**Міністерство освіти і науки, молоді та спорту**

**Національний технічний університет України**

**“Київський політехнічний інститут”**

Факультет інформатики та обчислювальної техніки

Кафедра АСОІУ

**РЕФЕРАТ**

з дисципліни “Англійська мова професійного спрямування”

на тему: **“Operating systems (OS)”**

Виконала: Павленкова Є.О.

ІС-31

Викладач: Соколова Л.Ф.

Київ 2016

Contents

[Introduction 2](#_Toc449562487)

[History 3](#_Toc449562488)

[Components of OS 4](#_Toc449562489)

[Kernel 4](#_Toc449562490)

[Program execution 4](#_Toc449562491)

[Interrupts 4](#_Toc449562492)

[Modes 5](#_Toc449562493)

[Memory management 6](#_Toc449562494)

[Types of operating systems 6](#_Toc449562495)

[Single- and multi-tasking 6](#_Toc449562496)

[Single- and multi-user 7](#_Toc449562497)

[Distributed 7](#_Toc449562498)

[Embedded 7](#_Toc449562499)

[Real-time 7](#_Toc449562500)

[Library 8](#_Toc449562501)

[Examples of operating systems 8](#_Toc449562502)

[Microsoft Windows 8](#_Toc449562503)

[Mac OS X 8](#_Toc449562504)

[Linux 9](#_Toc449562505)

[Summary in English 10](#_Toc449562506)

[Summary translation 11](#_Toc449562507)

[References 12](#_Toc449562508)

[Glossary 13](#_Toc449562509)

# Introduction

The operating system is the most important program that runs on a computer. An operating system (OS) is system software that manages computer hardware and software resources and provides common services for computer programs. The operating system is a component of the system software in a computer system. Application programs usually require an operating system to function. [5]

In the logical structure of a typical computer system operating system occupies a position between the devices with their microarchitecture, machine language, and perhaps their own (built-in) firmware (drivers) - with one hand - and the applications software on the other. The operating system is usually stored in the external memory of the computer.

As part of the operating system, there are three groups of components:

* a core containing the scheduler; device drivers directly control equipment; the networking subsystem, the file system;
* system libraries;
* shell utilities.

Operating systems 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.[4]

The operating system is also responsible for security, ensuring that unauthorized users do not access the system.

Your computer's operating system manages all of the software and hardware on the computer. Most of the time, there are many different computer programs running at the same time, and they all need to access your computer's central processing unit (CPU), memory, and storage. The operating system coordinates all of this to make sure each program gets what it needs.[4]

The purpose of an operating system is to provide an environment in which a user can execute programs. The main purpose of an operating is to make the computer system convenient to use and user can the computer hardware in an efficient manner.

Time-sharing operating systems schedule tasks for efficient use of the system and may also include accounting software for cost allocation of processor time, mass storage, printing, and other resources.[5]

Operating systems are found on many devices that contain a computer—from cellular phones and video game consoles to web servers and supercomputers.

Most of the computers are using the Window as their operating system.

Examples of other modern operating systems include Apple OS X, Linux and its variants.

Nowadays there are many types of operating systems and each of them has its advantages and disadvantages.

# History

Early computers were built to perform a series of single tasks, like a calculator. Basic operating system features were developed in the 1950s, such as resident monitor functions that could automatically run different programs in succession to speed up processing. Operating systems did not exist in their modern and more complex forms until the early 1960s.

Hardware features were added, that enabled use of runtime libraries, interrupts, and parallel processing. When personal computers became popular in the 1980s, operating systems were made for them similar in concept to those used on larger computers.

In the 1940s, the earliest electronic digital systems had no operating systems. Electronic systems of this time were programmed on rows of mechanical switches or by jumper wires on plug boards. These were special-purpose systems that, for example, generated ballistics tables for the military or controlled the printing of payroll checks from data on punched paper cards. After programmable general purpose computers were invented, machine languages (consisting of strings of the binary digits 0 and 1 on punched paper tape) were introduced that sped up the programming process (Stern, 1981)

OS/360 was used on most IBM mainframe computers beginning in 1966, including computers used by the Apollo program.

In the early 1950s, a computer could execute only one program at a time. Each user had sole use of the computer for a limited period of time and would arrive at a scheduled time with program and data on punched paper cards or punched tape. The program would be loaded into the machine, and the machine would be set to work until the program completed or crashed. Programs could generally be debugged via a front panel using toggle switches and panel lights. It is said that Alan Turing was a master of this on the early Manchester Mark 1 machine, and he was already deriving the primitive conception of an operating system from the principles of the Universal Turing machine.

Later machines came with libraries of programs, which would be linked to a user's program to assist in operations such as input and output and generating computer code from human-readable symbolic code. This was the genesis of the modern-day operating system. However, machines still ran a single job at a time. At Cambridge University in England the job queue was at one time a washing line (clothes line) from which tapes were hung with different colored clothes-pegs to indicate job-priority.

An improvement was the Atlas Supervisor introduced with the Manchester Atlas commissioned in 1962, "considered by many to be the first recognisable modern operating system". Brinch Hansen described it as "the most significant breakthrough in the history of operating systems."

# Components of OS

The components of an operating system all exist in order to make the different parts of a computer work together. All user software needs to go through the operating system in order to use any of the hardware, whether it be as simple as a mouse or keyboard or as complex as an Internet component.

## Kernel

A kernel connects the application software to the hardware of a computer.

With the aid of the firmware and device drivers, the kernel provides the most basic level of control over all of the computer's hardware devices. It manages memory access for programs in the RAM, it determines which programs get access to which hardware resources, it sets up or resets the CPU's operating states for optimal operation at all times, and it organizes the data for long-term non-volatile storage with file systems on such media as disks, tapes, flash memory, etc.

## Program execution

The operating system provides an interface between an application program and the computer hardware, so that an application program can interact with the hardware only by obeying rules and procedures programmed into the operating system. The operating system is also a set of services which simplify development and execution of application programs. Executing an application program involves the creation of a process by the operating system kernel which assigns memory space and other resources, establishes a priority for the process in multi-tasking systems, loads program binary code into memory, and initiates execution of the application program which then interacts with the user and with hardware devices.

## Interrupts

Interrupts are central to operating systems, as they provide an efficient way for the operating system to interact with and react to its environment. The alternative – having the operating system "watch" the various sources of input for events (polling) that require action – can be found in older systems with very small stacks (50 or 60 bytes) but is unusual in modern systems with large stacks. Interrupt-based programming is directly supported by most modern CPUs. Interrupts provide a computer with a way of automatically saving local register contexts, and running specific code in response to events. Even very basic computers support hardware interrupts, and allow the programmer to specify code which may be run when that event takes place.

When an interrupt is received, the computer's hardware automatically suspends whatever program is currently running, saves its status, and runs computer code previously associated with the interrupt; this is analogous to placing a bookmark in a book in response to a phone call. In modern operating systems, interrupts are handled by the operating system's kernel. Interrupts may come from either the computer's hardware or the running program.

When a hardware device triggers an interrupt, the operating system's kernel decides how to deal with this event, generally by running some processing code. The amount of code being run depends on the priority of the interrupt (for example: a person usually responds to a smoke detector alarm before answering the phone). The processing of hardware interrupts is a task that is usually delegated to software called a device driver, which may be part of the operating system's kernel, part of another program, or both. Device drivers may then relay information to a running program by various means.

A program may also trigger an interrupt to the operating system. If a program wishes to access hardware, for example, it may interrupt the operating system's kernel, which causes control to be passed back to the kernel. The kernel then processes the request. If a program wishes additional resources (or wishes to shed resources) such as memory, it triggers an interrupt to get the kernel's attention.

## Modes

Privilege rings for the x86 available in protected mode. Operating systems determine which processes run in each mode.

Modern CPUs support multiple modes of operation. CPUs with this capability use at least two modes: protected mode and supervisor mode. The supervisor mode is used by the operating system's kernel for low level tasks that need unrestricted access to hardware, such as controlling how memory is written and erased, and communication with devices like graphics cards. Protected mode, in contrast, is used for almost everything else. Applications operate within protected mode, and can only use hardware by communicating with the kernel, which controls everything in supervisor mode. CPUs might have other modes similar to protected mode as well, such as the virtual modes in order to emulate older processor types, such as 16-bit processors on a 32-bit one, or 32-bit processors on a 64-bit one.

When a computer first starts up, it is automatically running in supervisor mode. The first few programs to run on the computer, being the BIOS or EFI, bootloader, and the operating system have unlimited access to hardware – and this is required because, by definition, initializing a protected environment can only be done outside of one. However, when the operating system passes control to another program, it can place the CPU into protected mode.

In protected mode, programs may have access to a more limited set of the CPU's instructions. A user program may leave protected mode only by triggering an interrupt, causing control to be passed back to the kernel. In this way the operating system can maintain exclusive control over things like access to hardware and memory.

The term "protected mode resource" generally refers to one or more CPU registers, which contain information that the running program isn't allowed to alter. Attempts to alter these resources generally causes a switch to supervisor mode, where the operating system can deal with the illegal operation the program was attempting (for example, by killing the program).

## Memory management

Among other things, a multiprogramming operating system kernel must be responsible for managing all system memory which is currently in use by programs. This ensures that a program does not interfere with memory already in use by another program. Since programs time share, each program must have independent access to memory.

Cooperative memory management, used by many early operating systems, assumes that all programs make voluntary use of the kernel's memory manager, and do not exceed their allocated memory. This system of memory management is almost never seen any more, since programs often contain bugs which can cause them to exceed their allocated memory. If a program fails, it may cause memory used by one or more other programs to be affected or overwritten. Malicious programs or viruses may purposefully alter another program's memory, or may affect the operation of the operating system itself. With cooperative memory management, it takes only one misbehaved program to crash the system.

Memory protection enables the kernel to limit a process' access to the computer's memory. Various methods of memory protection exist, including memory segmentation and paging. All methods require some level of hardware support (such as the 80286 MMU), which doesn't exist in all computers.

In both segmentation and paging, certain protected mode registers specify to the CPU what memory address it should allow a running program to access. Attempts to access other addresses trigger an interrupt which cause the CPU to re-enter supervisor mode, placing the kernel in charge. This is called a segmentation violation or Seg-V for short, and since it is both difficult to assign a meaningful result to such an operation, and because it is usually a sign of a misbehaving program, the kernel generally resorts to terminating the offending program, and reports the error.

Windows versions 3.1 through ME had some level of memory protection, but programs could easily circumvent the need to use it. A general protection fault would be produced, indicating a segmentation violation had occurred; however, the system would often crash anyway.

# Types of operating systems

### Single- and multi-tasking

A single-tasking system can only run one program at a time, while a multi-tasking operating system allows more than one program to be running in concurrency.

This is achieved by time-sharing, dividing the available processor time between multiple processes which are each interrupted repeatedly in time-slices by a task scheduling subsstem of the operating system.

Multi-tasking may be characterized in preemptive and co-operative types.

In preemptive multitasking, the operating system slices the CPU time and dedicates a slot to each of the programs. Unix-like operating systems, e.g., Solaris, Linux, as well as AmigaOS support preemptive multitasking. Cooperative multitasking is achieved by relying on each process to provide time to the other processes in a defined manner. 16-bit versions of Microsoft Windows used cooperative multi-tasking. 32-bit versions of both Windows NT and Win9x, used preemptive multi-tasking.

### Single- and multi-user

Single-user operating systems have no facilities to distinguish users, but may allow multiple programs to run in tandem.[1]

A multi-user operating system extends the basic concept of multi-tasking with facilities that identify processes and resources, such as disk space, belonging to multiple users, and the system permits multiple users to interact with the system at the same time. Time-sharing operating systems schedule tasks for efficient use of the system and may also include accounting software for cost allocation of processor time, mass storage, printing, and other resources to multiple users.

### Distributed

A distributed operating system manages a group of distinct computers and makes them appear to be a single computer. The development of networked computers that could be linked and communicate with each other gave rise to distributed computing. Distributed computations are carried out on more than one machine. When computers in a group work in cooperation, they form a distributed system.[2]

### Embedded

Embedded operating systems are designed to be used in embedded computer systems. They are designed to operate on small machines like PDAs with less autonomy. They are able to operate with a limited number of resources. They are very compact and extremely efficient by design. Windows CE and Minix 3 are some examples of embedded operating systems.

### Real-time

A real-time operating system is an operating system that guarantees to process events or data within a certain short amount of time. A real-time operating system may be single- or multi-tasking, but when multitasking, it uses specialized scheduling algorithms so that a deterministic nature of behavior is achieved. An event-driven system switches between tasks based on their priorities or external events while time-sharing operating systems switch tasks based on clock interrupts.

### Library

A library operating system is one in which the services that a typical operating system provides, such as networking, are provided in the form of libraries. These libraries are composed with the application and configuration code to construct unikernels — which are specialised, single address space, machine images that can be deployed to cloud or embedded environments.[5]

# Examples of operating systems

Operating systems usually come preloaded on any computer you buy. Most people use the operating system that comes with their computer, but it's possible to upgrade or even change operating systems.

Modern operating systems use a graphical user interface, or GUI. A GUI lets you use your mouse to click icons, buttons, and menus, and everything is clearly displayed on the screen using a combination of graphics and text.

Each operating system's GUI has a different look and feel, so if you switch to a different operating system it may seem unfamiliar at first. However, modern operating systems are designed to be easy to use, and most of the basic principles are the same.

### Microsoft Windows

Microsoft created the Windows operating system in the mid-1980s. Over the years, there have been many different versions of Windows, but the most recent ones are Windows 10 (released in 2015), Windows 8 (2012), Windows 7 (2009), and Windows Vista (2007). Windows comes preloaded on most new PCs, which helps to make it the most popular operating system in the world.

If you're buying a new computer or are upgrading to a newer version of Windows, you can choose from several different editions of Windows, such as Home Premium, Professional, and Ultimate. You may need to do some research to decide which edition is right for you.

### Mac OS X

Mac OS is a line of operating systems created by Apple. It comes preloaded on all new Macintosh computers, or Macs. All of the recent versions are known as OS X , and the specific versions include Yosemite (released in 2014), Mavericks (2013), Mountain Lion (2012), Lion (2011), and Snow Leopard (2009). Apple also offers a version called Mac OS X Server, which is designed to be run on servers.

According to StatCounter Global Stats, Mac OS X users account for 9.5% of the operating systems market as of September 2014—much lower than the percentage of Windows users (almost 90%). One reason for this is that Apple computers tend to be more expensive. However, many people prefer the look and feel of Mac OS X.

### Linux

Linux is a family of open-source operating systems, which means they can be modified and distributed by anyone around the world. This is different from proprietary software like Windows, which can only be modified by the company that owns it (Microsoft). The advantages of Linux are that it is free, and there are many different distributions—or versions—you can choose from. Each distribution has a different look and feel, and the most popular ones include Ubuntu, Mint, and Fedora.

Linux is named after Linus Torvalds, who created the Linux kernel in 1991.

According to StatCounter Global Stats, Linux users account for less than 2% of the operating systems market as of September 2014. However, most servers run Linux because it's relatively easy to customize.[4]

***Other***

There have been many operating systems that were significant in their day but are no longer so, such as AmigaOS; OS/2 from IBM and Microsoft; Mac OS, the non-Unix precursor to Apple's Mac OS X; BeOS; XTS-300; RISC OS; MorphOS; Haiku; BareMetal and FreeMint. Some are still used in niche markets and continue to be developed as minority platforms for enthusiast communities and specialist applications. OpenVMS, formerly from DEC, is still under active development by Hewlett-Packard. Yet other operating systems are used almost exclusively in academia, for operating systems education or to do research on operating system concepts. A typical example of a system that fulfills both roles is MINIX, while for example Singularity is used purely for research.

Other operating systems have failed to win significant market share, but have introduced innovations that have influenced mainstream operating systems, not least Bell Labs' Plan 9.

# Summary in English

When a brand new computer comes off the factory assembly line, it can do nothing. The hardware needs software to make it work, it needs an operating system. Every general-purpose computer must have an operating system to run other programs and applications

An operating system is the most important software that runs on a computer. Without an operating system, a computer is useless.

Some of main functions of an operating system:

* execution of queries of programs (input and output data, starting and stopping other programs, and the allocation of more memory, etc.);
* loading programs into memory and their execution;
* allocation of resources (management RAM and cache, distribution between the processes, the organization of virtual memory, CPU, peripherals);
* providing the user interface and implementation of user interaction;
* saving error information system;
* transmission of information between different internal devices;
* multitasking implementation;
* efficient allocation of resources between the processes of the computer;
* protection of the system as well as user data and programs on the actions of users or applications;
* the interaction between the processes of data exchange, mutual synchronization;
* multiplayer mode and access rights.

The operating system is also responsible for security, ensuring that unauthorized users do not access the system.

There are many different types of operating systems. Each type is designed for its tasks. So everyone can choose the operating system he wants.

The most popular operating system are:

* Microsoft Windows (90%).
* Mac OS X (9.5%).
* Linux (2%).

Development of operating systems is one of the most required and important directions in IT. And it is also a very interesting process to develop a new OS. Operating System it is a program that can be written with any of programming languages. So anyone can try to create an OS, if he knows C, C++, Java or any other language. But creation of really good OS is difficult task.

# Summary translation

Коли новий комп'ютер поставляється з заводської складальної лінії, він нічого не може робити. Апаратне забезпечення потребує програмного забезпечення щоб комп'ютер запрацював, воно потребує операційну систему. Кожен комп'ютер загального призначення повинен мати операційну систему для запуску інших програм і додатків.

Операційна система є найбільш важливим програмним забезпеченням, комп'ютера. Без операційної системи, комп'ютер ні на що не здатен.

Деякі з основних функцій операційної системи:

* виконання запитів програм (вхідних і вихідних даних, запуск і зупинка інших програм, а також розподіл додаткової пам'яті, і т.д.);
* завантаження програм в пам'ять і їх виконання;
* виділення ресурсів (управління основною і кеш-пам'яттю, розподіл між процесами, організація віртуальної пам'яті, управління процесором, периферійними пристроями);
* надання користувацького інтерфейсу;
* збереження інформації про помилки системи;
* реалізація багатозадачності;
* ефективний розподіл ресурсів між процесами в комп'ютерній системі;
* захист системи, а також даних користувача і програм від дій користувачів або додатків;
* взаємодія між процесами: обмін даними, взаємна синхронізація;
* багатокористувацький режим роботи і розмежування прав доступу.

Є багато різних типів операційних систем. Кожен тип призначений для виконання своїх завдань. Таким чином, кожен може вибрати операційну систему яку він хоче.

Найбільш популярні операційні системи:

* Microsoft Windows (90%).
* Mac OS X (9.5%).
* Linux (2%).

Розвиток і розробка операційних систем є одним з найбільш необхідних і важливих напрямків в IT. А також розробляти нову ОС це дуже цікаво.

ОС це програма, що може бути написана на будь-якій мові програмування. Тож будь-хто може створити якусь ОС на С, С++, Java чи іншій мові. Але створення дійсно гарної ОС є важким завданням.

# References

1. Lorch, Jacob R., and Alan Jay Smith. "Reducing processor power consumption by improving processor time management in a single-user operating system." Proceedings of the 2nd annual international conference on Mobile computing and networking. ACM, 1996
2. Mishra, Balmukund, Ninni Singh, and Ravideep Singh. "Master-slave group based model for co-ordinator selection, an improvement of bully algorithm." Parallel, Distributed and Grid Computing (PDGC), 2014 International Conference on. IEEE, 2014
3. OS - operating system

[Електронний ресурс] – Режим доступу <http://www.webopedia.com/TERM/O/operating_system.html>

1. Understanding Operating Systems

[Електронний ресурс] – Режим доступу <http://www.gcflearnfree.org/computerbasics/2>

1. Wikipedia

[Електронний ресурс] – Режим доступу <https://en.wikipedia.org/wiki/Operating_system>

# Glossary

1. **Access control list** (ACL) is a table that tells a computer operating system which access rights each user has to a particular system object, such as a file directory or individual file.
2. **Accounting software** is a class of computer software, or program that helps accounting professionals mange accounts and perform accounting operations.
3. **Algorithm** is a formula or set of steps for solving a particular problem. To be an algorithm, a set of rules must be unambiguous and have a clear stopping point.
4. [**Application (application software)**](http://www.webopedia.com/TERM/A/application.html)is a program or group of programs designed for end users.
5. **Assembly line** is a manufacturing process in which parts are added as the semi-finished assembly moves from workstation to work station where the parts are added in sequence until the final assembly is produced.
6. **Batch processing** is the execution of a series of programs ("jobs") on a computer without manual intervention.
7. **Cache** is a component that transparently stores data so that future requests for that data can be served faster. The data that is stored within a cache might be values that have been computed earlier or duplicates of original values that are stored elsewhere.
8. [**Central Processing Unit**](http://www.webopedia.com/TERM/C/CPU.html) (CPU) is the part of the computer where most calculations take place.
9. **Data synchronization** **technologies** are technologies that synchronize a single set of data between two or more devices, automatically copying changes back and forth.
10. **Desktop environment** is a collection of software running on top of an operating system that makes up the desktop Graphical User Interface.
11. **Device** is any machine or component that attaches to a [computer](http://www.webopedia.com/TERM/C/computer.html).
12. **Distributed processing** is a term used to refer to a variety of computer systems that use more than one computer (or processor) to run an application.
13. **Distribution** is a variant refers to something that contains several similarities as something else but is not the same.
14. **Disk operating system** (DOS)is an acronym often used to describe MS-DOS and the Windows command line.
15. [**Driver**](http://www.webopedia.com/TERM/D/driver.html) is a program that controls a device. Every device, whether it be a printer, disk drive, or keyboard, must have a driver program.
16. **Embedded computer systems** is a specialized computer system that is part of a larger system or machine.
17. **End-user license agreement** (EULA) is the contract between the licensor and purchaser, establishing the purchaser's right to use the software.
18. **External command** is an MS-DOS command that is not included in command .com.
19. **File** is a collection of data or information that is stored in a computer that has a name, called the filename.
20. **File manager** is a software program that is often a portion of the operating system that helps a user manage all the files on their computer. For example, all file managers allow the user to view, edit, copy, and delete the files on their computer.
21. **File system** is a system that controls how data is stored and retrieved.
22. [**Firmware**](http://www.webopedia.com/TERM/F/firmware.html)is asoftware (programs or data) that has been written onto read-only memory (ROM). Firmware is a combination of software and hardware.
23. **Folder** is a virtual location where programs, files, and other folders can be located.
24. **Graphical user interface** (GUI)is an approach that uses the computer's graphics capabilities to make the program easier to use with pointing devices, menus and icons.
25. **Icon** is a small graphical representation of a program or file that, when clicked on, will be run or opened.
26. **Install** is a term used to describe the process of physically connecting a component within a computer.
27. **Hardware**  is the collection of physical elements that constitutes a computer system.
28. **Kernel** is a computer program that manages input/output requests from software and translates them into data processing instructions for the central processing unit and other electronic components of a computer. The kernel is a fundamental part of a modern computer's operating system.
29. **Machine language** is the lowest-level [programming language](http://www.webopedia.com/TERM/P/programming_language.html). Machine languages are the only [languages](http://www.webopedia.com/TERM/L/language.html) understood by [computers](http://www.webopedia.com/TERM/C/computer.html).
30. **Mainframe** **computer** is a computer that used primarily by large organizations for critical applications, bulk data processing such as census, industry and consumer statistics, enterprise resource planning and transaction processing.
31. [**Memory resident**](http://www.webopedia.com/TERM/M/memory_resident.html) is data that is permanently in [memory](http://www.webopedia.com/TERM/M/memory.html). Normally, a [computer](http://www.webopedia.com/TERM/C/computer.html) does not have enough memory to hold all the [programs](http://www.webopedia.com/TERM/P/program.html) you use.
32. **Memory management** is the act of managing computer memory at the system level.
33. **Mouse** is a device that controls the movement of the cursor or pointer on a display screen. A mouse is a small object you can roll along a hard, flat surface.
34. **Multi User Interface** (MUI) is a term used to describe an interface that runs over the top of an English version of Microsoft Windows and supports different languages. MUI allows large companies to distribute large installations of Microsoft Windows in multiple languages.
35. **Mutual exclusion** is a term that used to allow only one process to access critical section (a part of code which accesses the critical resource).
36. **Multitasking** is a concept of performing multiple tasks (also known as processes) over a certain period of time by executing them concurrently.
37. **Multiprocessing** is the use of two or more central processing units (CPUs) within a single computer system.
38. **Multi-user software** is a software that allows access by multiple users of a computer.
39. **Open-source operating systems** – is an OS that can be modified by anyone.
40. **Open core** is a business model for the monetization of commercially-produced open source software.
41. **Personal digital assistant** (PDA) is a handheld device that combines computing, telephone/fax, Internet and networking features.
42. [**Peripheral device**](http://www.webopedia.com/TERM/P/peripheral_device.html)is a computer device, such as a CD-ROM drive or printer, that is not part of the essential computer.
43. **Program**  is a collection of [instructions](https://en.wikipedia.org/wiki/Instruction_set) that performs a specific task when [executed](https://en.wikipedia.org/wiki/Execution_(computing)) by a [computer](https://en.wikipedia.org/wiki/Computer).
44. **Programmable interrupt controller** (PIC) is a device that is used to combine several sources of interrupt onto one or more CPU lines, while allowing priority levels to be assigned to its interrupt outputs.
45. **Proprietary software** (non-free software, closed-source software) is software that fails to meet the criteria for free or open source software.
46. **Query** is a request for information from a database.
47. **Random-access memory** (RAM) is a form of computer data storage.
48. **Runlevel** is a mode of operation in one of the computer operating systems that implement Unix System V-style initialization.
49. **Security software** isa term used to describe any software that provides security for a computer or network.
50. **Semaphore** is a variable or abstract data type that is used for controlling access, by multiple processes, to a common resource in a parallel programming or a multi user environment**.**
51. **Single address space operating system** (SASOS) is an operating system that provides for all processes only one globally shared virtual address space.
52. **Scheduling** is the method by which work specified by some means is assigned to resources that complete the work.
53. **Server** is a computer program or a machine that waits for requests from other machines or software (clients) and responds to them.
54. [**Shell**](http://www.webopedia.com/TERM/S/shell.html) is the outermost layer of a [program](http://www.webopedia.com/TERM/P/program.html). Shell is another term for [user interface](http://www.webopedia.com/TERM/U/user_interface.html).
55. **Software** is a set of instructions that directs a [computer](https://en.wikipedia.org/wiki/Computer) to perform specific operations.
56. **Supercomputer** is the fastest type of computer. Supercomputers are very expensive and are employed for specialized applications that require immense amounts of calculations.
57. **Supervisor program** is a computer program, usually part of an operating system, that controls the execution of other routines and regulates work scheduling, input/output operations, error actions, and similar functions and regulates the flow of work in a data processing system.
58. **System call** is a process when a program requests a service from an operating system's kernel.
59. **Task Manager** is an operating system component found in all versions of Microsoft Windows since Windows NT 4.0 and Windows 2000.
60. **Taskbar** a bar located at the bottom of the screen first introduced with Microsoft Windows 95 and found in all the versions of Windows that followed.
61. **Templating** is a creating a single virtual machine image as a guest operating system, then saving it as a tool for multiple running virtual machines.
62. **Time-sharing** is a method of dividing the available processor time between multiple processes which are each interrupted repeatedly in time-slices by a task scheduling subsystem of the operating system.
63. **Thread of execution** is the smallest sequence of programmed instructions that can be managed independently by an operating system scheduler.
64. **Unikernels** is a single address space machine images constructed by using library operating systems.
65. **Upgrade** is a new version of a software or hardware product designed to replace an older version of the same product. **User** an individual who uses a computer.
66. **User interface** is a junction between a user and a computer program. An interface is a set of commands or menus through which a user communicates with a program.
67. [**Utility**](http://www.webopedia.com/TERM/U/utility.html)is a program that performs a very specific task, usually related to managing system resources. Operating systems contain a number of utilities for managing devices.
68. **Virtual machine** (VM) is an emulation of a particular computer system.
69. **Virtual memory** ia animaginary memory area supported by some operating systems in conjunction with the hardware.
70. **Mobile operating systems** ia an OS that runs on mobile computing device.