

Operating Systems

Unit-I

Operating System:

An operating system (OS) is a set of programs that manage computer hardware resources and provide common services for application software. The operating system is the most important type of system software in a computer system. Without an operating system, a user cannot run an application program on their computer.

It is a program with following features:

- An operating system is a program that acts as an interface between the software and the computer hardware.
- It is an integrated set of specialized programs that are used to manage overall resources and operations of the computer.
- It is specialized software that controls and monitors the execution of all other programs that reside in the computer, including application programs and other system software.

Examples: Android, iOS, Linux, Mac OS X and Microsoft Windows.

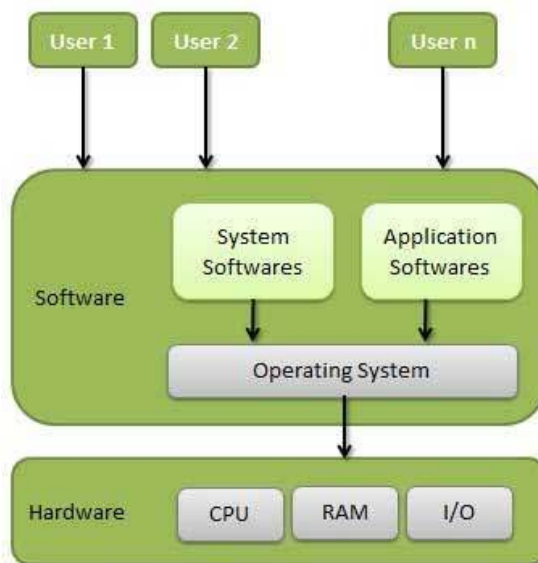
Operating System:

एक ऑपरेटिंग सिस्टम (O.S) प्रोग्रामों का एक समूह है जो कंप्यूटर हार्डवेयर संसाधनों का प्रबंधन करता है और एप्लिकेशन सॉफ्टवेयर के लिए सामान्य सेवाएं प्रदान करता है। ऑपरेटिंग सिस्टम एक कंप्यूटर सिस्टम में सिस्टम सॉफ्टवेयर का सबसे महत्वपूर्ण प्रकार है। ऑपरेटिंग सिस्टम के बिना, कोई उपयोगकर्ता अपने कंप्यूटर पर एक एप्लिकेशन प्रोग्राम नहीं चला सकता है।

यह निम्नलिखित विशेषताओं वाला एक प्रोग्राम है:

- एक ऑपरेटिंग सिस्टम एक प्रोग्राम है जो सॉफ्टवेयर और कंप्यूटर हार्डवेयर के बीच अंतरफलक के रूप में कार्य करता है।
- यह विशेष कार्यक्रमों का एक एकीकृत समूह है जो कंप्यूटर के समग्र संसाधनों और कार्यों को प्रबंधित करने के लिए उपयोग किया जाता है।
- यह विशेष सॉफ्टवेयर है जो कंप्यूटर में रहते हुए अन्य सभी प्रोग्रामों के निष्पादन को नियंत्रित और मॉनिटर करता है, जिसमें एप्लिकेशन प्रोग्राम और अन्य सिस्टम सॉफ्टवेयर शामिल हैं।

Examples: Android, iOS, Linux, Mac OS X and Microsoft Windows.



Functions of Operating System

1. Process Management:

In multiprogramming environment, the OS decides which process gets the processor when and for how much time. This function is called **process scheduling**. An Operating System does the following activities for processor management –

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

2. Memory Management:

Memory management refers to management of Primary Memory or Main Memory. Main memory provides a fast storage that can be access directly by the CPU. So for a program to be executed, it must in the main memory. Operating System does the following activities for memory management.

- Keeps tracks of primary memory i.e. what part of it are in use by whom, what part are not in use.
- In multiprogramming, OS decides which process 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.

3. I/O Managment:

OS manages device communication via their respective drivers. Operating System does the following activities for device management.

- Keeps tracks of all devices. Program responsible for this task is known as the 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 directions. 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.

ऑपरेटिंग सिस्टम के कार्य

1. Process Management:

मल्टिप्रोग्रामिंग वातावरण में, ओएस तय करता है कि प्रोसेसर कब और कितने समय तक मिलता है। इस फ़ंक्शन को **process scheduling** कहा जाता है। एक ऑपरेटिंग सिस्टम प्रोसेसर प्रबंधन के लिए निम्नलिखित कार्य करता है -

- प्रोसेसर और program की स्थिति के ट्रैक रखता है। इस कार्य के लिए जिम्मेदार program, ट्रैफ़िक नियंत्रक के रूप में जाना जाता है.
- प्रोसेसर (सीपीयू) को एक प्रक्रिया में आवंटित करता है.
- प्रोसेसर को डी-ऑलोकैट करता है जब कोई process जरूरी नहीं होती है

2. Memory Management:

मेमोरी प्रबंधन प्राथमिक मेमोरी या मुख्य मेमोरी के प्रबंधन को संदर्भित करता है। मुख्य मेमोरी एक तेज भंडारण प्रदान करता है जो सीधे सीपीयू द्वारा एक्सेस कर सकता है। इसलिए क्रियान्वित होने वाले किसी program के लिए, यह मुख्य मेमोरी में होना चाहिए। ऑपरेटिंग सिस्टम मेमोरी प्रबंधन के लिए निम्नलिखित गतिविधियां करता है:

- प्राथमिक मेमोरी के track को रखता है i.e. इसका किस भाग में उपयोग किया जाता है, किसके द्वारा नहीं इस्तेमाल होता है.
- मल्टिप्रोग्रामिंग में, OS तय करता है कि किस प्रक्रिया को स्मरण मिलेगा कब और कितना।

- स्मृति को आवंटित करता है जब प्रक्रिया इसे करने के लिए अनुरोध करती है।

- स्मृति को डी-आवंटित करता है जब process को इसकी आवश्यकता नहीं होती है.

3. I/O Managment:

OS संबंधित ड्राइवों के माध्यम से डिवाइस संचार का प्रबंधन करता है। ऑपरेटिंग सिस्टम डिवाइस प्रबंधन के लिए निम्नलिखित गतिविधियां करता है:

- सभी डिवाइसों के ट्रैक रखता है इस कार्य के लिए जिम्मेदार कार्यक्रम I / O नियंत्रक के रूप में जाना जाता है
- यह निर्णय लेता है कि कौन से प्रक्रिया डिवाइस कब और कब तक जाती है।
- डिवाइस को कुशल तरीके से आवंटित करता है ।
- डिवाइस को de-allocate करना ।

4. File Management:

आसान नेविगेशन और उपयोग के लिए एक फाइल सिस्टम को सामान्यतः डायरेक्टरी में व्यवस्थित किया जाता है ।

इन डायरेक्टरी में फ़ाइलें और अन्य डायरेक्टरी हो सकती हैं।

ऑपरेटिंग सिस्टम फ़ाइल प्रबंधन के लिए निम्नलिखित गतिविधियां करता है:

- सूचना, स्थान, उपयोग, स्थिति आदि का ट्रैक रखता है। सामूहिक सुविधाएं अक्सर फ़ाइल सिस्टम के रूप में जाने जाते हैं।
- यह निर्णय लेता है कि संसाधन कौन हैं।
- संसाधनों को आवंटित करता है।
- संसाधनों को डी-आवंटित करता है।

- De-allocates the resources.

5. Security Management

By means of password and similar other techniques, it prevents unauthorized access to programs and data.

5. Security Management

पासवर्ड और इसी तरह की अन्य तकनीकों के माध्यम से, यह कार्यक्रमों और डेटा तक अनधिकृत पहुंच को रोकता है।

Generations of Operating System

The First Generation (1940 to early 1950s)

When the first electronic computer was developed in 1940, it was created without any operating system. In early times, users have full access to the computer machine and write a program for each task in absolute machine language. The programmer can perform and solve only simple mathematical calculations during the computer generation, and this calculation does not require an operating system.

The Second Generation (1955 - 1965)

The first operating system (OS) was created in the early 1950s and was known as GMOS. General Motors has developed OS for the IBM computer. The second-generation operating system was based on a single stream batch processing system because it collects all similar jobs in groups or batches and then submits the jobs to the operating system using a punch card to complete all jobs in a machine. At each completion of jobs (either normally or abnormally), control transfer to the operating system that is cleaned after completing one job and then continues to read and initiates the next job in a punch card. After that, new machines were called mainframes, which were very big and used by professional operators.

The Third Generation (1965 - 1980)

During the late 1960s, operating system designers were very capable of developing a new operating system that could simultaneously perform multiple tasks in a single computer program called multiprogramming. The introduction of multiprogramming plays a very important role in developing operating systems that allow a CPU to be busy every time by performing different tasks on a computer at the same time. During the third generation, there was a new development of minicomputer's phenomenal growth starting in 1961 with the DEC PDP-1. These PDP's leads to the creation of personal computers in the fourth generation.

The Fourth Generation (1980 - Present Day)

The fourth generation of operating systems is related to the development of the personal computer. However, the personal computer is very similar to the minicomputers that were developed in the third generation. The cost of a personal computer was very high at that time; there were small fractions of minicomputers costs. A major factor related to creating

ऑपरेटिंग सिस्टम की पीढ़ी

पहली पीढ़ी (1940 से 1950 के दशक की शुरुआत तक)

1940 में जब पहला इलेक्ट्रॉनिक कंप्यूटर विकसित किया गया था, तब इसे बिना किसी ऑपरेटिंग सिस्टम के बनाया गया था। प्रारंभिक समय में, उपयोगकर्ताओं के पास कंप्यूटर मशीन तक पूर्ण पहुंच होती है और प्रत्येक कार्य के लिए पूर्ण मशीनी भाषा में एक प्रोग्राम लिखते हैं। प्रोग्रामर कंप्यूटर निर्माण के दौरान केवल सरल गणितीय गणनाओं को निष्पादित और हल कर सकता है, और इस गणना के लिए ऑपरेटिंग सिस्टम की आवश्यकता नहीं होती है।

दूसरी पीढ़ी (1955 - 1965)

पहला ऑपरेटिंग सिस्टम (OS) 1950 के दशक की शुरुआत में बनाया गया था और इसे GMOS के नाम से जाना जाता था। जनरल मोटर्स ने आईबीएम कंप्यूटर के लिए ओएस विकसित किया है। दूसरी पीढ़ी का ऑपरेटिंग सिस्टम सिंगल स्ट्रीम बैच प्रोसेसिंग सिस्टम पर आधारित था क्योंकि यह सभी समान नौकरियों को समूहों या बैचों में एकत्र करता है और फिर मशीन में सभी कार्यों को पूरा करने के लिए पंच कार्ड का उपयोग करके ऑपरेटिंग सिस्टम को जॉब सबमिट करता है। प्रत्येक कार्य के पूरा होने पर (या तो सामान्य रूप से या असामान्य रूप से), ऑपरेटिंग सिस्टम को नियंत्रण हस्तांतरण जो एक कार्य को पूरा करने के बाद साफ किया जाता है और फिर पढ़ना जारी रखता है और पंच कार्ड में अगली नौकरी शुरू करता है। उसके बाद, नई मशीनों को मेनफ्रेम कहा जाता था, जो बहुत बड़े थे और पेशेवर ऑपरेटरों द्वारा उपयोग किए जाते थे।

तीसरी पीढ़ी (1965 - 1980)

1960 के दशक के अंत के दौरान, ऑपरेटिंग सिस्टम डिज़ाइनर एक नया ऑपरेटिंग सिस्टम विकसित करने में बहुत सक्षम थे जो एक ही कंप्यूटर प्रोग्राम में एक साथ कई कार्य कर सकता था जिसे मल्टीप्रोग्रामिंग कहा जाता है। मल्टीप्रोग्रामिंग की शुरुआत ऑपरेटिंग सिस्टम को विकसित करने में बहुत महत्वपूर्ण भूमिका निभाती है जो एक ही समय में कंप्यूटर पर अलग-अलग कार्य करके सीपीयू को हर बार व्यस्त रखने की अनुमति देता है। तीसरी पीढ़ी के दौरान, 1961 में DEC PDP-1 के साथ मिनीकंप्यूटर के अभूतपूर्व विकास का एक नया विकास हुआ। ये पीढ़ी चौथी पीढ़ी में पर्सनल कंप्यूटर के निर्माण की ओर ले जाती है।

चौथी पीढ़ी (1980 - वर्तमान दिन)

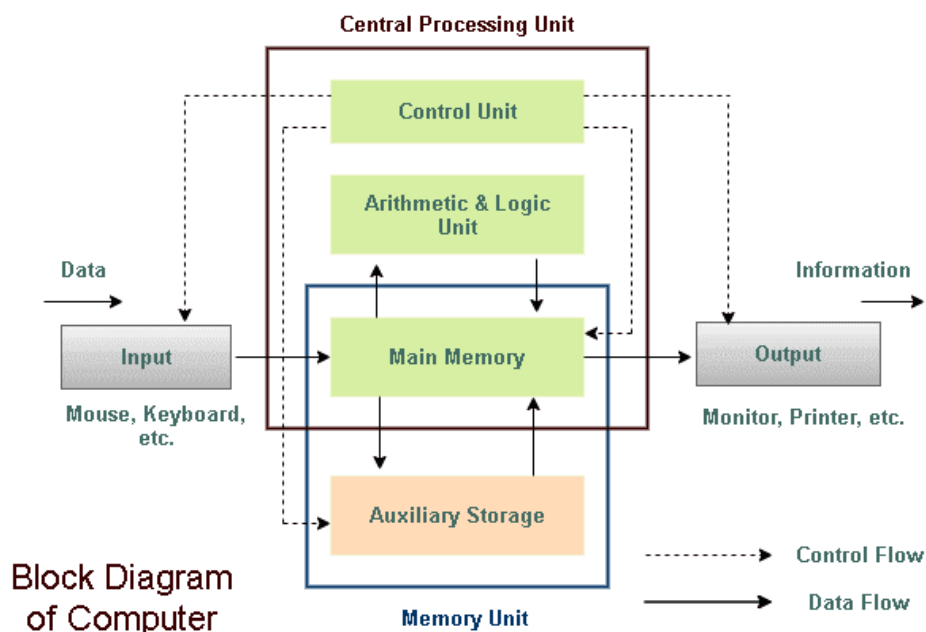
ऑपरेटिंग सिस्टम की चौथी पीढ़ी पर्सनल कंप्यूटर के विकास से संबंधित है। हालाँकि, पर्सनल कंप्यूटर तीसरी पीढ़ी में विकसित किए गए मिनी कंप्यूटरों के समान हैं। उस समय पर्सनल कंप्यूटर की कीमत बहुत अधिक थी; मिनीकंप्यूटर लागत के छोटे अंश थे। पर्सनल कंप्यूटर बनाने से संबंधित एक प्रमुख कारक माइक्रोसॉफ्ट और विंडोज ऑपरेटिंग सिस्टम का जन्म था। Microsoft ने 1975

personal computers was the birth of Microsoft and the Windows operating system. Microsoft created the first window operating system in 1975. After introducing the Microsoft Windows OS, Bill Gates and Paul Allen had the vision to take personal computers to the next level. Therefore, they introduced the MS-DOS in 1981; however, it was very difficult for the person to understand its cryptic commands. Today, Windows has become the most popular and most commonly used operating system technology. And then, Windows released various operating systems such as Windows 95, Windows 98, Windows XP and the latest operating system, Windows 7. Currently, most Windows users use the Windows 10 operating system. Besides the Windows operating system, Apple is another popular operating system built in the 1980s, and this operating system was developed by Steve Jobs, a co-founder of Apple. They named the operating system Macintosh OS or Mac OS.

में पहला विंडो ऑपरेटिंग सिस्टम बनाया। Microsoft Windows OS को पेश करने के बाद, बिल गेट्स और पॉल एलन के पास पर्सनल कंप्यूटर को अगले स्तर पर ले जाने की दृष्टि थी। इसलिए, उन्होंने 1981 में MS-DOS की शुरुआत की; हालाँकि, व्यक्ति के लिए इसकी गुप्त आज्ञाओं को समझना बहुत कठिन था। आज, विंडोज सबसे लोकप्रिय और सबसे अधिक इस्तेमाल की जाने वाली ऑपरेटिंग सिस्टम तकनीक बन गई है। और फिर, विंडोज ने विभिन्न ऑपरेटिंग सिस्टम जैसे विंडोज 95, विंडोज 98, विंडोज एक्सपी और नवीनतम ऑपरेटिंग सिस्टम, विंडोज 7 जारी किए। वर्तमान में, अधिकांश विंडोज उपयोगकर्ता विंडोज 10 ऑपरेटिंग सिस्टम का उपयोग करते हैं। विंडोज ऑपरेटिंग सिस्टम के अलावा, Apple एक और लोकप्रिय ऑपरेटिंग सिस्टम है जिसे 1980 के दशक में बनाया गया था, और इस ऑपरेटिंग सिस्टम को Apple के सह-संस्थापक स्टीव जॉब्स द्वारा विकसित किया गया था। उन्होंने ऑपरेटिंग सिस्टम का नाम Macintosh OS या Mac OS रखा।

Computer Architecture / Block Diagram of CPU

कंप्यूटर आर्किटेक्चर / सीपीयू का ब्लॉक आरेख



Input

All the data received by the computer goes through the input unit. The input unit comprises different devices. Like a mouse, keyboard, scanner, etc. In other words, each of these devices acts as a mediator between the users and the computer.

The data that is to be processed is put through the input unit. The computer accepts the raw data in binary form. It then processes the data, and produces the desired output.

The 3 major functions of the input unit are-

- Take the data to be processed by the user.
- Convert the given data into machine-readable form.

इनपुट

कंप्यूटर द्वारा प्राप्त सभी डेटा इनपुट यूनिट के माध्यम से जाता है। इनपुट यूनिट में विभिन्न डिवाइस शामिल हैं। जैसे माउस, कीबोर्ड, स्कैनर आदि। दूसरे शब्दों में, इनमें से प्रत्येक डिवाइस उपयोगकर्ताओं और कंप्यूटर के बीच मध्यस्थ के रूप में कार्य करता है।

जिस डेटा को प्रोसेस किया जाना है उसे इनपुट यूनिट के माध्यम से डाला जाता है। कंप्यूटर कच्चे डेटा को बाइनरी रूप में स्वीकार करता है। यह तब डेटा को संसाधित करता है, और वांछित आउटपुट उत्पन्न करता है।

इनपुट यूनिट के 3 प्रमुख कार्य हैं-

- उपयोगकर्ता द्वारा संसाधित किए जाने वाले डेटा को लें।
- दिए गए डेटा को मशीन-पठनीय रूप में परिवर्तित करें।
- और फिर, परिवर्तित डेटा को कंप्यूटर की मुख्य मेमोरी में संचारित करें। एकमात्र उद्देश्य उपयोगकर्ता और कंप्यूटर

- And then, transmit the converted data into the main memory of the computer. The sole purpose is to connect the user and the computer. In addition, this creates easy communication between them.

CPU – Central Processing Unit

Central Processing Unit or the CPU, is the brain of the computer. It works the same way a human brain works. As the brain controls all human activities, the CPU too controls all tasks.

Moreover, the CPU conducts all the arithmetical and logical operations in the computer.

Now the CPU comprises of two units, namely – ALU (Arithmetic Logic Unit) and CU (Control Unit). Both of these units work in sync. The CPU processes the data as a whole.

Let us see what particular tasks are assigned to both units.

ALU – Arithmetic Logic Unit

The Arithmetic Logic Unit is made of two terms, arithmetic and logic. There are two major functions that this unit performs.

- Data inserted through the input unit into the primary memory. Performs the basic arithmetical operation on it. Like addition, subtraction, multiplication, and division. It performs all sorts of calculations required on the data. Then sends back data to the storage.
- The unit is also responsible for performing logical operations like, AND, OR, Equal to, Less than, etc. In addition to this it conducts merging, sorting, and selection of the given data.

Control Unit

The control unit as the name suggests is the controller of all the activities/tasks and operations. All this is performed inside the computer.

The memory unit sends a set of instructions to the control unit. Then the control unit in turn converts those instructions. After that these instructions are converted to control signals.

These control signals help in prioritizing and scheduling the activities. Thus, the control unit coordinates the tasks inside the computer in sync with the input and output units.

Memory Unit

All the data that has to be processed or has been processed is stored in the memory unit. The memory unit acts as a hub of all the data. It transmits it to the required part of the computer whenever necessary.

को जोड़ना है। इसके अलावा, यह उनके बीच आसान संचार बनाता है।

सीपीयू - सेंट्रल प्रोसेसिंग यूनिट

सेंट्रल प्रोसेसिंग यूनिट या सीपीयू कंप्यूटर का दिमाग है। यह वैसे ही काम करता है जैसे इंसान का दिमाग काम करता है। जैसे मस्तिष्क सभी मानवीय गतिविधियों को नियंत्रित करता है, वैसे ही सीपीयू भी सभी कार्यों को नियंत्रित करता है।

इसके अलावा, सीपीयू कंप्यूटर में सभी अंकगणितीय और तार्किक संचालन करता है।

अब CPU में दो इकाइयाँ शामिल हैं, अर्थात् - ALU (अरिथमेटिक लॉजिक यूनिट) और CU (कंट्रोल यूनिट)। ये दोनों इकाइयाँ सिंक में काम करती हैं। CPU डेटा को समग्र रूप से प्रोसेस करता है।

आइए देखें कि दोनों इकाइयों को कौन से विशेष कार्य सौंपे गए हैं।

ALU - अंकगणितीय तर्क इकाई

अरिथमेटिक लॉजिक यूनिट दो शब्दों, अंकगणित और तर्क से बना है। यह इकाई दो प्रमुख कार्य करती है।

- इनपुट यूनिट के माध्यम से प्राथमिक मेमोरी में डाला गया डेटा। इस पर बुनियादी अंकगणितीय ऑपरेशन करता है। जैसे जोड़, घटा, गुणा और भाग। यह डेटा पर आवश्यक सभी प्रकार की गणना करता है। फिर डेटा को स्टोरेज में वापस भेजता है।
- इकाई तार्किक संचालन करने के लिए भी जिम्मेदार है, जैसे, और, या, इसके बराबर, इससे कम, आदि। इसके अलावा यह दिए गए डेटा का विलय, छँटाई और चयन करता है।

कंट्रोल यूनिट

नियंत्रण इकाई जैसा कि नाम से पता चलता है, सभी गतिविधियों / कार्यों और संचालन का नियंत्रक है। यह सब कंप्यूटर के अंदर किया जाता है।

मेमोरी यूनिट कंट्रोल यूनिट को निर्देशों का एक सेट भेजती है। फिर नियंत्रण इकाई बदले में उन निर्देशों को परिवर्तित करती है। उसके बाद इन निर्देशों को नियंत्रण संकेतों में बदल दिया जाता है।

ये नियंत्रण संकेत गतिविधियों को प्राथमिकता देने और शेड्यूल करने में मदद करते हैं। इस प्रकार, नियंत्रण इकाई इनपुट और आउटपुट इकाइयों के साथ कंप्यूटर के अंदर के कार्यों का समन्वय करती है।

स्मृति इकाई

सभी डेटा जिन्हें संसाधित किया जाना है या संसाधित किया गया है, स्मृति इकाई में संग्रहीत किया जाता है। मेमोरी यूनिट सभी डेटा के हब के रूप में कार्य करती है। यह जब भी आवश्यक होता है इसे कंप्यूटर के आवश्यक भाग तक पहुंचाता है।

The memory unit works in sync with the CPU. This helps in faster accessing and processing of the data. Thus, making tasks easier and faster.

There are two types of computer memory-

- **Primary memory** – This type of memory cannot store a vast amount of data. Therefore, it is only used to store recent data. The data stored in this is temporary. It can get erased once the power is switched off. Therefore, is also called temporary memory or the main memory. RAM stands for Random Access Memory. It is an example of primary memory. This memory is directly accessible by the CPU. It is used for reading and writing purposes. For data to be processed, it has to be first transferred to the RAM and then to the CPU.
- **Secondary memory** – As explained above, the primary memory stores temporary data. Thus it cannot be accessed in the future. For permanent storage purposes, secondary memory is used. It is also called the permanent memory or the auxiliary memory. The hard disk is an example of secondary memory. Even in a power failure data does not get erased easily.

Output

There is nothing to be amazed by what the output unit is used for. All the information sent to the computer once processed is received by the user through the output unit. Devices like printers, monitors, projector, etc. all come under the output unit.

The output unit displays the data either in the form of a soft copy or hard copy. The printer is for the hard copy. The monitor is for the display. The output unit accepts the data in binary form from the computer. It then converts it into a readable form for the user.

मेमोरी यूनिट सीपीयू के साथ सिंक में काम करती है। यह डेटा की तेजी से पहुंच और प्रसंस्करण में मदद करता है। इस प्रकार, कार्यों को आसान और तेज़ बनाना।

कंप्यूटर मेमोरी दो प्रकार की होती है-

- **प्राइमरी मेमोरी** - इस प्रकार की मेमोरी बहुत अधिक मात्रा में डेटा स्टोर नहीं कर सकती है। इसलिए, इसका उपयोग केवल हाल के डेटा को संग्रहीत करने के लिए किया जाता है। इसमें स्टोर किया गया डेटा अस्थायी होता है। बिजली बंद होने के बाद इसे मिटाया जा सकता है। इसलिए, इसे अस्थायी मेमोरी या मुख्य मेमोरी भी कहा जाता है। RAM का मतलब रैंडम एक्सेस मेमोरी है। यह प्राइमरी मेमोरी का उदाहरण है। यह मेमोरी सीपीयू द्वारा सीधे एक्सेस की जा सकती है। इसका उपयोग पढ़ने और लिखने के उद्देश्यों के लिए किया जाता है। डेटा को संसाधित करने के लिए, इसे पहले RAM और फिर CPU में स्थानांतरित करना होगा।
- **सेकेंडरी मेमोरी** - जैसा कि ऊपर बताया गया है, प्राइमरी मेमोरी अस्थायी डेटा को स्टोर करती है। इस प्रकार इसे भविष्य में एक्सेस नहीं किया जा सकता है। स्थायी भंडारण उद्देश्यों के लिए, द्वितीयक मेमोरी का उपयोग किया जाता है। इसे स्थायी स्मृति या सहायक स्मृति भी कहते हैं। हार्ड डिस्क सेकेंडरी मेमोरी का एक उदाहरण है। बिजली गुल होने पर भी डाटा आसानी से नहीं मिटता।

उत्पादन

आउटपुट यूनिट का उपयोग किस लिए किया जाता है, इससे चकित होने की कोई बात नहीं है। एक बार प्रोसेस हो जाने के बाद कंप्यूटर को भेजी जाने वाली सभी सूचनाएँ आउटपुट यूनिट के माध्यम से उपयोगकर्ता द्वारा प्राप्त की जाती हैं। प्रिंटर, मॉनिटर, प्रोजेक्टर आदि जैसे उपकरण सभी आउटपुट यूनिट के अंतर्गत आते हैं।

आउटपुट यूनिट डेटा को या तो सॉफ्ट कॉपी या हार्ड कॉपी के रूप में प्रदर्शित करता है। प्रिंटर हार्ड कॉपी के लिए है। मॉनिटर डिस्प्ले के लिए है। आउटपुट यूनिट कंप्यूटर से डेटा को बाइनरी रूप में स्वीकार करती है। यह तब इसे उपयोगकर्ता के लिए एक पठनीय रूप में परिवर्तित करता है।

Types of Operating Systems

1. **Single User O.S:** A single-user operating system is a system in which only one user can access the computer system at a time.

Example: MS-DOS

2. **Multi User O.S:** a multi-user operating system has been designed for more than one user to access the computer at one time. Generally, a network is laid down, so that a computer can be remotely used. Mainframes and minicomputers work on multi-user operating systems. Each user is provided with a terminal and all these terminals are connected to a main computer.

Example: Unix, Linux and mainframes

ऑपरेटिंग सिस्टम के प्रकार

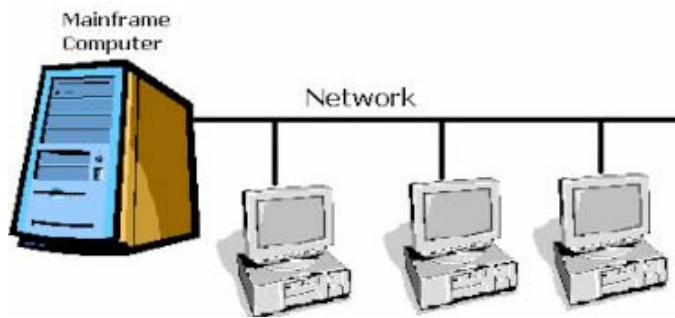
1. **Single User O.S:** single-user ऑपरेटिंग सिस्टम एक ऐसा सिस्टम है जिसमें केवल एक उपयोगकर्ता कंप्यूटर सिस्टम को एक समय में एक्सेस कर सकता है।

उदाहरण: MS-DOS

2. **Multi User O.S:**

एक मल्टी-यूजर ऑपरेटिंग सिस्टम एक समय में कंप्यूटर तक पहुंचने के लिए एक से अधिक उपयोगकर्ता के लिए डिज़ाइन किया गया है। आम तौर पर, एक नेटवर्क बनाया जाता है, जिससे कि कंप्यूटर दूरस्थ रूप से उपयोग किया जा सके। मेनफ्रेम और मिनीकॉम्प्यूटर multi-user ऑपरेटिंग सिस्टम पर काम करते हैं। प्रत्येक user टर्मिनल के साथ प्रदान किया जाता है और ये सभी टर्मिनल मुख्य कंप्यूटर से जुड़े होते हैं।

उदाहरण: यूनिक्स, लिनक्स और मेनफ्रेम



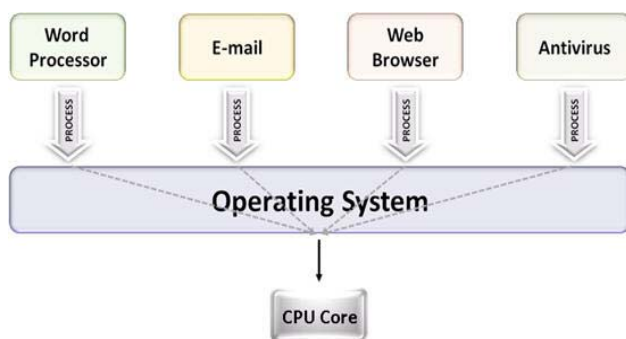
3. Multi Programming O.S: To overcome the problem of underutilization of CPU and main memory, the multiprogramming was introduced.

The multiprogramming is interleaved execution of multiple jobs by the same computer.

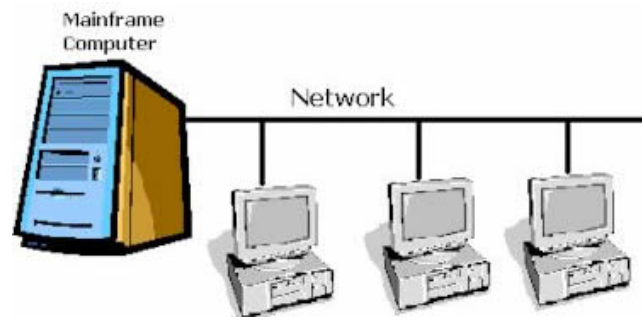
In multiprogramming system, when one program is waiting for I/O transfer; there is another program ready to utilize the CPU. So it is possible for several jobs to share the time of the CPU.

A computer running excels and firefox browser simultaneously is an example of multiprogramming.

4. Multi tasking O.S: Multitasking is the ability of an operating system to execute more than one task simultaneously on a single processor machine. Though we say so but in reality no two tasks on a single processor machine can be executed at the same time. Actually CPU switches from one task to the next task so quickly that appears as if all the tasks are executing at the same time. More than one program can reside into the same CPU at one point of time.



5. Multi Processing O.S: Multiprocessing is the ability of an operating system to execute more than one process simultaneously on a multi processor machine. In this, a computer uses more than one CPU at a time.



3. Multi Programming O.S:

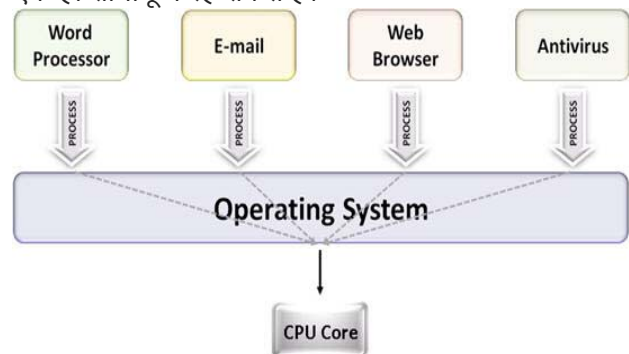
CPU और मुख्य मेमोरी के under utilization की समस्या को दूर करने के लिए, multi-programming शुरू की गई थी। multi-programming एक ही कंप्यूटर द्वारा एक से अधिक प्रोग्रामों के execution में मदद करती है।

मल्टिप्रोग्रामिंग सिस्टम में, जब एक प्रोग्राम I / O के लिए प्रतीक्षा कर रहा होता है तब कोई अन्य प्रोग्राम CPU का उपयोग करने के लिए तैयार रहता है। इस तरह कई सारे प्रोग्राम्स एक साथ CPU का समय share करता है।

एक कंप्यूटर में एक साथ चल रहे excels और firefox browser , multiprogramming का एक उदाहरण है।

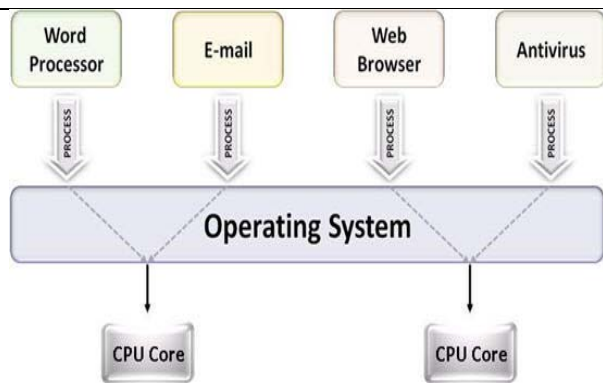
4. Multi tasking O.S:

मल्टीटास्किंग एक प्रोसेसर मशीन पर एक साथ एक से अधिक कार्य चलाने के लिए एक ऑपरेटिंग सिस्टम की क्षमता है। हालांकि हम यह कहते हैं ,लेकिन वास्तविकता में एक प्रोसेसर मशीन पर कोई भी दो कार्य एक ही समय में चलाया नहीं जा सकता है। असल में CPU एक कार्य से दूसरे कार्य को इतनी तेजी से स्विच करता है कि ऐसा लगता है जैसे सभी कार्य एक ही समय में चलाये जा रहे हो । एक से अधिक प्रोग्राम एक समय में एक ही सीपीयू में रह सकते हैं।



5. Multi Processing O.S:

मल्टीप्रोसेसिंग एक प्रोसेसर मशीन पर एक साथ एक से अधिक process को चलाने के लिए एक ऑपरेटिंग सिस्टम की क्षमता कहलाती है। इसमें, एक कंप्यूटर एक समय में एक से अधिक CPU उपयोग करता है ।



6. Real Time O.S:

A real-time system is defined as a data processing system in which the time interval required to process and respond to inputs is so small that it controls the environment. The time taken by the system to respond to an input and display of required updated information is termed as the **response time**. Real-time systems are used when there are rigid time requirements on the operation of a processor. A real-time operating system must have well-defined, fixed time constraints, otherwise the system will fail. For example, Scientific experiments, medical image systems, industrial control systems, weapon systems, robots, air traffic control systems, etc.

There are two types of real-time operating systems.

(a) Hard real-time systems

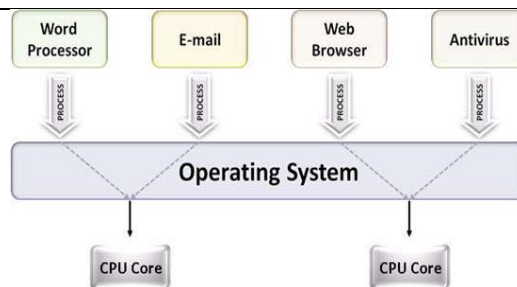
Hard real-time systems guarantee that critical tasks complete on time. In hard real-time systems, secondary storage is limited or missing and the data is stored in ROM. In these systems, virtual memory is almost never found.

(b) Soft real-time systems

Soft real-time systems are less restrictive. A critical real-time task gets priority over other tasks and retains the priority until it completes. Soft real-time systems have limited utility than hard real-time systems. For example, multimedia, virtual reality, Advanced Scientific Projects likes undersea exploration and planetary rovers, etc.

7. Time Sharing O.S:

A **time sharing system** allows many users to share the computer resources simultaneously. Each user uses the resources of the mainframe -i.e. memory, CPU etc. The users feel that they are exclusive user of the CPU. The time sharing systems were developed to provide an interactive use of the computer system. A time shared system uses CPU scheduling and multiprogramming to provide each user with a small portion of a time-shared computer. The OS allocates a set of time to each user. When this time is expired, it passes control to the next user on the system. The time allowed is extremely small and the users are given the impression that they each have their own CPU and they are the sole owner of the CPU. This short period



6. Real Time O.S:

एक real-time system को एक डेटा प्रोसेसिंग सिस्टम के रूप में परिभाषित किया जाता है जिसमें इनपुट की प्रक्रिया और जवाब देने के लिए आवश्यक समय अंतराल इतनी छोटी है कि यह सिस्टम को नियंत्रित करता है।

सिस्टम द्वारा आवश्यक समय-समय पर अपडेट की गई जानकारी के इनपुट और प्रदर्शन का जवाब देने के लिए लिया गया समय प्रतिक्रिया समय कहा जाता है।

रीयल-टाइम सिस्टम का उपयोग किया जाता है जब प्रोसेसर के संचालन पर कठोर समय की आवश्यकता होती है।

एक वास्तविक समय ऑपरेटिंग सिस्टम में अच्छी तरह से परिभाषित और समय निश्चित होना होनी चाहिए, अन्यथा सिस्टम विफल हो जाएगा।

उदाहरण के लिए, वैज्ञानिक प्रयोग, चिकित्सा छवि प्रणाली, औद्योगिक नियंत्रण प्रणाली, हथियार प्रणाली, रोबोट, हवाई यातायात नियंत्रण प्रणाली आदि।

real-time system के दो प्रकार होते हैं:

(a) Hard real-time systems

हार्ड रीयल-टाइम सिस्टम गारंटी देते हैं कि महत्वपूर्ण कार्य समय पर पूर्ण हो। हार्ड रीयल-टाइम सिस्टम में, सेकंडरी स्टोरेज सीमित या नहीं होते हैं और डेटा को रोम में संग्रहीत किया जाता है। इन पद्धतियों में, virtual स्मृति लगभग कभी नहीं होती है।

(b) Soft real-time systems

सॉफ्ट रीयल-टाइम सिस्टम कम प्रतिबंधात्मक हैं एक महत्वपूर्ण वास्तविक समय के कार्य को अन्य कार्यों पर प्राथमिकता दी जाती है और प्राथमिकता तब तक बरकरार रखती है जब तक कि यह पूर्ण नहीं हो जाता। सॉफ्ट रीयल-टाइम सिस्टम में हार्ड रीयल-टाइम सिस्टम की तुलना में सीमित सुविधाएं हैं

उदाहरण के लिए, मल्टीमीडिया, आभासी वास्तविकता, उन्नत वैज्ञानिक प्रोजेक्ट्स undersea अन्वेषण और ग्रहों के रोवर्स आदि।

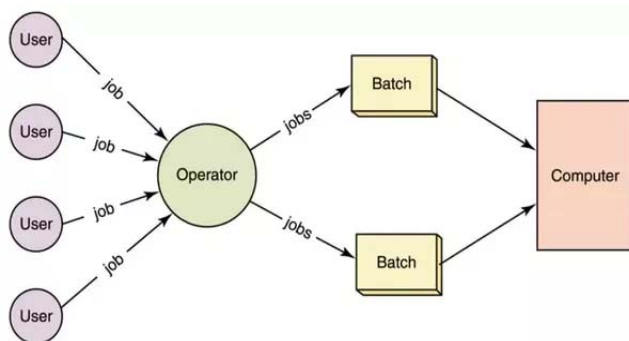
7. Time Sharing O.S:

एक Time Sharing O.S में कई सारे users को एक साथ computer resources को share करने की सुविधा प्रदान करता है. हर एक user, mainframe computer के resources जैसे की memory, CPU आदि उपयोग करते है. हर एक user को यही लगता है की वह CPU का exclusive मालिक है. Time Sharing का विकास कंप्यूटर में interactive अवधारणा को पूरा करना था. इस सिस्टम में CPU Scheduling और multi programming का इस्तेमाल किया जाता है जिसमे हर एक यूजर को एक निश्चित समय दिया जाता है. OS इस समय को निर्धारित करता है. जब समय समाप्त हो जाता है तो अगले यूजर को समय दिया जाता है. यह समय बहुत कम होता है और यूजर लगता है CPU उनके पास

of time during that a user gets attention of the CPU; is known as a *time slice* or a *quantum*.

8. Batch Processing O.S:

The users of a batch operating system do not interact with the computer directly. Each user prepares his job on an off-line device like punch cards and submits it to the computer operator. To speed up processing, jobs with similar needs are batched together and run as a group. The programmers leave their programs with the operator and the operator then sorts the programs with similar requirements into batches.



Advantages:

1. Suppose a job takes a very long time (1 day or so). Then, such processes can be performed even in the **absence** of humans.
2. They don't require any **special** hardware and system support to input data.

Disadvantages:

1. It is very difficult to **debug** batch systems.
2. Lack of **interaction** between user and operating system.
3. Suppose an error occurs in one of the jobs of a batch. Then, all the **remaining** jobs get affected i.e; they have to wait until the error is resolved.

9. Parallel Processing O.S:

Parallel Processing Systems are designed to speed up the execution of programs by dividing the program into multiple fragments and processing these fragments simultaneously. Such systems are multiprocessor systems also known as tightly coupled systems.

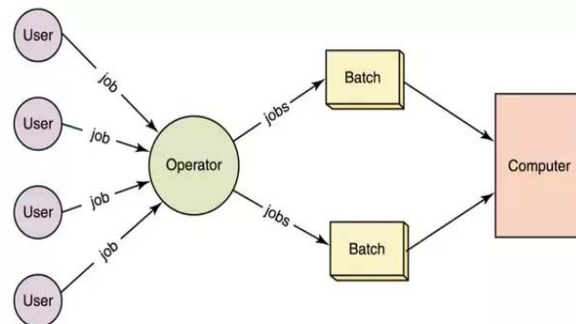
10. Distributed O.S: Distributed Operating System is a system in which the operating system is distributed through several workstations. All nodes have their own memory and own copy of operating systems.

Distributed systems are a collection of several separate(individual) systems which **communicate** (through a LAN or WAN) **and cooperate** with each other (using some software) in order to provide the users, access to various resources that the system maintains. One **important** point to note about distributed systems is that they are **loosely-coupled** i.e; hardware and software

है. यह समय जिसमें एक यूजर को CPU का ध्यान मिलता है उसे time slice या quantum कहते हैं.

8. Batch Processing O.S:

बैच ऑपरेटिंग सिस्टम के उपयोगकर्ता सीधे कंप्यूटर से इंटरैक्ट नहीं करते हैं प्रत्येक उपयोगकर्ता अपने प्रोग्राम को एक ऑफ लाइन डिवाइस पर पंच कार्ड की तरह तैयार करता है और उसे कंप्यूटर ऑपरेटर के पास भेजता है। processing को गति देने के लिए, समान जरूरतों वाले काम को एक समूह के रूप में चलाया जाता है। प्रोग्रामर अपने कार्यक्रम को ऑपरेटर के साथ छोड़ देते हैं तो बैचों में समान आवश्यकताओं वाले कार्यक्रमों को चलाते हैं।



Advantages

1. मान लीजिए एक कार्यक्रम में बहुत समय लगता है (1 दिन)। फिर, ऐसी प्रक्रियाएं मनुष्यों की अनुपस्थिति में भी की जा सकती हैं
2. इन्हें इनपुट डेटा के लिए किसी विशेष हार्डवेयर और सिस्टम समर्थन की आवश्यकता नहीं है।

Disadvantages

1. बैच सिस्टम डिबग करना बहुत कठिन है।
2. उपयोगकर्ता और ऑपरेटिंग सिस्टम के बीच बातचीत का अभाव।
3. मान लें कि एक बैच के किसी एक कार्य में कोई त्रुटि आती जाती है। इससे, शेष सभी कार्यक्रम प्रभावित हो जाते हैं और उन्हें त्रुटि का समाधान होने तक इंतजार करना पड़ता है।

9. Parallel Processing O.S:

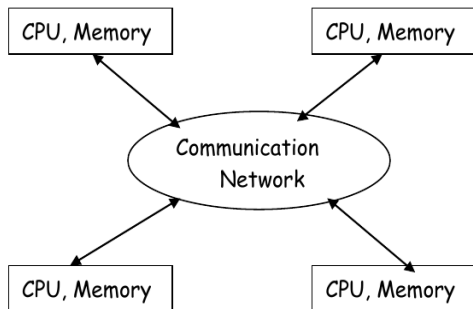
समानांतर प्रसंस्करण सिस्टम को कई टुकड़ों में प्रोग्राम को विभाजित करके और इन टुकड़ों को एक साथ प्रोसेस करके प्रोग्राम के निष्पादन को गति देने के लिए डिज़ाइन किया गया है। इस तरह के सिस्टम मल्टीप्रोसेसर सिस्टम हैं जो tightly coupled सिस्टम के रूप में भी जाना जाता है।

10. Distributed O.S:

Distributed ऑपरेटिंग सिस्टम एक ऐसा सिस्टम है जिसमें ऑपरेटिंग सिस्टम कई वर्कस्टेशन के माध्यम से वितरित किया जाता है। सभी नोड्स की अपनी स्मृति और ऑपरेटिंग सिस्टम की अपनी प्रति होती है।

Distributed प्रणाली कई अलग-अलग (व्यक्तिगत) प्रणालियों का एक संग्रह है जो संचार (एक LAN या WAN के माध्यम से) और एक दूसरे के साथ सहयोग करते हैं (कुछ सॉफ्टवेयर का उपयोग करते हुए) उपयोगकर्ताओं को सिस्टम के विभिन्न संसाधनों की मदद मिल सके। Distributed प्रणालियों के बारे में ध्यान देने योग्य एक महत्वपूर्ण बिंदु यह है कि वे loosely-

may communicate with each other but they need not depend upon each other.



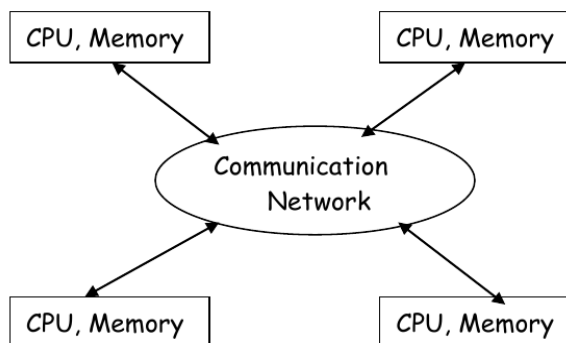
Advantages of distributed operating systems: -

- Give more performance than single system
- If one pc in distributed system malfunction or corrupts then other node or pc will take care of
- More resources can be added easily
- Resources like printers can be shared on multiple pc's

Disadvantages of distributed operating systems: -

- Security problem due to sharing
- Some messages can be lost in the network system
- Bandwidth is another problem if there is large data then all network wires to be replaced which tends to become expensive
- Overloading is another problem in distributed operating systems
- If there is a database connected on local system and many users accessing that database through remote or distributed way, then performance become slow
- The databases in network operating is difficult to administrate than single user system.

coupled हैं यानि की हार्डवेयर और सॉफ्टवेयर एक दूसरे के साथ संवाद कर सकते हैं लेकिन उन्हें एक-दूसरे पर निर्भर नहीं होते हैं।



Advantages

- सिंगल सिस्टम से अधिक प्रदर्शन देते हैं।
- यदि एक पीसी वितरित प्रणाली में खराबी या भ्रष्ट हो जाये तो अन्य नोड या पीसी इसका ध्यान रखता है।
- अधिक संसाधनों को आसानी से जोड़ा जा सकता है
- प्रिंटर जैसे संसाधन एकाधिक पीसी पर साझा किए जा सकते हैं

Disadvantages

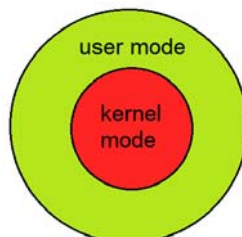
- साझा करने के कारण सुरक्षा समस्या
- नेटवर्क सिस्टम में कुछ संदेश खो सकते हैं
- Bandwidth एक और समस्या है अगर बड़े डेटा है तो सभी नेटवर्क तारों की जगह होगी जो महंगा हो जाती है
- Distributed ऑपरेटिंग सिस्टम में ओवरलोडिंग एक और समस्या है
- यदि स्थानीय सिस्टम पर एक डाटाबेस जुड़ा हुआ है और बहुत से उपयोगकर्ता दूरस्थ या वितरित तरीके के माध्यम से उस डाटाबेस तक पहुंच रहे हैं, तो प्रदर्शन धीमा हो जाता है
- नेटवर्क ऑपरेटिंग में डाटाबेस एकल यूजर सिस्टम से control करना मुश्किल है

Introduction to System Calls

To understand system calls, first one needs to understand the difference between **kernel mode** and **user mode** of a CPU. Every modern operating system supports these two modes.

सिस्टम कॉल का परिचय

सिस्टम कॉल को समझने के लिए, पहले कर्नेल (kernel) मोड और सीपीयू के उपयोगकर्ता(user) मोड के बीच अंतर को समझने की आवश्यकता है। प्रत्येक आधुनिक ऑपरेटिंग सिस्टम इन दो तरीकों का समर्थन करता है।



Kernel Mode

- When CPU is in **kernel mode**, the code being executed can access any memory address and any hardware resource.
- Hence kernel mode is a very privileged and powerful mode.
- If a program crashes in kernel mode, the entire system will be halted.

User Mode

- When CPU is in **user mode**, the programs don't have direct access to memory and hardware resources.
- In user mode, if any program crashes, only that particular program is halted.
- That means the system will be in a safe state even if a program in user mode crashes.
- Hence, most programs in an OS run in user mode.

System Call

When a program in user mode requires access to RAM or a hardware resource, it must ask the kernel to provide access to that resource. This is done via something called a **system call**.

When a program makes a system call, the mode is switched from user mode to kernel mode. This is called a **context switch**.

Then the kernel provides the resource which the program requested. After that, another context switch happens which results in change of mode from kernel mode back to user mode.

Generally, system calls are made by the user level programs in the following situations:

- Creating, opening, closing and deleting files in the file system.
- Creating and managing new processes.
- Creating a connection in the network, sending and receiving packets.
- Requesting access to a hardware device, like a mouse or a printer.

कर्नेल मोड

- जब सीपीयू कर्नेल मोड में होता है, तो निष्पादित कोड किसी भी स्मृति पते और किसी भी हार्डवेयर संसाधन तक पहुंच सकता है।
- इसलिए कर्नेल मोड एक बहुत ही विशेषाधिकार और शक्तिशाली मोड है।
- यदि कोई प्रोग्राम कर्नेल मोड में क्रैश हो जाता है, तो पूरी प्रणाली रुक जाएगी।

उपयोगकर्ता मोड

- जब सीपीयू उपयोगकर्ता मोड में होता है, तो प्रोग्राम में स्मृति और हार्डवेयर संसाधनों तक सीधे पहुंच नहीं होती है।
- उपयोगकर्ता मोड में, यदि कोई प्रोग्राम क्रैश हो जाता है, तो केवल उस विशेष प्रोग्राम को रोक दिया जाता है।
- इसका मतलब है कि सिस्टम एक सुरक्षित स्थिति में होगा भले ही उपयोगकर्ता मोड में कोई प्रोग्राम क्रैश हो।
- इसलिए, ओएस में अधिकांश प्रोग्राम उपयोगकर्ता मोड में चलते हैं।

सिस्टम कॉल

जब उपयोगकर्ता मोड में किसी प्रोग्राम को रैम या हार्डवेयर संसाधन तक पहुंच की आवश्यकता होती है, तो उसे कर्नेल से उस संसाधन तक पहुंच प्रदान करने के लिए कहना चाहिए। यह सिस्टम कॉल के माध्यम से किया जाता है।

जब कोई प्रोग्राम सिस्टम कॉल करता है, तो मोड को उपयोगकर्ता मोड से कर्नेल मोड में स्विच किया जाता है। इसे एक संदर्भ स्विच (context switch) कहा जाता है।

फिर कर्नेल संसाधन प्रदान करता है जिसे प्रोग्राम ने अनुरोध किया था। उसके बाद, एक और संदर्भ स्विच होता है जिसके परिणामस्वरूप मोड कर्नेल मोड से उपयोगकर्ता मोड में परिवर्तन होता है।

आम तौर पर, निम्न स्थितियों में उपयोगकर्ता स्तर के प्रोग्राम द्वारा सिस्टम कॉल किए जाते हैं:

- फाइल सिस्टम में फाइलें बनाना, खोलना, बंद करना और हटाना।
- नई प्रक्रियाओं का निर्माण और प्रबंधन।
- नेटवर्क में कनेक्शन बनाना, पैकेट भेजना और प्राप्त करना।
- माउस या प्रिंटर की तरह हार्डवेयर डिवाइस तक पहुंच का अनुरोध करना।

What is the context switching in the operating system?

The Context switching is a technique or method used by the operating system to switch a process from one state to another to execute its function using CPUs in the system. When switching perform in the system, it stores the old running process's status in the form of registers and assigns the CPU to a new process to execute its tasks. While a new process is running in the system, the previous process must wait in a ready queue. The execution of the old process starts at that point where another process stopped it. It defines the characteristics of a multitasking operating system in which multiple processes shared the same CPU to perform multiple tasks without the need for additional processors in the system.

The need for Context switching

A context switching helps to share a single CPU across all processes to complete its execution and store the system's tasks status. When the process reloads in the system, the execution of the process starts at the same point where there is conflicting.

Following are the reasons that describe the need for context switching in the Operating system.

1. The switching of one process to another process is not directly in the system. A context switching helps the operating system that switches between the multiple processes to use the CPU's resource to accomplish its tasks and store its context. We can resume the service of the process at the same point later. If we do not store the currently running process's data or context, the stored data may be lost while switching between processes.
2. If a high priority process falls into the ready queue, the currently running process will be shut down or stopped by a high priority process to complete its tasks in the system.
3. If any running process requires I/O resources in the system, the current process will be switched by another process to use the CPUs. And when the I/O requirement is met, the old process goes into a ready state to wait for its execution in the CPU. Context switching stores the state of the process to resume its tasks in an operating system. Otherwise, the process needs to restart its execution from the initials level.
4. If any interrupts occur while running a process in the operating system, the process status is saved as registers using context switching. After resolving the interrupts, the process switches from a wait state to a ready state to resume its execution at the same point later, where the operating system interrupted occurs.
5. A context switching allows a single CPU to handle multiple process requests

ऑपरेटिंग सिस्टम में संदर्भ स्विचिंग क्या है?

संदर्भ स्विचिंग एक तकनीक या विधि है जिसका उपयोग ऑपरेटिंग सिस्टम द्वारा सिस्टम में सीपीयू का उपयोग करके अपने कार्य को निष्पादित करने के लिए एक राज्य से दूसरे राज्य में एक प्रक्रिया को स्विच करने के लिए किया जाता है। सिस्टम में प्रदर्शन स्विच करते समय, यह पुरानी चल रही प्रक्रिया की स्थिति को रजिस्ट्रों के रूप में संग्रहीत करता है और सीपीयू को अपने कार्यों को निष्पादित करने के लिए एक नई प्रक्रिया को असाइन करता है। जबकि सिस्टम में एक नई प्रक्रिया चल रही है, पिछली प्रक्रिया को तैयार कतार में प्रतीक्षा करनी चाहिए। पुरानी प्रक्रिया का निष्पादन उस बिंदु से शुरू होता है जहां दूसरी प्रक्रिया ने इसे रोक दिया था। यह एक मल्टीटास्किंग ऑपरेटिंग सिस्टम की विशेषताओं को परिभाषित करता है जिसमें सिस्टम में अतिरिक्त प्रोसेसर की आवश्यकता के बिना कई प्रक्रियाओं को एक ही सीपीयू को कई कार्यों को करने के लिए साझा किया जाता है।

प्रसंग स्विचिंग की आवश्यकता

एक संदर्भ स्विचिंग इसके निष्पादन को पूरा करने और सिस्टम के कार्यों की स्थिति को संग्रहीत करने के लिए सभी प्रक्रियाओं में एक एकल सीपीयू को साझा करने में मदद करता है। जब प्रक्रिया सिस्टम में पुनः लोड होती है, तो प्रक्रिया का निष्पादन उसी बिंदु पर शुरू होता है जहां परस्पर विरोधी है।

निम्नलिखित कारण हैं जो ऑपरेटिंग सिस्टम में संदर्भ स्विचिंग की आवश्यकता का वर्णन करते हैं।

1. एक प्रक्रिया को दूसरी प्रक्रिया में बदलना सीधे सिस्टम में नहीं है। एक संदर्भ स्विचिंग ऑपरेटिंग सिस्टम की मदद करता है जो अपने कार्यों को पूरा करने और इसके संदर्भ को संग्रहीत करने के लिए सीपीयू के संसाधन का उपयोग करने के लिए कई प्रक्रियाओं के बीच स्विच करता है। हम बाद में उसी बिंदु पर प्रक्रिया की सेवा फिर से शुरू कर सकते हैं। यदि हम वर्तमान में चल रही प्रक्रिया के डेटा या संदर्भ को संग्रहीत नहीं करते हैं, तो प्रक्रियाओं के बीच स्विच करते समय संग्रहीत डेटा खो सकता है।
2. यदि कोई उच्च प्राथमिकता प्रक्रिया तैयार कतार में आती है, तो सिस्टम में अपने कार्यों को पूरा करने के लिए वर्तमान में चल रही प्रक्रिया को उच्च प्राथमिकता प्रक्रिया द्वारा बंद या बंद कर दिया जाएगा।
3. यदि किसी चल रही प्रक्रिया को सिस्टम में I/O संसाधनों की आवश्यकता होती है, तो वर्तमान प्रक्रिया को CPU का उपयोग करने के लिए दूसरी प्रक्रिया द्वारा स्विच किया जाएगा। और जब I/O आवश्यकता पूरी हो जाती है, तो पुरानी प्रक्रिया CPU में इसके निष्पादन की प्रतीक्षा करने के लिए तैयार स्थिति में चली जाती है। संदर्भ स्विचिंग एक ऑपरेटिंग सिस्टम में अपने कार्यों को फिर से शुरू करने के लिए प्रक्रिया की स्थिति को संग्रहीत करता है। अन्यथा, प्रक्रिया को प्रारंभिक स्तर से इसके निष्पादन को पुनराारंभ करने की आवश्यकता है।
4. यदि ऑपरेटिंग सिस्टम में किसी प्रक्रिया को चलाने के दौरान कोई रुकावट आती है, तो प्रक्रिया की स्थिति को संदर्भ स्विचिंग का उपयोग करके रजिस्टर के रूप में सहेजा जाता है। इंटरप्ट को हल करने के बाद, प्रक्रिया प्रतीक्षा स्थिति से तैयार स्थिति में बदल जाती है ताकि बाद में उसी बिंदु पर इसके निष्पादन को फिर से शुरू किया जा सके, जहां ऑपरेटिंग सिस्टम बाधित होता है।

simultaneously without the need for any additional processors.

6. **Example of Context Switching**
7. Suppose that multiple processes are stored in a Process Control Block (PCB). One process is running state to execute its task with the use of CPUs. As the process is running, another process arrives in the ready queue, which has a high priority of completing its task using CPU. Here we used context switching that switches the current process with the new process requiring the CPU to finish its tasks. While switching the process, a context switch saves the status of the old process in registers. When the process reloads into the CPU, it starts the execution of the process when the new process stops the old process. If we do not save the state of the process, we have to start its execution at the initial level. In this way, context switching helps the operating system to switch between the processes, store or reload the process when it requires executing its tasks.

Context switching triggers

Following are the three types of context switching triggers as follows.

1. **Interrupts:** A CPU requests for the data to read from a disk, and if there are any interrupts, the context switching automatic switches a part of the hardware that requires less time to handle the interrupts.
2. **Multitasking:** A context switching is the characteristic of multitasking that allows the process to be switched from the CPU so that another process can be run. When switching the process, the old state is saved to resume the process's execution at the same point in the system.
3. **Kernel/User switch:** It is used in the operating systems when switching between the user mode, and the kernel/user mode is performed.

5. एक संदर्भ स्विचिंग एक एकल सीपीयू को बिना किसी अतिरिक्त प्रोसेसर की आवश्यकता के एक साथ कई प्रक्रिया अनुरोधों को संभालने की अनुमति देता है।

6. प्रसंग स्विचिंग का उदाहरण

7. मान लीजिए कि एक प्रक्रिया नियंत्रण ब्लॉक (पीसीबी) में कई प्रक्रियाएं संग्रहीत हैं। सीपीयू के उपयोग के साथ अपने कार्य को निष्पादित करने के लिए एक प्रक्रिया चल रही है। जैसे ही प्रक्रिया चल रही है, एक और प्रक्रिया तैयार कतार में आती है, जिसमें सीपीयू का उपयोग करके अपने कार्य को पूरा करने की उच्च प्राथमिकता होती है। यहां हमने संदर्भ स्विचिंग का उपयोग किया है जो वर्तमान प्रक्रिया को नई प्रक्रिया के साथ बदल देता है जिसके लिए सीपीयू को अपने कार्यों को पूरा करने की आवश्यकता होती है। प्रक्रिया को स्विच करते समय, एक संदर्भ स्विच पुरानी प्रक्रिया की स्थिति को रजिस्ट्रों में सहेजता है। जब प्रक्रिया सीपीयू में पुनः लोड हो जाती है, तो यह प्रक्रिया का निष्पादन शुरू कर देता है जब नई प्रक्रिया पुरानी प्रक्रिया को रोक देती है। यदि हम प्रक्रिया की स्थिति को नहीं सहेजते हैं, तो हमें प्रारंभिक स्तर पर इसका निष्पादन शुरू करना होगा। इस तरह, संदर्भ स्विचिंग ऑपरेटिंग सिस्टम को प्रक्रियाओं के बीच स्विच करने, प्रक्रिया को स्टोर करने या फिर से लोड करने में मदद करता है जब उसे अपने कार्यों को निष्पादित करने की आवश्यकता होती है।

प्रसंग स्विचिंग ट्रिगर

निम्नलिखित तीन प्रकार के संदर्भ स्विचिंग ट्रिगर निम्नानुसार हैं।

1. **व्यवधान:** एक सीपीयू डिस्क से डेटा को पढ़ने के लिए अनुरोध करता है, और यदि कोई व्यवधान होता है, तो संदर्भ स्विचिंग हार्डवेयर के एक हिस्से को स्वचालित रूप से स्विच कर देता है जिसे इंटरप्ट को संभालने के लिए कम समय की आवश्यकता होती है।
2. **मल्टीटास्किंग:** एक संदर्भ स्विचिंग मल्टीटास्किंग की विशेषता है जो प्रक्रिया को सीपीयू से स्विच करने की अनुमति देती है ताकि दूसरी प्रक्रिया को चलाया जा सके। प्रक्रिया को स्विच करते समय, सिस्टम में उसी बिंदु पर प्रक्रिया के निष्पादन को फिर से शुरू करने के लिए पुरानी स्थिति को सहेजा जाता है।
3. **कर्नेल/उपयोगकर्ता स्विच:** इसका उपयोग ऑपरेटिंग सिस्टम में उपयोगकर्ता मोड के बीच स्विच करते समय किया जाता है, और कर्नेल/उपयोगकर्ता मोड निष्पादित किया जाता है।