hardware is called graceful degradation. Systems designed for graceful degradation are also called fault tolerant

Real Time Systems

A real time system is a system that controls the industrial process.
Often used as a control device in a dedicated application such as controlling scientific experiments, medical imaging systems, industrial control systems, and some display systems.
The real time systems are connected directly to various input sources and the outputs connected to various controlling mechanisms
o Eg :Vx Works, QNX, LINUX, UBUNTU



There are Two types of Real time systems

1. Hard real-time system. :

Secondary storage limited or absent, data stored in short-term memory, or read-only memory (ROM)
Guarantees that critical task be completed on time
Conflicts with time-sharing systems, not supported by general-purpose operating systems.

2. Soft real-time system :

Limited utility in industrial control or robotics

Less restrictive type

Useful in applications (multimedia, virtual reality) requiring advanced operating-system features.

Operating System Components

1. Process Management
2. Main Memory Management
3. Secondary-Storage Management
4. I/O System Management
5. File Management
6. Protection System
7. Networking
8. Command-Interpreter System

1. Process Management

A process is a program in execution. When a program is loaded into main memory,it becomes process.

Program is a passive entity. But process is a active entity, with a program counter specifying next instruction to execute.

A process needs certain resources, including CPU time, memory, files, and I/O devices, to accomplish its task.

The operating system is responsible for the following activities in connection with process management.

Creating and deleting both user and system process

Suspending and resuming process

Provision of mechanisms for process synchronization and process communication

Providing mechanisms for deadlock handling

2. **Main Memory Management**

Memory is a large array of words or bytes, each with its own address. It is a repository of quickly accessible data shared by the CPU and I/O devices. Main memory is a volatile storage device. It loses its contents in the case of system failure.

The operating system is responsible for the following activities in connections with memory management:

Keep track of which parts of memory are currently being used and by whom.

Decide which processes to load when memory space becomes available.
Allocate and de allocate memory space as needed.

3. Secondary-Storage Management

Since main memory (primary storage) is volatile and too small to accommodate all data and programs permanently, the computer system must provide secondary storage to back up main memory.

Most modern computer systems use disks as the principle on-line storage medium, for both programs and data.

The operating system is responsible for the following activities in connection with disk management:

Free space management
Storage allocation
Disk scheduling

4. I/O System Management

One of the purpose of an OS is to hide the peculiarities of specific hardware devices from the user.

The I/O subsystem consists of:

A memory management component that includes buffering , caching and spooling
A general device-driver interface
Drivers for specific hardware devices

Only the device driver knows the peculiarities of specific device to which it is assigned

5. File Management

A file is a collection of related information defined by its creator. Commonly, files represent programs (both source and object forms) and data.

The operating system is responsible for the following activities in connections with file management:

File creation and deletion.
Directory creation and deletion.
Support of primitives for manipulating files and directories.
Mapping files onto secondary storage.
File backup on stable (nonvolatile) storage media.

6. Protection System

Protection refers to a mechanism for controlling access by programs, processes, or users to both system and user resources.

The protection mechanism must:

Distinguish between authorized and unauthorized usage.
Specify the controls to be imposed.
Provide a means of enforcement.

7. Networking (Distributed Systems)

A distributed system is a collection processor that do not share memory or a clock. Each processor has its own local memory. The processors in the system are connected through a communication network.

A distributed system provides user access to various system resources.

Access to a shared resource allows:

Computation speed-up

Increased data availability

Enhanced reliability

8. Command-Interpreter System

Command interpreter is the interface between OS and user

Many commands are given to the operating system by control statements which deal with:

process creation and management

I/O handling

secondary-storage management

main-memory management

file-system access

protection

networking

When a new job is started in a batch system or user logon to a timeshared system, a program that reads and interprets control ststementsis executed automatically.This program is known as Control card interpreter or Command line interpreter or Shell

Its function is to get and execute the next command statement.

Features of an operating system

1. Scheduling
2. Memory Management
3. Allocation of resources
4. Keeping track of usage
5. Data and User security
6. Providing system services such as print spooling
7. Managing input / output
8. Handling Network communication

Comparison of UNIX, LINUX and Windows

	UNIX	DOS	LINUX	Windows
What is it?	Unix is an operating system that is very popular in universities,		Linux is an example of Open Source software deveolpment opearting	Windows is the family of operating system(OS) from Microsoft,which

	companies, big enterprises.		system(OS).	is the most famous Os in the world.
Cost	Unix operating systems were developed mainly for mainframe systms,servers and workstations.		Linux can be freely distributed, downloaded freely.	For desktop or home use, Windows can be expensive. A single copy can cost around $50 to $450 depending on the version of windows.
User	Unix operating systems were developed mainly for mainframe systms,servers and workstations.		Everyone. From home users to developers.	Everyone. From home users to developers.
GUI	Command based OS		KDE,Gnome,Unity	Everyone. From home users to developers.
File system support	Jfs,gpfs,hfs,hfs+,ufs, Xfs, Zfs format		Ext2, Ext3,Ext4, Jfs, Xfs, FAT, FAT32	FAT,FAT32,exFAT, NTFSFAT,FAT32, exFAT, NTFS
Company/ deveoper	Deveoped by group of AT&T employees at Bell lab and Dennis Ritchie.		Linus Torvalds	Microsoft
User experience	Not mch user friendly		There are GUI applicions, but most of the work done through Terminal.	Everything can be controlled through GUI.
Examples	OS X,Solaris, All linux		Ubuntu, Fedora,RedHat, Debian, Archlinux,Android	Windows 8,8.1,7,vista,XP