# राजस्थान
# सूचना सहायक परीक्षा
## (FOR RSMSSB & IBPS)

Golden Edition

PART - 1

INFUSION NOTES
WHEN ONLY THE BEST WILL DO

COMPLETE TECHNICAL PART

RSMSSB एवं IBPS द्वारा आयोजित –

# राजस्थान

# सूचना

# सहायक

## सीधी भर्ती परीक्षा

First Hinglish Portmanteau Golden Edition.

**For Competitive Exams.**

*H. Choudhary.*

**प्रकाशक –**

**इन्फ्यूजन नोट्स ®**
**बस स्टैंड, 1st फ्लोर, शॉप न. 03 प्रिंटर्स नगर,**
**टोंक रोड, सीतापुरा, सांगानेर, जयपुर 302029.**
**फोन न. –** 9887809083.
**https://**www.infusionnotes.com

INFUSION NOTES
WHEN ONLY THE BEST WILL DO

**डिजिटल पार्टनर ® –**

MyBookLo
Your Digital Library App

https://mybooklo.com/courses

**कंटेंट राइटर एवं लेखक © –**

**Under Guidance & Supervision of -**
**H.Chaudhary (Harry Sir / IT Guru)**
**YouTube/@ITEXAMGURUJI**
**Twitter.com/it_exam_guruji**

IT Adda : IT EXAM GURUJI
GET IT ON
Google Play

**Publisher & Author Note:**

Every possible effort has been made to ensure that the information contained in this book is accurate, and the publisher or the Author can't accept responsibility for any errors or omissions, however caused.

All liability for loss, disappointment, negligence or other damage caused by the reliance of the Technical Programming or other information contained in this book, of in the event of bankruptcy or liquidation or cessation of trade of any company, individual; or firm mentioned, is hereby excluded.

All other marks are property of their respective owners. The examples of companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted herein are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

The author and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

This book expresses the author views and opinions. The information contained in this book is provided without any express, statutory, or implied warranties. Neither the authors, and Publisher, nor its resellers, or distributors will be held liable for any damages caused or alleged to be caused either directly or indirectly by this book.

**Copyright © H.Choudhary.**
All rights reserved.

**Warning-**

This book or any portion thereof may not be reproduced or used in any manner whatsoever without the express written permission of the author H. Choudhary. No person/publisher /institute should use full/part of the text/design/questions/material of the book. If any body (person/publisher/institute/xerox shop./website/Blog) found at fault legal action will be taken accordingly as per indian and international copyright act.

ISBN: 978-93-5786-101-4
ISBN 978-93-5786-101-4
9 789357 861014

# सूचना सहायक - 2023 के नोट्स खरीदने के लिए नीचे दी गयी लिंक पर क्लिक करें

whatsapp करें – https://wa.link/j53pta

online order करें – https://bit.ly/3xjbC7f

## या फिर इस नंबर पर कॉल करके अपना order बुक करवाएं -

# 9887809083

FREE SAMPLE

# TOC


| SR | SUB | SUBJECT LIST OR TOPICS | PAGE# |
|---|---|---|---|
| **(1) Introduction To Operating System.** | | | |
| **1.** | **1.1** | ऑपरेटिंग सिस्टम का परिचय | **1** |
| | **1.2** | टाइप्स ऑफ सॉफ्टवेयर | **2** |
| | **1.3** | कर्नेल– बॉस प्रोग्राम | **3** |
| | **1.4** | शैल– इंटरफेस प्रोग्राम | **3** |
| | **1.5** | ओ.एस के फंक्शन्स एवं सर्विसेज | **4** |
| | **1.6** | जॉब पूल एवं प्रोसेस कांसेप्ट | **5** |
| | **1.7** | प्रोसेस की विभिन्न स्टेट | **5** |
| | **1.8** | डेडलॉक कांसेप्ट | **5** |
| | **1.9** | रैम एवं ड्रिल्स | **6** |
| | **1.10** | टाइप्स ऑफ ऑपरेटिंग सिस्टम | **7** |
| | **1.11** | आर्किटेक्चर ऑफ ऑपरेटिंग सिस्टम | **11** |
| | **1.12** | मोनो, माइक्रो, एक्सओ एवं हाइब्रिड कर्नेल | **11-13** |
| | **1.13** | वर्चुअल मशीन कांसेप्ट | **13** |
| | **1.14** | सिस्टम कॉल्स कांसेप्ट | **14** |
| | **1.15** | डिवाइस ड्राइवर्स | **15** |
| | **1.16** | बॉयोस कांसेप्ट | **15** |
| | **1.17** | बूटस्ट्रैप कांसेप्ट | **16** |
| | **1.18** | बूट सीक्वेंस कांसेप्ट एवं फर्मवेयर | **16** |
| | **1.19** | मॉडल टेस्ट पेपर ऑफ ओ.एस | **17-21** |
| | | | |
| **(2) Introduction To File System.** | | | |
| **2.** | **2.1** | फाइल सिस्टम का परिचय | **22** |
| | **2.2** | फाइल कन्ट्रोल ब्लॉक | **22** |
| | **2.3** | डिवाइस ड्राइवर एवं बूट कन्ट्रोल ब्लॉक | **22** |
| | **2.4** | वॉल्यूम कन्ट्रोल ब्लॉक | **23** |
| | **2.5** | फाइल एक्सटेंशन्स | **23** |
| | **2.6** | फाइल ऐट्रिब्यूट्स | **25** |
| | **2.7** | फाइल ऑपरेशन्स | **26** |
| | **2.8** | फाइल एलोकेशन मेथड्स | **26** |
| | **2.9** | कॉन्टीग्योअस एलोकेशन | **27** |
| | **2.10** | लिंक्ड एलोकेशन | **28** |
| | **2.11** | इंडेक्स्ड एलोकेशन | **29** |
| | **2.12** | डायरेक्टरी सिस्टम | **30** |
| | **2.13** | सिंगल लेवल डायरेक्टरी स्ट्रक्चर | **31** |
| | **2.14** | टू लेवल डायरेक्टरी स्ट्रक्चर | **32** |
| | **2.15** | ट्री डायरेक्टरी स्ट्रक्चर | **32** |
| | **2.16** | ग्राफ डायरेक्टरी स्ट्रक्चर | **33** |
| | **2.17** | मॉडल टेस्ट पेपर ऑफ फाइल सिस्टम | **App** |

| SR | SUB | SUBJECT LIST OR TOPICS | PAGE# |
|---|---|---|---|
| | | **(3) Exploring Windows O.S** | |
| 3. | 3.1 | एक्सप्लोरिंग विंडोज | 34 |
| | 3.2 | विभिन्न वर्जन्स ऑफ विंडोज (विंडोज 1.0 से 11 तक ) | 35 |
| | 3.3 | मॉय कंप्यूटर एवं रिसाइकिल बिन | 36 |
| | 3.4 | टास्कबार एवं स्टार्ट मेन्यू | 36 |
| | 3.5 | फाइल एक्स्प्लोरर | 37 |
| | 3.6 | क्रिएटिंग फाइल एंड फोल्डर्स | 37 |
| | 3.7 | डेस्कटॉप एवं टाइलबार | 37 |
| | 3.8 | विंडोज यूटिलिटीज – पेंट एवं क्लिपबोर्ड, क्लिपबोर्ड हाईजेकिंग | 38 |
| | 3.9 | विंडोज नोटपैड टेक्स्ट एडिटर | 39 |
| | 3.10 | विंडोज वर्डपैड – बेसिक वर्ड प्रोसेसर | 39 |
| | 3.11 | विंडोज कैलकुलेटर एवं स्निप्पिंग टूल और मैग्नीफायर | 40 |
| | 3.12 | एप्लीकेशन सॉफ्टवेयर एवं उनके उदाहरण | 40 |
| | 3.13 | सिस्टम सॉफ्टवेयर एवं उनके उदाहरण | 41 |
| | 3.14 | डिवाइस ड्राइवर्स एवं सिस्टम यूटिलिटीज | 41 |
| | 3.15 | स्टोरेज डिवाइस मैनेजमेंट यूटिलिटीज | 41 |
| | 3.16 | फाइल मैनेजमेंट यूटिलिटीज | 41 |
| | 3.17 | मॉलिसियस सॉफ्टवेयर | 42 |
| | 3.18 | प्रोग्रामिंग टूल्स एवं IDEs | 42 |
| | 3.19 | प्लगइन्स एवं एक्सटेंशन्स | 42 |
| | 3.20 | एंबेडेड सॉफ्टवेयर एवं माइक्रो कोड | 42 |
| | 3.21 | कम्पाइलर (क्रॉस, बूटस्ट्रैप, डिकम्पाइलर, ट्रांसपाइलर) | 43 |
| | 3.22 | असेम्बलर एवं इंटरप्रेटर | 44 |
| | 3.23 | लिंकर एवं लोडर | 44 |
| | | | |
| | | **(4) Introduction To UNIX, LINUX, DOS Systems.** | |
| 4. | 4.1 | UNIX ओएस का परिचय | 47 |
| | 4.2 | BSD UNIX, GNU एवं ARPANET. | 48 |
| | 4.3 | बेसिक UNIX Commands. | 49 |
| | 4.4 | Linux ओएस का परिचय | 51 |
| | 4.5 | बेसिक Linux Command Line. | 51 |
| | 4.6 | DOS ओएस का परिचय | 58 |
| | 4.7 | बेसिक DOS Commands. | 59-65 |
| | | | |
| | | **(5) Digital Security Matters.** | |
| 5 | 5.1 | डिजिटल सिक्योरिटी का परिचय | 66 |
| | 5.2 | वल्नेरेबिलिटी (Vulnerability) | 66 |
| | 5.3 | वायरस, वोर्म्स, ट्रोजन हॉर्स | 67 |
| | 5.4 | Threats (कॉन्फिडेंटिअलिटी, इंटीग्रिटी,अवेलेबिलिटी) | 67 |
| | 5.5 | DOS Attack (डिनायल ऑफ सर्विस) | 68 |
| | 5.6 | ब्लैक हैट एवं वाइट हैट हैकर्स | 69 |

| SR | SUB | SUBJECT LIST OR TOPICS | PAGE# |
|---|---|---|---|
| | | **(5) Digital Security Matters.** | |
| **5.** | **5.7** | BotNet (Zombie) नेटवर्क | **70** |
| | **5.8** | ऑपरेटिंग सिस्टम सिक्योरिटी | **71** |
| | **5.9** | TCB (ट्रस्टेड कंप्यूटिंग बेस) | **72** |
| | **5.10** | MINIX 3 ऑपरेटिंग सिस्टम | **73** |
| | **5.11** | ऑथेंटिकेशन एवं ऑथोराइजेशन | **73** |
| | **5.12** | सिक्योरिटी मैजरिंग लेवल्स | **74** |
| | **5.13** | बफर ओवरफ्लो अटैक्स | **74** |
| | **5.14** | लॉजिक बॉम्ब एवं बैकडोर | **76** |
| | **5.15** | लॉगिन स्पूफिंग | **76** |
| | **5.16** | मॉलवेयर एवं की–लॉगर | **77** |
| | **5.17** | ट्रोजन हॉर्सेज | **77** |
| | **5.18** | कंप्यूटर वायरस के टाइप्स | **77** |
| | **5.19** | रूटकिट एवं पेगासस | **78** |
| | **5.20** | जेलब्रेकिंग एवं हैक्टिविज्म | **78-79** |
| | **5.21** | फोन फ्रीकिंग एवं DDOs अटैक | **79** |
| | **5.22** | एनोनिमस एवं वॉयरलैस हैकिंग टूल्स | **80** |
| | **5.23** | कांसेप्ट ऑफ TPM | **80** |
| | **5.24** | डिजिटल सिग्नेचर | **81** |
| | **5.25** | फर्स्ट वायरस एवं फर्स्ट ट्रोजन हॉर्स | **81** |
| | **5.26** | भारत में साइबर अपराध | **83** |
| | **5.27** | ई–कॉमर्स से संबंधित मामलों के लिए कानून | **83** |
| | **5.28** | IT Act– 2000, 2008 66A, Section 69A | **83** |
| | **5.29** | मॉडल टेस्ट पेपर | **84-90** |
| | | | |
| | | **(6) Introduction to Firewall & its Utilities.** | |
| 6. | 6.1 | फायरवॉल, टाइप्स ऑफ फायरवॉल | 91 |
| | 6.2 | पैकेट फिल्टर फायरवॉल | 92 |
| | 6.3 | एप्लीकेशन लेवल प्रॉक्सी सर्वरर्स (गेटवे) | 93 |
| | 6.4 | स्टेटफुल पैकेट इंस्पेक्शन फायरवॉल | 94 |
| | 6.5 | OSI की सेवन लेयर्स (फायरवॉल के सन्दर्भ में) | 94 |
| | 6.6 | यूटिलिटीज ऑफ फायरवॉल | 95 |
| | 6.7 | पर्सनल फायरवॉल | 96 |
| | 6.8 | DMZ नेटवर्क्स | 96 |
| | 6.9 | वर्चुअल प्राइवेट नेटवर्क्स | 96-97 |
| | | | |
| | | **(7) IP Addresssing With Subnetting & Masking.** | |
| **7.** | **7.1** | IP एड्रेसिंग का परिचय | **98** |
| | **7.2** | IP क्लासेज (डेसीमल नोटेशन) | **98** |
| | **7.3** | IP (बाइनरी टू डेसीमल कन्वर्जन) | **99** |
| | **7.4** | IP क्लास फाइंडिंग ड्रिल्स – सबनेटिंग एवं राऊटर प्रोसेस | **100-101** |
| | **7.5** | मास्किंग (विथ सबनेटिंग एवं विथआउट सबनेटिंग) | **102** |

| SR | SUB | SUBJECT LIST OR TOPICS | PAGE# |
|---|---|---|---|
| **(8) Computer Fundamentals & Architecture.** | | | |
| **8.** | **8.1** | कंप्यूटर का परिचय | **103** |
| | **8.2** | हिस्ट्री ऑफ कंप्यूटर | **103** |
| | **8.3** | ओल्डर कम्प्यूटर्स वर्जन्स | **104** |
| | **8.4** | कंप्यूटर की विशेषताएं | **104** |
| | **8.5** | कंप्यूटर की वीकनेस | **104** |
| | **8.6** | कंप्यूटर की जनरेशन्स | **105** |
| | **8.7** | एप्लीकेशन के आधार पर कम्प्यूटर्स के टाइप्स | **105** |
| | **8.8** | पर्पस के आधार पर कम्प्यूटर्स के टाइप्स | **106** |
| | **8.9** | साइज एवं वर्क पॉवर के आधार पर कम्प्यूटर्स के टाइप्स | **106-107** |
| | **8.10** | सुपर कम्प्यूटर्स एवं भारत की भूमिका | **107** |
| | **8.11** | ब्लॉक डायग्राम ऑफ कंप्यूटर विथ डिटेल | **108-110** |
| | **8.12** | कैरेक्टरिस्टिक ऑफ कंप्यूटर | **111** |
| | **8.13** | वॉन न्यूमेंन आर्किटेक्चर मॉडल | **111** |
| | **8.14** | वॉन न्यूमेंन बोटलनेक | **113** |
| | **8.15** | ALU एंड REGISTERS के टाइप्स | **114** |
| | **8.16** | CPU एंड BUSES | **114** |
| | **8.17** | डाटा बस, एड्रेस बस, कन्ट्रोल बस | **114** |
| | **8.18** | मदर बोर्ड एवं इसके कंपोनेंट्स | **115** |
| | **8.19** | बूटस्ट्रैपिंग यूजिंग BIOS एंड UEFI | **115** |
| | **8.20** | चिपसेट का परिचय | **117** |
| | **8.21** | नार्थ एवं साउथ ब्रिज | **118** |
| | **8.22** | सिस्टम क्लॉक एवं सिस्टम बस | **120** |
| | **8.23** | इंटर्नल एवं एक्सटर्नल बस | **121** |
| | **8.24** | एड्रेस, डाटा एवं कन्ट्रोल बस | **122** |
| | **8.25** | डाटा मेजरमेंट यूनिट्स | **123** |
| **(9) Input & Output Devices.** | | | |
| **9.** | **9.1** | इनपुट एवं आउटपुट डिवाइसिस का परिचय | **125** |
| | **9.2** | सभी इनपुट डिवाइसिस चित्र सहित विस्तार से | **125-133** |
| | **9.3** | सभी आउटपुट डिवाइसिस चित्र सहित विस्तार से | **134-140** |
| **(10) Storage Devices (Memory).** | | | |
| **10.** | **10.1** | कंप्यूटर की स्टोरेज डिवाइसिस का परिचय | **141** |
| | **10.2** | टाइप्स ऑफ कंप्यूटर मैमोरी | **142** |
| | **10.3** | RAM, DRAM, एवं SRAM | **143** |
| | **10.4** | ROM एवं MROM, PROM, EPROM, EEPROM, FLASH ROM , Memory Hierarchy | **144** |
| | **10.5** | CPU में उपस्थित CPU के लिए मैमोरीज | **146** |
| | **10.6** | Latch एवं Register Memory | **146** |
| | **10.7** | Register के टाइप्स | **146** |
| | **10.8** | Cache मैमोरी विस्तार से | **147** |
| | **10.9** | Virtual मैमोरी विस्तार से, Thrashing, Page Fault. | **147** |
| | **10.10** | DDR और Cloud एवं Virtual स्टोरेज | **148** |

| | | | |
|---|---|---|---|
| | 10.11 | सेकेंडरी मैमोरी एवं टाइप्स विस्तार से | 149 |
| | 10.11 | सीरियल या सीक्वेंसियल एक्सेस एवं टाइप्स | 149 |
| | 10.12 | डायरेक्ट या रैंडम एक्सेस एवं टाइप्स | 150 |
| | 10.13 | हार्ड डिस्क फिजिकल स्ट्रक्चर विस्तार से | 150-151 |
| | 10.14 | एक्सेस, सीक एवं लेटेंसी टाइम | 151 |
| | 10.15 | सॉलिड स्टेट ड्राइव्स | 152 |
| | 10.16 | ऑप्टिकल डिस्क एवं CD ROM | 152-153 |
| | 10.17 | CD RW एवं DVD | 153 |
| | 10.18 | BluRay एवं Flash Memory, Memory Card, SD Card, Pen Drives. | 153 |
| | 10.19 | मॉडल टेस्ट पेपर | 155 |
| **(11) Networking Fundamentals.** | | | |
| **11.** | **11.1** | डेटा कम्युनिकेशन का परिचय | **162** |
| | **11.2** | ट्रांसमिशन मीडियम | **162** |
| | **11.3** | प्रोटोकॉल्स एवं नेटवर्क्स | **162-163** |
| | **11.4** | डिस्ट्रिब्यूटेड प्रोसेसिंग इन नेटवर्क्स | **163** |
| | **11.5** | नेटवर्क क्राइटेरिया | **164** |
| | **11.6** | स्टैंडर्ड्स ऑर्गेनाइजेशन (IEEE, ANSI, EIA, ISO, ITU-T) | **164** |
| | **11.7** | एलिमेंट्स ऑफ प्रोटोकॉल (सिंटेक्स ,सिमेंटिक्स ,टाइमिंग) | **165** |
| | **11.8** | लाइन कॉन्फीगुरेशन (पॉइंट टु पॉइंट एंड मल्टीपॉइंट) | **166** |
| | **11.9** | नेटवर्क टोपोलॉजीज (मैश, स्टार, ट्री, बस, रिंग) | **167** |
| | **11.10** | ट्रांसमिशन मोड्स (सिम्पलेक्स , हाफ डुप्लेक्स , फुल डुप्लेक्स) | **169** |
| | **11.11** | टाइप्स ऑफ नेटवर्क्स (LAN, MAN, WAN, PAN, CAN ,CDN), FireWire, USB, Zigbee | **170-175** |
| | **11.12** | मॉडल टेस्ट पेपर | **176** |
| **(12) OSI & TCP-IP MODELS.** | | | |
| **12.** | **12.1** | OSI मॉडल का परिचय | **181** |
| | **12.2** | OSI की लेयर्स एवं सीक्वेंस ट्रिक | **181** |
| | **12.3** | विस्तार से सभी लेयर्स की सर्विसेज | **184** |
| | **12.4** | (एप्लीकेशन, प्रेजेंटेशन,सेशन, ट्रांसपोर्ट लेयर) | **183-186** |
| | **12.5** | (नेटवर्क, डाटा लिंक, फिजिकल लेयर) | **187-188** |
| | **12.6** | TCP-IP मॉडल का परिचय | **190** |
| | **12.7** | OSI की लेयर्स एवं सीक्वेंस ट्रिक | **192** |
| | **12.8** | मॉक टेस्ट सीरीज | **193** |
| **(13) SIGNALS.** | | | |
| **13.** | **13.1** | Signals का परिचय एवं एनालॉग सिग्नल्स एवं डिजिटल सिग्नल्स | **199** |
| | **13.2** | एनालॉग सिग्नल्स का पीरियड, फ्रीक्वेंसी और एम्पलीट्यूड | **200** |
| | **13.3** | न्यूमेरिकल्स एवं बैंडविड्थ ऑफ सिग्नल्स | **201** |
| | **13.4** | एनालॉग सिग्नल्स का बिट इंटरवल एवं बिट रेट एवं न्यूमेरिकल्स | **202** |
| | **13.5** | मॉक टेस्ट सीरीज | **204** |
| **(14) Networking & InterNetworking Devices.** | | | |
| **14.** | **14.1** | नेटवर्किंग एवं इंटर नेटवर्किंग डिवाइसिस का परिचय | **209** |
| | **14.2** | इंटरनेटवर्क एवं इंटरनेट | **209** |
| | **14.3** | नेटवर्किंग डिवाइसिस – रिपीटर्स एवं ब्रिजिज एवं टाइप्स | **210-213** |

| | | | |
|---|---|---|---|
| | **14.4** | स्पानिंग ट्री अल्गोरिथम | **213** |
| | **14.5** | इंटर नेटवर्किंग डिवाइसिस – राऊटर्स एवं राउटिंग कांसेप्ट | **214** |
| | **14.6** | लीस्ट कॉस्ट राउटिंग | **215** |
| | **14.7** | हॉप काउंट राउटिंग | **215** |
| | **14.8** | नॉन एडप्टिव राउटिंग एवं एडप्टिव राउटिंग | **216** |
| | **14.9** | पैकेट लाइफ टाइम (टाइम टू लिव TTL) | **216** |
| | **14.10** | लूप्स एंड बाउंसिंग प्रॉब्लम एंड सोलूशन | **216** |
| | **14.11** | गेटवेज एंड टाइप्स बेस्ड ऑन डेटा फ्लो एंड फंक्शनैलिटी | **217** |
| | **14.12** | स्विचेज एंड वर्किंग कांसेप्ट | **219** |
| | **14.13** | स्टोर एंड फॉरवर्ड मेथड एंड कट जैतवनही स्विच | **221** |
| | **14.14** | मॉडेम, Wifi, Lifi वर्किंग | **221-226** |
| | **14.15** | मॉक टेस्ट सीरीज | **228** |
| **(15) Introduction To Internet Technology.** | | | |
| **15.** | **15.1** | इंटरनेट टेक्नोलॉजी का परिचय | **234** |
| | **15.2** | हिस्ट्री ऑफ इंटरनेट | **234** |
| | **15.3** | इंट्रानेट एवं इसके फायदे | **235** |
| | **15.4** | इंटरनेट एवं इंट्रानेट | **236** |
| | **15.5** | एक्सट्रानेट एंड इम्प्लीमेंटेशन | **236** |
| | **15.6** | इंटरनेट सर्विसेज | **237** |
| | **15.7** | TCP-IP प्रोटोकॉल सुइट | **239** |
| | **15.8** | क्लाइंट सर्वर मॉडल | **239** |
| | **15.9** | BOOTP & DHCP - Protocols | **240** |
| | **15.10** | DNS का परिचय विस्तार से | **241** |
| | **15.11** | जेनेरिक, सेकंड लेवल एंड ccTLDs डोमेन | **242** |
| | **15.12** | टेलनेट – टर्मिनल नेटवर्क | **243** |
| | **15.13** | FTP एंड SMTP वर्किंग | **243-244** |
| | **15.14** | MIME एंड POP | **244-245** |
| | **15.15** | SNMP एंड HTTP | **246-248** |
| | **15.16** | URL विस्तार से | **249** |
| | **15.17** | WWW (W3) का परिचय विस्तार से | **250** |
| | **15.18** | Hypertext and Hypermedia | **251** |
| | **15.19** | Web Browser | **252** |
| | **15.20** | ब्राउजर आर्किटेक्चर | **253** |
| | **15.21** | Cookies **&** its Types | **254** |
| | **15.22** | Search Engine **&** its Components | **255** |
| | **15.23** | Search Engine Working | **255** |
| | **15.24** | Crawling, Indexing, Serving search results | **256-258** |
| | **15.25** | Search Engine Optimization (SEO) | **259** |
| | **15.26** | Webmaster **&** Freenet, TOR **&** Crowdfunding, Deep Web **&** Data Buffer. | **260** |
| | **15.27** | Web Servers **&** Proxy Servers. | **250** |
| **(16) Introduction To Offline & Online Messaging.** | | | |
| **16.** | **16.1** | Messaging का परिचय | **261** |
| | **16.2** | ऑनलाइन मैसेजिंग – ऑफलाइन मैसेजिंग | **262** |
| | **16.3** | मैसेजिंग टूल्स | **262-265** |

| | | | |
|---|---|---|---|
| **(17) Number Systems.** | | | |
| **17.** | **17.1** | Number System का परिचय | **266** |
| | **17.2** | डेसीमल (Base 10) एवं बाइनरी (Base 2) | **266** |
| | **17.3** | ऑक्टल (Base 8) एवं हेक्साडेसीमल (Base 16) | **267** |
| | **17.4** | Decimal to Binary कन्वर्जन | **269** |
| | **17.5** | Decimal Fraction to Binary कन्वर्जन | **270** |
| | **17.6** | Decimal to Hexadecimal कन्वर्जन | **271** |
| | **17.7** | Binary To Decimal कन्वर्जन | **272** |
| | **17.8** | Binary Fraction To Decimal कन्वर्जन | **273** |
| | **17.9** | Binary To Hexadecimal कन्वर्जन | **274** |
| | **17.10** | Hexadecimal to Binary कन्वर्जन | **274** |
| | **17.11** | Hexadecimal to Decimal कन्वर्जन | **274** |
| | **17.12** | One's Complement | **275** |
| | **17.13** | From Decimal to One's Complement | **275** |
| | **17.14** | Two's Complement | **276** |
| | **17.15** | Decimal to Two's Complement | **276** |
| **(18) Web Technology. [HTML & INTERACTIVE TOOLS]** | | | |
| **18.** | **18.1** | Web Publishing का परिचय | **277** |
| | **18.2** | Types of Web Hosting . | **278** |
| | **18.3** | HTML का परिचय एवं वर्जन्स | **279** |
| | **18.4** | Structure of a HTML Web Page . | **281** |
| | **18.5** | HTML Tags. | **282** |
| | **18.6** | HTML5 Semantic Tags . | **282** |
| | **18.7** | HTML प्रोग्रामिंग विथ प्रोग्राम्स | **283** |
| | **18.8** | HTML – BASIC TAGS विथ प्रैक्टिकल्स | **289** |
| | **18.9** | HTML – ATTRIBUTES. | **292** |
| | **18.10** | HTML INTERACTIVE TOOLS. | **294** |
| | **18.11** | Tables & Nesting of Tables. | **297-305** |
| **(19) MICROSOFT OFFICE SUITE.** | | | |
| **19.** | **19.1** | MS Office Suite का परिचय | **306** |
| | **19.2** | MS WORD (Microsoft Word) कम्पलीट | **307-333** |
| | **19.3** | MICROSOF EXCEL कम्पलीट | **332-340** |
| | **19.4** | MICROSOF PowerPoint कम्पलीट | **341-354** |
| | **19.5** | MICROSOF ACCESS कम्पलीट | **355-364** |
| **(20) Programming Using C.** | | | |
| **20.** | **20.1** | C लैंग्वेज का परिचय **(C-1)** | **365** |
| | **20.2** | The Format & Features of C. | |
| | **20.3** | फ्लोचार्ट्स, अल्गोरिथ्म्स एवं PSEUDOCODE. | |
| | **20.4** | C Tokens. **(C-2)** | **374** |
| | **20.5** | Keywords. | |
| | **20.6** | Identifiers. | |
| | **20.7** | Literals/Constants. | |
| | **20.8** | Data Types. | |
| | **20.9** | Operators. | |

| | | | |
|---|---|---|---|
| | **20.10** | Separators. | |
| | **20.11** | Control Statements-Conditions का परिचय **(C-3)** | **390** |
| | **20.12** | if एंड if else | |
| | **20.13** | nested if-else एवं if-else-if | |
| | **20.14** | Switch Case. | |
| | **20.15** | Common Programming Errors. | |
| | **20.16** | Control Statements & Looping. **(C-4)** | **398** |
| | **20.17** | while loop. | |
| | **20.18** | do-while loop. | |
| | **20.19** | for loop. | |
| | **20.20** | Jump Statements [break] [continue] [goto] [return] | |
| | **20.21** | Type Casting - Implicit & Explicit Type Casting. | |
| | **20.22** | Arrays का परिचय **(C-5)** | **413** |
| | **20.23** | One Dimensional Array (1-D) | |
| | **20.24** | Features – Operations on Arrays | |
| | **20.25** | Two Dimensional Array (2-D) (Matrix) | |
| | **20.26** | Algebra of Matrices. | |
| | **20.27** | String Character (Array) का परिचय **(C-6)** | **425** |
| | **20.28** | String input (getchar( ) एवं gets( ) ) | |
| | **20.29** | String output (puts( ) ) | |
| | **20.30** | String Handling Functions. | |
| | **20.31** | Functions का परिचय **(C-7)** | **430** |
| | **20.32** | Function Definition & Return Type. | |
| | **20.33** | Argument Passing Mechanism (Call by value) & (Call by reference) | |
| | **20.34** | Recursion. | |
| | **20.35** | Pointers का परिचय **(C-8)** | **441** |
| | **20.36** | Memory and pointer. | |
| | **20.37** | Pointers and addresses. | |
| | **20.38** | Null Pointer. | |
| | **20.39** | Pointers to pointers & Pointers and Array. | |
| | **20.40** | Pointers and Functions. | |
| | **20.41** | Static Memory Allocation & Dynamic Memory Allocation. | |
| | **20.42** | Structures का परिचय **(C-9)** | **455** |
| | **20.43** | Structure using Array. | |
| | **20.44** | Structures within Structure (Nested Structure) | |
| | **20.45** | Unions. | |
| | **20.46** | Enumerated Data Type. | |
| | **20.47** | Bit Fields. | |
| | **20.48** | Typedef | |
| | **20.49** | File handling through C **(C-10)** | **464** |
| | **20.50** | File Operations | **465** |
| | **20.51** | File Related Functions | **465** |
| | **20.52** | Concept of Buffer | **465** |
| **(21) IDEs and Its Advantages.** | | | |
| **21.** | **21.1** | IDE का परिचय विस्तार से | **469** |
| | **21.2** | IDE Tools. | **469** |
| | **21.3** | History of IDE. | **469** |
| | **21.4** | Benefits of Using IDEs. | **470** |

| | | | |
|---|---|---|---|
| | **21.5** | Languages That Are Supported by IDE . | **470** |
| | **21.6** | Different Types of IDE. | **470** |
| | **21.7** | Multi-Language IDE . | **470** |
| | **21.8** | IDE for Mobile Development. | **471** |
| | **21.9** | HTML IDE. | **471** |
| | **21.10** | Cloud-Based IDE . | **471** |
| | **21.11** | IDE Specific to Apple or Microsoft. | **471** |
| | **21.12** | Application security and the IDE. | **471** |
| | **21.13** | IDE Advantages. | **472-73** |
| **(22) System Backup & Restore.** | | | |
| **22.** | **22.1** | Complete Procedure of Backup and Restore. | **474-483** |
| **(23) Single & Double Bus Structure.** | | | |
| **23.** | **23.1** | Difference between Single & Double Bus Structure. | **484-485** |
| **(24) Concepts of Object Oriented programming (OOP)** | | | |
| **24.** | **24.1** | Difference Between C & C++ | **486** |
| | **24.2** | Difference Between Procedure & Object Oriented Programming. | **488** |
| | **24.3** | Principles or Features of object oriented programming. | **488** |
| | **24.4** | Encapsulation. | **489** |
| | **24.5** | Data Abstraction. | **489** |
| | **24.6** | Polymorphism. | **490** |
| | **24.7** | Types of Polymorphism. | **490** |
| | **24.8** | Inheritance. | **494** |
| | **24.9** | Dynamic Binding. | **494** |
| | **24.10** | Message Passing. | **494** |
| | **24.11** | Benefits of object oriented programming (OOPs) | **495** |
| | **24.12** | BRIEF HISTORY OF C++ | **495** |
| | **24.13** | BRIEF HISTORY OF JAVA | **495** |
| | **24.14** | Brief History of Python | **496** |
| **(25) A.I, M.L, and Initiatives.** | | | |
| **25.** | **25.1** | Brief History of Artificial Intelligence. (A.I) | **497** |
| | **25.2** | भारत में आर्टिफिशियल इंटेलिजेंस (A.I) और मशीन लर्निंग (M.L) | **497** |
| | **25.3** | Types of A.I | **498** |
| | **25.4** | भारत में आर्टिफिशियल इंटेलिजेंस की संभावनाएँ | **499** |
| | **25.5** | सरकार द्वारा किये जा रहे प्रयास | **499** |
| | **25.6** | आर्टिफिशियल इंटेलिजेंस पर 7–सूत्री रणनीति | **499** |
| | **25.7** | सऊदी अरब का इंटेलिजेंट रोबोट सोफिया | **499-500** |
| | **25.8** | Learn with Google AI | **500** |
| | **25.9** | Initiatives by MeitY in Emerging Technologies. | **501** |
| **(26) Latest Technologies in Computer Science.** | | | |
| **26.** | **26.1** | Quantum Computing & Robotics. | **502** |
| | **26.2** | Cybersecurity & Bioinformatics. | **502** |
| | **26.3** | Data Science & Full Stack Development. | **502** |
| | **26.4** | Virtual Reality and Augmented Reality. | **503** |
| | **26.5** | Edge Computing & Artificial Intelligence. | **503** |

| | | | |
|---|---|---|---|
| **(27) Blockchain Technology & RBI.** | | | |
| **27.** | **27.1** | Introduction to Blockchain Technology. | **504** |
| | **27.2** | Introduction to Bitcoin Cryptocurrency. | **505** |
| | **27.3** | Data Hash & Cryptomining. | **505** |
| | **27.4** | Difference Between Bitcoin and Blockchain. | **506** |
| | **27.5** | RBI Digital Rupee. | **506** |
| | **27.6** | RBI LaxmiCoin. | **506** |
| | **27.8** | Collaborations With IITs. | **507-508** |
| **(28) Introduction to Big Data and Hadoop.** | | | |
| **28.** | **28.1** | Introduction to Big Data and Hadoop. | **509** |
| | **28.2** | Characteristics of Big Data. | **509** |
| | **28.3** | 5Vs of Big Data. | **510** |
| **(29) Introduction to Cloud Computing.** | | | |
| **29.** | **29.1** | Introduction to Cloud Computing. | **511** |
| | **29.2** | What is Cloud Computing. | **511** |
| | **29.3** | Who Uses Cloud Computing. | **511** |
| | **29.4** | How Does Cloud Computing Work. | **511** |
| | **29.5** | Cloud Computing Deployment Models. | **512** |
| | **29.6** | Public Cloud. | **512** |
| | **29.7** | Private Cloud. | **512** |
| | **29.8** | Hybrid Cloud. | **512** |
| **(30) Freeware, Shareware and Open Source Softwares.** | | | |
| **30.** | **30.1** | Introduction to Freeware, Shareware and Open Source Softwares. | **513-514** |


***

गुरु मार्गदर्शक, उपदेशक, दिशानिर्देशक हो सकता है लेकिन अर्जुन की तरह घोर अभ्यास करके प्रतियोगिता में गाण्डीव धनुष से अचूक निशाना लगाकर विजय तुमको हांसिल की होगी, वर्ना कहीं से भी, कुछ भी, कैसे भी पढ़ लो, गुरु भी कुछ नहीं कर सकता। मुझे आपका पढ़ाई के प्रति समर्पण चाहिए। शायद जब आप मन लगाकर दृढ़ संकल्प से इस पुस्तक को पढ़ लोगे तब आपको आभास होगा कि एक गुरु, किसको किसमें बदलने की क्षमता रखता है।

# Chapter ∞ 1 ∞

# (Introduction To Operating Systems)

## परिचय–

जिस प्रकार इंसान के शरीर को चलाने के लिए उस शरीर में आत्मा का होना आवश्यक होता है उसी प्रकार एक कंप्यूटर को चलाने के लिए उस में ऑपरेटिंग सिस्टम का होना आवश्यक होता है। ऑपरेटिंग सिस्टम एक सिस्टम सॉफ्टवेयर होता है जो विभिन्न प्रकार के कंप्यूटर प्रोग्रामिंग लैंग्वेज के प्रोग्राम्स का सेट होता है। ऑपरेटिंग सिस्टम के बिना कंप्यूटर को चलाना असंभव है ऑपरेटिंग सिस्टम के बिना कंप्यूटर मात्र एक प्लास्टिक, कांच, लोहे का ढांचा मात्र है जिसको हार्डवेयर कहते हैं।

ऑपरेटिंग सिस्टम, कंप्यूटर की इनपुट डिवाइस, आउटपुट डिवाइस, मैमोरी, सीपीयू सहित अन्य चीजों का मैनेजमेंट करता है ऑपरेटिंग सिस्टम ऐसी सुविधाएं देता है जिससे अन्य एप्लीकेशन सॉफ्टवेयर भी ऑपरेटिंग सिस्टम के अंदर आकर कंप्यूटर पर अपना कार्य कर सकें।

ऑपरेटिंग सिस्टम की कुछ सामान्य जिम्मेदारियां होती हैं जैसे कंप्यूटर में की–बोर्ड से या किसी और से दिए गए इनपुट को पहचानना, उसे सीपीयू से एक्सीक्यूट करवाना और उसे कंप्यूटर के मॉनिटर पर प्रदर्शित करना, फाइल्स का कंप्यूटर की मैमोरी में रिकॉर्ड सुरक्षित रखना, एवं उनका मैनेजमेंट करना कंप्यूटर के साथ जुड़ी हुई अन्य आवश्यक डिवाइसों जैसे डिस्क ड्राइवर, प्रिंटर्स, स्पीकर इन सभी चीजों का मैनेजमेंट करना और एक साथ चलने वाले विभिन्न प्रोग्राम्स का इंटरफेस, जैसे – कंप्यूटर यूजर एवं कंप्यूटर हार्डवेयर के बीच में इंटरफेस प्रदान करना।

ऑपरेटिंग सिस्टम का जी.यू.आई जिसको ग्राफिकल यूजर इंटरफेस कहते हैं के माध्यम से कंप्यूटर पर कार्य करने की क्षमता में सरलता प्रदान करता है। ऑपरेटिंग सिस्टम, कंप्यूटर के विभिन्न हार्डवेयर और एप्लीकेशन प्रोग्राम्स का मैनेजमेंट भी करता है जब कंप्यूटर पर एक से अधिक यूजर होते हैं तब ऑपरेटिंग सिस्टम मैमोरी मैनेजमेंट, इनपुट/आउटपुट मैनेजमेंट, विभिन्न प्रकार के प्रोसेस/रिसोर्सेज का मैनेजमेंट, सुरक्षा इत्यादि प्रदान करता है।

हम कह सकते हैं कि ऑपरेटिंग सिस्टम, *यूजर्स* और *कंप्यूटर हार्डवेयर* के बीच मध्यस्थता का काम करते हैं और अन्य एप्लीकेशन्स को चलाते हैं। आजकल पूरी दुनिया में माइक्रोसॉफ्ट कंपनी के ऑपरेटिंग सिस्टम जैसे विंडोज एक्स.पी, विंडोज–7, विंडोज–10, विंडोज–11 काफी लोकप्रिय हैं। ऐसा नहीं है कि सिर्फ माइक्रोसॉफ्ट कंपनी ही ऑपरेटिंग सिस्टम बनाती है अन्य कंपनियां भी ऑपरेटिंग सिस्टम बनाती है जैसे– एप्पल कंपनी के मैक ओ.एस, लाइनेक्स के बहुत सारे ऑपरेटिंग सिस्टम फ्लेवर्स है लेकिन माइक्रोसॉफ्ट कंपनी के ऑपरेटिंग सिस्टम ज्यादा यूजर फ्रेंडली एवं ग्राफिकल इंटरफेस वाले हैं जिन पर अनजान व्यक्ति भी काम कर सकता है।

माइक्रोसॉफ्ट कंपनी अपने ऑपरेटिंग सिस्टम को डीवीडी में डालकर उस डीवीडी को मार्केट में बेचती है या खुद के वेबसाइट सर्वर से डायरेक्ट पेड डाउनलोड करने की सुविधा भी देती है और जब हम कंप्यूटर खरीदते हैं तब जिस सर्विस सेंटर से वह खरीदा है वह सर्विस सेंटर वाला उस ऑपरेटिंग सिस्टम की डीवीडी से ऑपरेटिंग सिस्टम को हमारे नए कंप्यूटर के हार्डवेयर (हार्ड डिस्क, एसएसडी) मैं इंस्टॉल करता है यानी उस डीवीडी में से पूरा ऑपरेटिंग सिस्टम पैकेज हमारे कंप्यूटर के (हार्ड डिस्क, एसएसडी) के अंदर डाल देता है बस फिर क्या है हमारा कंप्यूटर, डीवीडी में जो निर्देश/प्रोग्राम (सिस्टम सॉफ्टवेयर – ऑपरेटिंग सिस्टम) लिखे थे उनके अनुसार कार्य करना शुरू कर देता है। तो आपको पता चल गया होगा कि ऑपरेटिंग सिस्टम कितना पावरफुल होता है जो एक डेड हार्डवेयर मशीन को जीवनदान देकर उसको कार्य करने के लायक बना देता है। इसको और अच्छे से समझने के लिए सबसे पहले हमें जानना होगा कि सॉफ्टवेयर के कितने प्रकार होते हैं और उनको बांटा क्यों गया है तो चलिए सीखते हैं।

सामान्यतः जिन लोगों को नहीं पता है कि ऑपरेटिंग सिस्टम क्या होता है वह सोचते होंगे यह कैसा होता होगा तो दोस्तों आपने डीवीडी देखी होगी उसके अंदर ऑपरेटिंग सिस्टम को डाल दिया जाता है कंप्यूटर की भाषा में सीडी या डीवीडी में किसी डाटा को डालना मतलब उसको राइट करना या बर्न करना कहते है।

**सॉफ्टवेयर के प्रकार –**

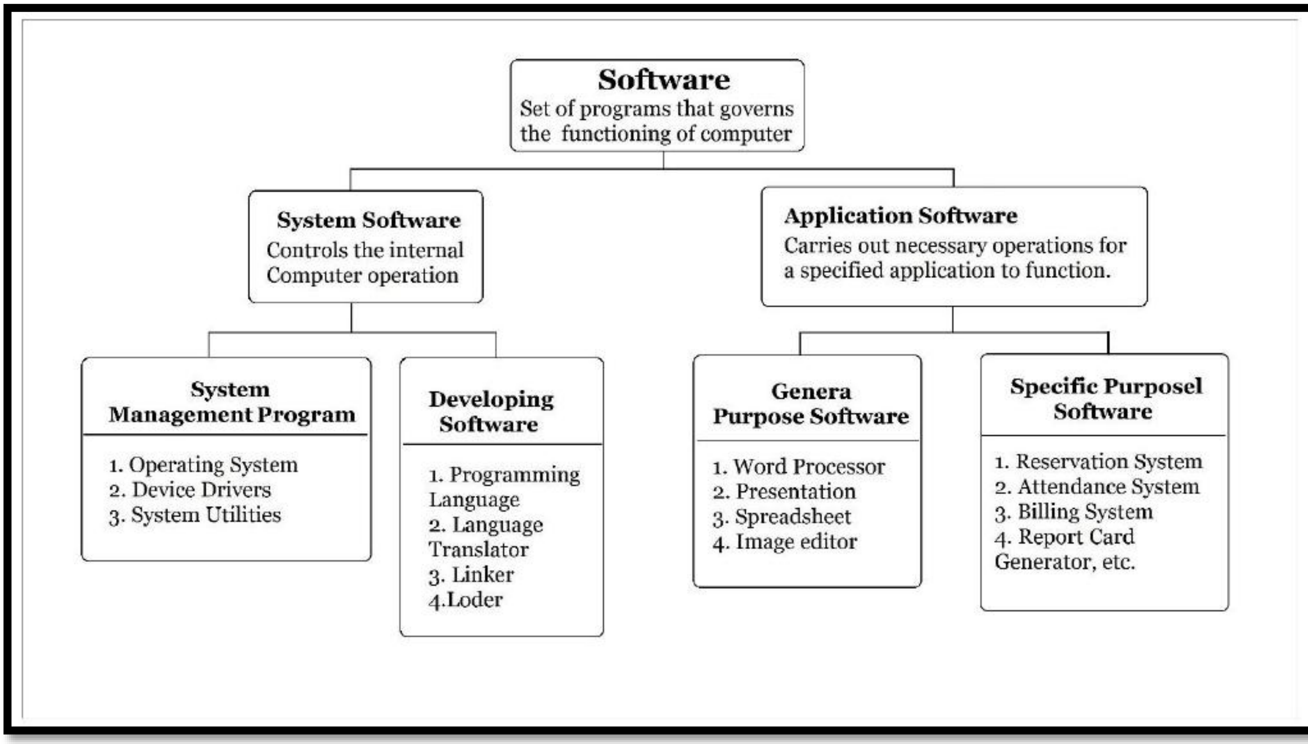



Figure 1.1 Types of Software.

**सॉफ्टवेयर दो प्रकार के होते हैं –**

- **सिस्टम सॉफ्टवेयर**
- **एप्लीकेशन सॉफ्टवेयर**

**सिस्टम सॉफ्टवेयर –**

ऐसे सॉफ्टवेयर जिनके बिना कंप्यूटर को चला पाना असंभव है या यह कहें ऐसे सॉफ्टवेयर जो कंप्यूटर को चलाने के लिए चाहिए ही चाहिए उनके बिना कंप्यूटर चला पाना नामुमकिन है ऐसे सॉफ्टवेयर सिस्टम सॉफ्टवेयर कहलाते हैं। जैसा कि नाम से पता चल रहा है सिस्टम सॉफ्टवेयर, तो आप याद रख सकते हैं ऐसे सॉफ्टवेयर जो सिस्टम को चलाने के लिए चाहिए ही चाहिए, सिस्टम सॉफ्टवेयर कहलाते हैं। जैसे मेरे कंप्यूटर को चलाने के लिए मुझे ऑपरेटिंग सिस्टम चाहिए ही चाहिए, ऑपरेटिंग सिस्टम के बिना तो कंप्यूटर चलेगा नहीं, ऐसे में ऑपरेटिंग सिस्टम को हम सिस्टम सॉफ्टवेयर कहेंगे।

सिस्टम सॉफ्टवेयर के उदाहरण – समस्त ऑपरेटिंग सिस्टम जैसे विंडोज ओएस, लिनक्स ओएस, यूनिक्स ओएस, मैक ओएस, क्रोम ओएस बेस्ड लिनक्स बाई गूगल, फ्री बीएसडी ओएस बेस्ड यूनिक्स, सिलेबल ओएस बेस्ड यूनिक्स, रियेक्ट ओएस बेस्ड विंडोज, संट ओएस बेस्ड लिनक्स, डेबियन ओएस बेस्ड लिनक्स, डिपिन ओएस बेस्ड लिनक्स, सोलिरिस ओएस बेस्ड यूनिक्स बाई सन माइक्रोसिस्टम्स (ओरेकल) इत्यादि।

**एप्लीकेशन सॉफ्टवेयर –**

ऐसे सॉफ्टवेयर जो कंप्यूटर को चलाने के लिए जरूरी नहीं है लेकिन सामान्य इस्तेमाल करने वाले कुछ कार्यों में इनकी जरूरत पड़ती है। जैसे मुझे मेरे कंप्यूटर पर इंटरनेट भी चलाना है तो उसके लिए मेरे पास क्रोम ब्राउजर होना चाहिए जिसमें वेबसाइट खुलती हैं तो उस क्रोम ब्राउजर को हम एप्लीकेशन सॉफ्टवेयर कह सकते हैं जो अलग से कंप्यूटर में इंस्टॉल करके डाले जाते हैं। सिस्टम सॉफ्टवेयर के कुछ उदाहरण निम्न हैं। जैसे – माइक्रोसॉफ्ट वर्ड प्रोसेसर, टेली एकाउंटिंग, वेब ब्राउजर, मीडिया प्लेयर, कंसोल गेम, फोटो एडिटर इत्यादि।

**कर्नेल –**

कर्नेल, ऑपरेटिंग सिस्टम पैकेज का सबसे मुख्य प्रोग्राम होता है ऐसा कह सकते हैं यह पूरे ऑपरेटिंग सिस्टम का बॉस होता है इसका पूरे ऑपरेटिंग सिस्टम, उसकी हर चीज, हर कार्य प्रणाली, पर कंट्रोल होता है कर्नेल, यूजर से डायरेक्टली इंटरैक्ट नहीं करता यह कंप्यूटर ऑपरेटिंग सिस्टम के अन्य प्रोग्राम, हार्डवेयर, डिवाइस, प्रोसेसर, मैमोरी, डिस्क ड्राइव, शैल इत्यादि के साथ समन्वय एवं संचार स्थापित करता है। कर्नेल, ऑपरेटिंग सिस्टम की हर घटना, कार्यप्रणाली पर निगरानी रखता है और उनको कंट्रोल करता है इसको इस चित्र के माध्यम से समझ सकते हैं।

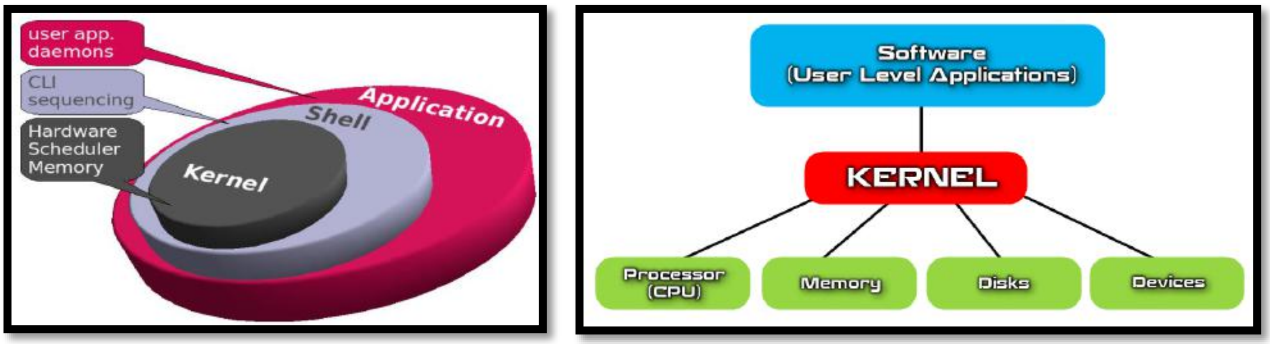



**Figure 1.2 Understanding Kernels.**

कर्नेल ही ऑपरेटिंग सिस्टम का एक ऐसा प्रोग्राम है जो सबसे पहले बूटिंग के टाइम कंप्यूटर की मैमोरी में लोड होता है और जब तक कंप्यूटर चालू रहता है तब तक यह प्रोग्राम वहीं पर रहता है और वहां से निगरानी, नियंत्रण, प्रबंधन सभी कार्यों को बॉस की तरह बैठकर बखूबी निभाता है ऐसा भी कह सकते हो यह ऑपरेटिंग सिस्टम का सेंट्रल कोर है। कर्नेल कंप्यूटर को चालू करवाने के साथ–साथ कुछ महत्वपूर्ण कार्य जैसे मैमोरी, प्रोसेस, फाइल, इनपुट/आउटपुट, कम्युनिकेशन इन सारे कार्यों को बखूबी निभाता है इसमें हमको नया शब्द पढ़ने को मिला जिसको हमने शैल कहा चलिए देखते हैं शैल क्या होता है।

**Heavy Drill –**

सामान्य सिस्टम स्टार्टअप के दौरान, कंप्यूटर का बेसिक इनपुट/आउटपुट सिस्टम, या BIOS, हार्डवेयर बूटस्ट्रैप या इनिशियलाइजेशन को पूरा करता है। फिर ये एक बूटलोडर रन करता है जो स्टोरेज डिवाइस जैसे हार्ड डिस्क या एसएसडी जहां पर ऑपरेटिंग सिस्टम का पैकेज इन्सटाल्ड होता है से कर्नेल को रेम मैमोरी के एक प्रोटेक्टेड स्पेस में लोड करता है। जैसे ही कर्नेल प्रोग्राम कंप्यूटर रेम मैमोरी में लोड हो जाता है उसके बाद, BIOS अपने नियंत्रण को कर्नेल को ट्रांसफर कर देता है। इसके बाद यह सिस्टम स्टार्टअप को पूरा करने और डेस्कटॉप या अन्य यूजर इंटरफेस के माध्यम से यूजर्स को नियंत्रण उपलब्ध कराने के लिए अन्य ओएस कंपोनेंट्स को लोड करता है।

**Heavy Drill –** Winload.exe (Windows Boot Loader) is a small piece of software, called a system loader, that's started by BOOTMGR, the boot manager used in Windows 11, Windows 10, Windows 8, Windows 7, and Windows Vista operating systems.

**शैल –**

जिस प्रकार ऑपरेटिंग सिस्टम में कर्नेल एक प्रोग्राम था उसी प्रकार ऑपरेटिंग सिस्टम में एक और प्रोग्राम है जिसका नाम शैल है। शैल, यूजर को इंटरफेस प्रदान करता है इसका मुख्य कार्य कमांड्स को पढ़ना और उनको कर्नेल के पास ले जाकर समझाना की यूजर ने यह कमांड दी है वह यह कार्य करवाना चाहता है फिर कर्नेल उस शैल से उस कमांड को लेकर अन्य हार्डवेयर एवं प्रोग्राम के जरिए उस कमांड को एग्जीक्यूट कर पूरा करवाता है। हम कंप्यूटर को दो प्रकार से निर्देश या कमांड देते हैं।

- **GUI** (ग्राफिकल यूजर इंटरफेस) के जरिये – जहां पर यूजर कमांड लिखकर निर्देश नहीं देता है बल्कि एक ग्राफिकल इंटरफेस वातावरण के जरिए माउस के माध्यम से विंडो मीनू के जरिए बटन दबाकर कमांड देता है।
- **CUI** के जरिये – कमांड प्रॉम्प्ट सीएमडी, सीएलआई/सीयूआई– कमांड लाइन इंटरफेस/कमांड लाइन इनपुट/करैक्टर यूजर इंटरफेस के जरिये

## ऑपरेटिंग सिस्टम के फंक्शंस एवं सर्विसेस–

**प्रोग्राम्स को एग्जीक्यूट करवाना –**

ऑपरेटिंग सिस्टम का बहुत ही महत्वपूर्ण कार्य है जब भी कोई उपयोगकर्ता (यूजर) कंप्यूटर को किसी कार्य को पूरा करने के लिए कमांड देता है तो ऑपरेटिंग सिस्टम की जिम्मेदारी है कि वह उस कमांड को या उस कार्य को पूरा करवाए। इसके लिए ऑपरेटिंग सिस्टम यूजर की कमांड प्रोग्राम को मेन मैमोरी (रैम) में लोड करवा कर सीपीयू के पास भेजता है एवं सीपीयू से उसको एग्जीक्यूट (पूरा) करवा कर वापस आउटपुट को, आउटपुट डिवाइस के माध्यम से कंप्यूटर के उपयोगकर्ता (यूजर) के सामने प्रदर्शित करता है अगर प्रोग्राम में या कमांड में कोई कमी है तो ऑपरेटिंग सिस्टम की जिम्मेदारी है की एरर मैसेज दिखाकर यूजर को सूचित करें।

**इनपुट एवं आउटपुट ऑपरेशंस –**

जब–जब कंप्यूटर यूजर इनपुट डिवाइस की सहायता से कंप्यूटर में इनपुट देता है तब तब ऑपरेटिंग सिस्टम इनपुट डिवाइस से उस डाटा को लेकर मेन मैमोरी (रैम) में लोड करवा कर सीपीयू से ऑपरेशन को (कमांड को) एग्जीक्यूट करवाकर आउटपुट डिवाइस पर यूजर के लिए प्रदर्शित करता है।

**संवाद (कम्युनिकेशन) –**

कंप्यूटर सिस्टम में बहुत सारी प्रोसेस एक साथ चलती रहती हैं जब एक प्रोसेस को दूसरी प्रोसेस से सूचना का आदान प्रदान करने की आवश्यकता महसूस होती है तो वह आपस में मैमोरी को शेयर करके प्रोसेस को कम्युनिकेट करती हैं यानी प्रोसेस एक दूसरे के साथ कम्युनिकेशन करती हैं इसका मतलब यह हुआ कि एक प्रोसेस दूसरी प्रोसेस के साथ सूचनाओं का आदान प्रदान करती है इसको इंटरप्रोसेस कम्युनिकेशन कहते हैं।

**लेखा–जोखा (एकाउंटिंग) –**

जब एक से अधिक यूजर कंप्यूटर सिस्टम पर काम कर रहे होते हैं तो ऑपरेटिंग सिस्टम उनकी हर गतिविधि पर नजर रखता है और साथ ही उन सब चीजों को नोट करता है जो उन उपयोगकर्ताओं के द्वारा कंप्यूटर के साथ की गई है। इसमें यह शामिल होता है कि कौनसा यूजर कौन से रिसोर्स को कितने समय तक यूज कर रहा है कंप्यूटर ऑपरेटिंग सिस्टम इन सब चीजों की ट्रैकिंग करता है और इनको जिस स्थान पर सुरक्षित रखता है उसको यूजर सेशन लॉग कहते हैं।

**सुरक्षा (प्रोटेक्शन) –**

ऑपरेटिंग सिस्टम, कंप्यूटर सिस्टम पर कार्य करने वाले सभी उपयोगकर्ताओं की सुरक्षा की जिम्मेदारी को बखूबी निभाता है हर यूजर के लिए यूजर नेम और पासवर्ड जैसी सुविधाएं देता है। जिससे निश्चित प्रकार का यूजर निश्चित प्रकार के अकाउंट में जाकर कंप्यूटर में मांगी गई सूचनाओं को वेरीफाई करके सफलतापूर्वक अंदर जाने की अनुमति देता है वहां पर जाकर उपयोगकर्ता अपना अपना कार्य करते हैं और कार्य को सुरक्षित करते हैं।

ऑपरेटिंग सिस्टम यह भी ध्यान रखता है कि एक यूजर दूसरे यूजर की फाइल और डाटा के साथ छेड़छाड़ नहीं कर पाए यानी हर यूजर का डाटा प्राइवेट और सुरक्षित रहे हालांकि एक दूसरे के साथ अपनी फाइल एवं डाटा को विशेष विकल्प के द्वारा शेयर भी कर सकता है कुल मिलाकर ऑपरेटिंग सिस्टम का यह अति महत्वपूर्ण कार्य है कि वह किसी उपयोगकर्ता की निजी जानकारी एवं डाटा फाइल सभी को एकदम सुरक्षित रखें। सुरक्षा की इस बात को हम दूसरे शब्दों में यह भी कह सकते हैं कि जब बहुत सारे प्रोसेस एक साथ कंप्यूटर में एग्जीक्यूट हो रहे होते हैं तो कोई भी प्रोसेस दूसरी प्रोसेस के साथ इंटरफेरेंस नहीं करें यह सिस्टम के रिसोर्सेज को कंट्रोल करता है तथा प्रोसेस एवं डाटा को उपयोगकर्ता के लिए एकदम सुरक्षित रखता है।

**फाइलों का प्रबंधन (फाइल मैनेजमेंट) –**

कंप्यूटर सिस्टम में यूजर का डाटा, फाइल इत्यादि हमेशा सेकेंडरी मैमोरी (हार्ड डिस्क) में ही परमानेंट सेव किए जाते हैं समय–समय पर यूजर अपने डाटा फाइल्स के साथ विभिन्न प्रकार के ऑपरेशन गतिविधियां करता रहता है जैसे वह कभी नई फाइल बनाकर सेव करता है कभी पुरानी फाइल में एडिट करके नया डाटा जोड़कर फिर से सुरक्षित करता है कभी वह अनावश्यक डाटा को सेकेंडरी मैमोरी से डिलीट भी करता है डुबलीकेट डाटा को कॉपी बनाने से पहले आपको मैसेज द्वारा बताता है और समय–समय पर अनावश्यक फाइलों को डिलीट भी करता है ऑपरेटिंग सिस्टम फाइलों के साथ इस प्रकार की विभिन्न ऑपरेशंस को अच्छे से प्रबंधित करके रखता है कई बार यूजर फाइल्स में सुरक्षा हेतु पासवर्ड, प्रोटेक्टेड मोड, इस तरीके के कई प्रकार के ऑपरेशन करता है तो इस प्रकार की गतिविधियां बहुत ही अच्छे तरीके से ऑपरेटिंग सिस्टम हैंडल करता है।

**गलतियों को हैंडल करना (एरर हैंडलिंग) –**

अगर कंप्यूटर के किसी भी रिसोर्स या डिवाइस (हार्डवेयर) में ऑपरेशंस के दौरान कोई एरर मैसेज आता है तो ऑपरेटिंग सिस्टम उसको बखूबी हैंडल करता है अगर वह स्वयं ठीक करने लायक है तो ऑपरेटिंग सिस्टम उसको स्वयं हैंडल करता है अगर वह थोड़ा सा कॉम्प्लिकेटेड है तो वह उस एरर मैसेज को कंप्यूटर यूजर के सामने डिस्प्ले करता है और कंप्यूटर यूजर अपनी चॉइस के अनुसार उसको फिक्स करता है किसी डिवाइस का इनवेलिड इनप्रॉपर यूज होने पर ऑपरेटिंग सिस्टम एरर मैसेज देता है।

जैसे– लॉगइन के वक्त गलत पासवर्ड डालने पर ऑपरेटिंग सिस्टम एरर मैसेज देता है और किसी आवश्यक एक्टिव प्रोसेस को टास्क मैनेजर से किल करने पर भी ऑपरेटिंग सिस्टम एरर मैसेज देता है कुछ एरर हार्डवेयर एरर होते हैं जैसे मैमोरी एलोकेशन एरर, अर्थमैटिक ऑपरेशंस, उदाहरण – डिवाइडेड बाय जीरो इत्यादि।

## प्रमुख धारणाएँ –

**जॉब पूल –**

जब भी कोई प्रोग्राम एग्जीक्यूट होने के लिए जाता है (प्रोग्राम अंडर एक्सिक्यूशन या एक्सिक्यूशन की अवस्था में) तो उसको ऑपरेटिंग सिस्टम की भाषा में प्रोसेस या जॉब कहते हैं सभी प्रोसेस या जॉब सिस्टम की मैमोरी में एकत्रित रहती हैं जिसको जॉब पूल कहते हैं जो जो प्रोसेस एग्जीक्यूट होना चाहती हैं वह जॉब पूल में एकत्रित हो जाती हैं और मेन मैमोरी रैम में लोड होने के लिए कतार में लगकर अपनी अपनी बारी का इंतजार करती हैं जब उनकी बारी आती है तो वह मेन मैमोरी रैम में लोड हो जाती हैं और वहां से सीपीयू में जाकर एग्जीक्यूट होती हैं।

**प्रोसेस –**

जब भी कोई प्रोग्राम, एग्जीक्यूट होने की अवस्था में होता है तो उसको प्रोसेस कहते हैं। यानी कोई भी प्रोग्राम जब चल रहा होता है तो वह एक प्रोसेस कहलाता है और वह किसी विशेष कार्य को संपन्न करता है। जब भी कोई प्रोसेस क्रिएट होती है तो उसको कंप्यूटर सिस्टम के विभिन्न रिसोर्सेज जैसे– मैमोरी, रजिस्टर्स, फाइल्स, सीपीयू, इत्यादि की आवश्यकता होती है। प्रोसेस की कई स्टेट होती हैं जैसे– न्यू, रेडी, रनिंग, वेटिंग, टर्मिनेटेड।

- **न्यू स्टेट–** इस स्टेट में नई प्रोसेस बनती है।
- **रेडी स्टेट–** कोई भी प्रोसेस शुरू होने के पश्चात तुरंत रेडी स्टेट में आ जाती है इस स्टेट में प्रोसेस कंप्यूटर की मेन मैमोरी (रैम) में लोड होकर प्रोसेसर के एलोकेट होने की प्रतीक्षा करती है।
- **रनिंग स्टेट–** जब किसी प्रोसेस के इंस्ट्रक्शंस एग्जीक्यूट हो रहे होते हैं तो प्रोसेस रनिंग स्टेट में कहलाती है जब कोई प्रोसेस सीपीयू तथा अन्य सिस्टम रिसोर्सेज के कंट्रोल में आती है तो एग्जीक्यूट करना प्रारंभ कर देती है रनिंग प्रोसेस किसी दूसरे प्रोसेस को एकजुट करने के लिए कंट्रोल स्थानांतरित कर सकती है रनिंग प्रोसेस द्वारा, इनपुट आउटपुट ऑपरेशंस के दौरान कंट्रोल को किसी दूसरी प्रोसेस या ऑपरेटिंग सिस्टम को स्थानांतरित किया जाना शेड्यलिंग नीति पर निर्भर करता है।
- **वेटिंग स्टेट–** इस स्टेट में कोई भी प्रोसेस किसी इवेंट के घटित होने की प्रतीक्षा करती रहती है जैसे इनपुट आउटपुट ऑपरेशंस के खत्म होने की प्रतीक्षा करना या किसी इससे पहले प्रोसेस के खत्म होने के बाद उसकी बारी आने की प्रतीक्षा करना।
- **टर्मिनेटेड स्टेट–** जब कोई प्रोसेस अपने अंतिम स्टेटमेंट को एग्जीक्यूट कर लेती है तो वह ऑपरेटिंग सिस्टम द्वारा टर्मिनेट कर दी जाती है यानी जब उसका कार्य पूरा हो गया जिसको वह करने आई थी। उसके बाद उसको टर्मिनेट कर के जॉब पूल से हटा दिया जाता है प्रोसेस का यह स्टेट टर्मिनेट स्टेट कहलाता है टर्मिनेट स्टेट में कोई प्रोसेस अपनी पैरंट प्रोसेस को डाटा भी रिटर्न कर सकती है।

**डेडलॉक–**

जब दो या दो से अधिक प्रोसेस एक साथ चल रही होती हैं तब कभी–कभी अचानक से ऐसी स्थिति में आकर फँस जाती है जहां पर उन सभी प्रोसेस को एक ही समय पर एक ही रिसोर्स चाहिए तो वह आपस में रिसोर्स ब्लॉकेज की स्थिति उत्पन्न करती हैं यानी हर प्रोसेस उस रिसोर्स को पाने के लिए झगडती हैं यानी इस प्रोसेस को मैं पहले लूंगा, दूसरी प्रोसेस कहती है मैं पहले लूंगा, तीसरी प्रोसेस कहती है मैं पहले लूंगा, और वह आपस में एक रिसोर्स ब्लॉकेज की स्थिति उत्पन्न कर देती हैं जिसको डेडलॉक कहते हैं। इस स्थिति में रिसोर्स कुछ भी हो सकता है जैसे कि प्रिंटर।

# Types of Operating Systems

### मल्टीप्रोग्रामिंग ऑपरेटिंग सिस्टम –

मल्टीप्रोग्रामिंग ऑपरेटिंग सिस्टम में एक से अधिक प्रोग्राम (जॉब्स) को मैमोरी में एक साथ लोड करके उन्हें एक साथ प्रोसेस किया जाता है मल्टीप्रोग्रामिंग ऑपरेटिंग सिस्टम में मल्टीप्रोग्रामिंग नामक तकनीकी का प्रयोग एक साथ एक से अधिक जॉब्स को मैमोरी में लोड करने एवं उन्हें एक साथ प्रोसेस करने के लिए किया जाता है मल्टीप्रोग्रामिंग तकनीकी निम्न तरीके से कार्य करती है।

- मल्टीप्रोग्रामिंग ऑपरेटिंग सिस्टम एक से अधिक जॉब्स को मैमोरी में एक साथ लोड करता है।
- मल्टीप्रोग्रामिंग ऑपरेटिंग सिस्टम इनमें से किसी एक जॉब को एकजुट करना शुरू करता है।
- जब कोई एक जॉब एग्जिक्यूट कर रहा होता है तो ऑपरेटिंग सिस्टम बाकी की उन सभी जॉब्स का एक कतार लाइन लगाकर रखवाता है जो सीपीयू की उपलब्धता के लिए प्रतीक्षा कर रहे होते हैं।
- जब वर्तमान में एकजुट हो रहे जॉब्स में इनपुट आउटपुट ऑपरेशंस की आवश्यकता होती है तो ऑपरेटिंग सिस्टम अगले जॉब को प्रोसेसिंग के लिए सीपीयू को पर स्थानांतरित कर देता है।
- जब पहले वाले जॉब का इनपुट आउटपुट ऑपरेशन समाप्त होता है तो ऑपरेटिंग सिस्टम फिर से लाइन में लगा देता है ताकि सीपीयू उपलब्ध हो तो इसको पूरा किया जा सके।
- इस प्रकार ऑपरेटिंग सिस्टम एक जॉब से दूसरे जॉब स्थानांतरित करता रहता है सीपीयू कभी भी आइडल स्थिति में नहीं रहता है।

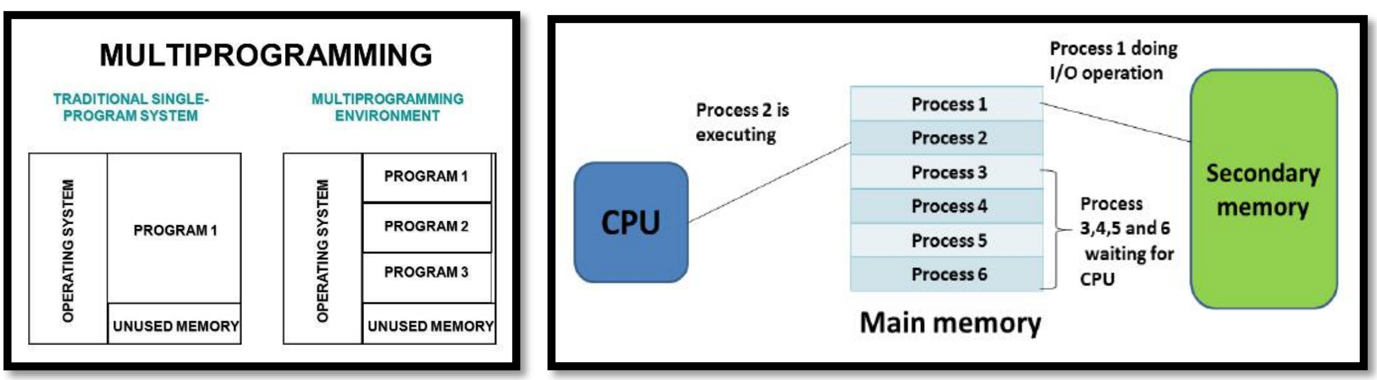



**Figure 1.3 Multi-programming.**

### टाइम शेयरिंग ऑपरेटिंग सिस्टम –

टाइम शेयरिंग ऑपरेटिंग सिस्टम, मल्टीप्रोग्रामिंग ऑपरेटिंग सिस्टम का ही एक रूप है टाइम शेयरिंग सिस्टम समय के कुछ रेस्पॉन्स के साथ कार्य करता है टाइम शेयरिंग सिस्टम मल्टी यूजर्स के लिए एक क्रम में सीपीयू का समान समय एलोकेट करता है एवं ऑपरेटिंग सिस्टम द्वारा प्रत्येक यूजर को सीपीयू द्वारा दिए जाने वाला एलोकेट टाइम, '**टाइम स्लाइस**' कहलाता है यह टाइम 5 से 100 मिली सेकंड तक होता है जैसे ही टाइम स्लाइस की अवधि समाप्त होती है ऑपरेटिंग सिस्टम द्वारा सीपीयू के लिए अगले यूजर को एलोकेट कर दिया जाता है।

टाइम शेयरिंग ऑपरेटिंग सिस्टम के बारे में, कोई प्रोसेस केवल दिए गए समय के लिए एग्जीक्यूट होती है यदि प्रोसेस ऑपरेटिंग सिस्टम द्वारा दिए गए समय से अधिक समय लेता है तो वह प्रोसेस ऑपरेटिंग सिस्टम द्वारा इंटरप्ट कर क्यू (लाइन) के अंत में वापस उसका नंबर आने तक प्रतीक्षा में लगा दी जाती है जहां वह प्रोसेस अपने अगले आने वाले टाइम स्लाइस लोकेशन के लिए प्रतीक्षा करता रहता है टाइम शेयरिंग ऑपरेटिंग सिस्टम का सबसे अच्छा उदाहरण **राउंड रोबिन शेड्यूलिंग एल्गोरिथ्म** से समझ सकते हैं।

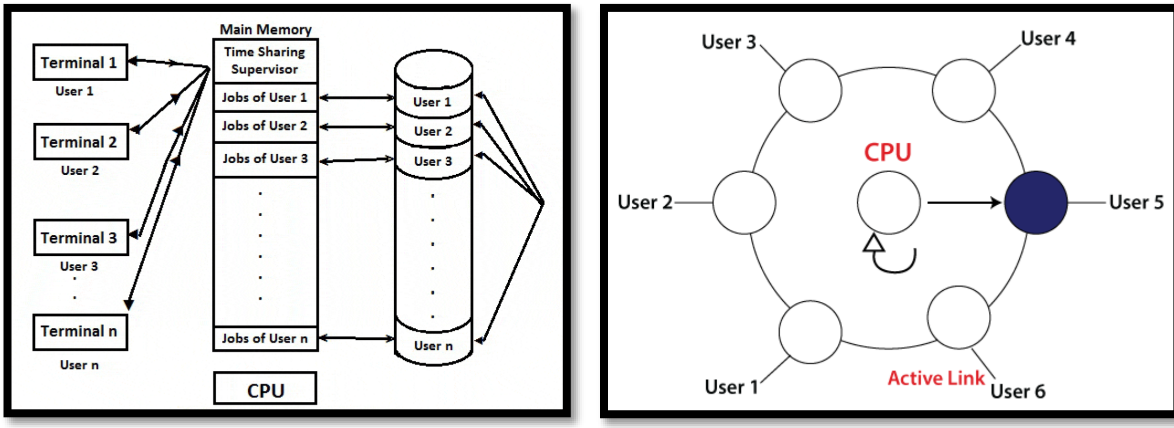



**Figure 1.4 Time Sharing Systems.**

**बैच प्रोसेसिंग ऑपरेटिंग सिस्टम –**

इस प्रकार के ऑपरेटिंग सिस्टम में जितनी भी प्रोसेस (जॉब्स) हैं उनका एक ग्रुप बनाया जाता है जिसको बैच कहते हैं और बैच के रूप में जॉब्स के ग्रुप को एग्जीक्यूट करने के लिए भेजा जाता है ऑपरेटिंग सिस्टम, शेड्यूलिंग की सहायता से प्रायोरिटी (प्राथमिकता) एवं रिसोर्स के आधार पर जॉब्स को एग्जीक्यूट करवाने के लिए उत्तरदाई होता है।

उदाहरण के लिए बैंक की नेफ्ट सेवा को समझ सकते हैं जिसमें पूरे देश में पूरे दिन में जितने भी नेफ्ट ट्रांजेक्शंस होते हैं उनको बैच के माध्यम से हर आधे घंटे के अंतराल पर बैंक सर्वर के द्वारा बैच बनाकर एग्जीक्यूट किया जाता है। इस प्रकार के बैच ऑपरेटिंग सिस्टम में कंप्यूटर यूजर की आवश्यकता ना के बराबर होती है क्योंकि जो जो जॉब्स एग्जीक्यूट करानी है वह जॉब कतार (पूल) में पहले इकट्ठा होती है फिर एक साथ उनको प्रोसेस कराया जाता है।

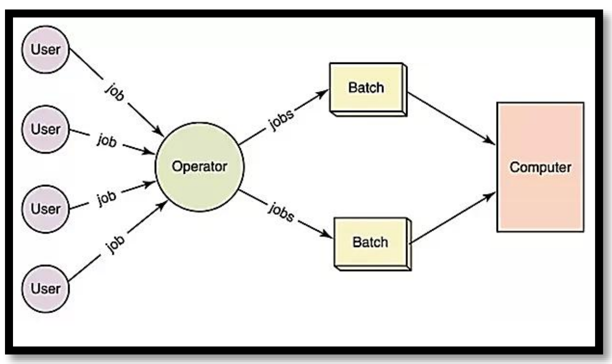



**Figure 1.5 Batch Processing Systems.**

**रियल टाइम ऑपरेटिंग सिस्टम –**

ऐसा ऑपरेटिंग सिस्टम होता है जो कि दिए गए समय में कार्य को बहुत तेज तुरंत करता है यह रियल टाइम एप्लीकेशंस को सपोर्ट करता है। उसका मुख्य का उपयोग इंडस्ट्रियल, साइंटिफिक, कभी–कभी यह सैटेलाइट, बैंकों में भी प्रयोग किया जाता है रियल टाइम ऑपरेटिंग सिस्टम मुख्यतः वहीं पर उपयोग किया जाता है जहां पर समय की आवश्यकता बहुत स्ट्रिक्ट होती है जैसे– मिसाइल सिस्टम, एयर ट्रेफिक कंट्रोल सिस्टम, रोबोट, बैंकिंग चैनल, इत्यादि। यह दो प्रकार के होते हैं।

- **हार्ड रियल टाइम सिस्टम**
- **सॉफ्ट रियल टाइम सिस्टम**

**हार्ड रियल टाइम ऑपरेटिंग सिस्टम** – अगर कोई एक्शन/कार्यवाही बिल्कुल एक ठीक निश्चित समय पर करवानी हो तो हार्ड रियल टाइम ओएस सिस्टम प्रयोग करते हैं जैसे कार इंजन कंट्रोल सिस्टम, हर्ट पेसमेकर, मिसाइल सिस्टम, हार्ड रियल टाइम सिस्टम का उदाहरण हैं।

**सॉफ्ट रियल टाइम सिस्टम** – इस प्रकार के सिस्टम में कोई टास्क एक निश्चित समय पर नहीं होकर थोड़ा इधर–उधर हो जाती है तो उसको एक्सेप्ट कर लिया जाता है।

### पैरलर प्रोसेसिंग –

पैरलर प्रोसेसिंग में कई प्रोसेसर एक साथ मिलकर कार्य करते हैं यहां पर किसी भी कार्य को पहले टुकड़ों (सेग्मेंट्स) में विभाजित किया जाता है और फिर विभिन्न प्रोसेसर के पास वो टुकड़े (सेग्मेंट्स) एक्सीक्यूट होने के लिए चले जाते हैं और सभी प्रोसेसर पैरलर (समानांतर) मिलकर एक साथ एक समय पर उस काम को पूरा कर देते हैं पैरलर प्रोसेसिंग को पैरलर कंप्यूटिंग भी कहते हैं। पैरलर प्रोसेसिंग बहुत ज्यादा फास्ट प्रोसेसिंग करते हैं क्योंकि उनमें एक से अधिक सीपीयू कोर होती हैं जो एक साथ मिलकर कार्य करती हैं ज्यादातर कंप्यूटर्स में केवल एक सीपीयू होता है लेकिन आपको पता होना चाहिए कई कंप्यूटर में हजारों सीपीयू भी होते हैं।

जैसे– सुपर कंप्यूटर पैरलर प्रोसेसिंग के द्वारा ही कार्य करते हैं टेक्नोलॉजी पर जब वक्त बहुत कम है और कार्य बहुत ज्यादा होता है और वह दिए गए कम समय में पूरा करना होता है तब पैरलर प्रोसेसिंग टेक्नोलॉजी का उपयोग किया जाता है पैरलर प्रोसेसिंग ज्यादातर ऐसी प्रोजेक्ट्स पर यूज किया जाता है जहां पर बहुत कॉम्प्लेक्स प्रॉब्लम्स हो जैसे– वेदर की मॉडलिंग करना, बहुत कॉन्प्लेक्स और बड़ी गणनाए करने के लिए, इसका अच्छा उदाहरण देखें तो किसी दुकान पर खरीददार 1000 हैं और वह लाइन में लगे हुए हैं और दुकानदार काउंटर पर एक व्यक्ति है लेकिन अगर काउंटर पर दुकान के 10 व्यक्ति बैठे हो तो 1000 व्यक्तियों की लाइन जल्दी खत्म हो जाएगी यही पैरलर में प्रोसेसिंग होता है।

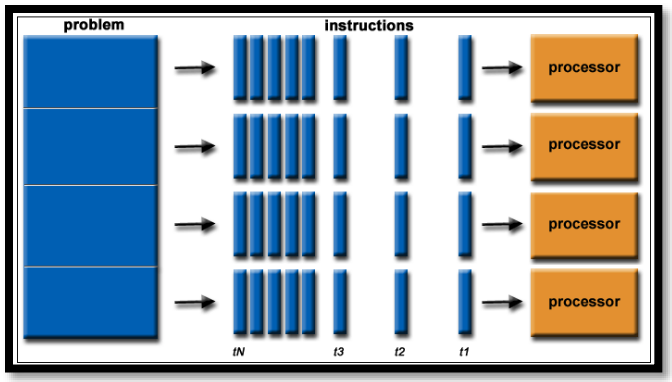



**Figure 1.6 Parallel Processing Systems.**

### मल्टी प्रोसेसर ऑपरेटिंग सिस्टम –

जैसा कि आपको पता है कुछ सालो पहले तक ज्यादातर सिस्टम सिंगल प्रोसेसर सिस्टम होते थे और उनमें केवल एक मुख्य सीपीयू होता है लेकिन मल्टी प्रोसेसर सिस्टम्स में एक से अधिक प्रोसेसर होते हैं जिनको पैरलर सिस्टम्स भी कहते हैं यह प्रोसेसर आपस में कम्युनिकेशन, शेयरिंग, कंप्यूटर बस, सिस्टम की क्लॉक, एवं मैमोरी रिसोर्सेज, डिवाइसेज शेयर करते हैं। मल्टी प्रोसेसर सिस्टम्स में एक से अधिक प्रोसेसर होने के कारण कम समय में ज्यादा से ज्यादा काम पूरा करवा सकते हैं मल्टी प्रोसेसर सिस्टम अन्य सिस्टम की अपेक्षा (मल्टीपल सिंगर प्रोसेसर) की अपेक्षा ज्यादा मैमोरी बचाते हैं क्योंकि वह कंप्यूटर की डिवाइसेज, स्टोरेज, पावर सप्लाई शेयर करते हैं अगर सिस्टम में कोई एक प्रोसेसर खराब हो जाता है या फेल हो जाता है तो सिस्टम के द्वारा काम चलता रहता है काम रुकता नहीं है बस थोड़ा सा स्पीड कम हो जाता है।

### एस.एम.पी (सिमिट्रिक मल्टिप्रोसेसिंग) –

इस प्रकार की मल्टी प्रोसेसर सिस्टम्स में हर प्रोसेसर को स्पेसिफिक टास्क असाइन कर दी जाती है हर प्रोसेसर ऑपरेटिंग सिस्टम की आईडेंटिकल कॉपी को रन करता है और यह कॉपी जरूरत पड़ने पर आपस में एक दूसरे से कम्युनिकेट भी करती हैं मल्टीप्रोसेसर सिस्टम्स में कोई भी प्रोसेसर मास्टर प्रोसेसर नहीं होता और ना ही कोई भी स्लेव (गुलाम) प्रोसेसर होता जबकि ऐसे एसएमपी सिमिट्रिक मल्टिप्रोसेसिंग में एक मास्टर प्रोसेसर होता है और वह अन्य प्रोसेसर को टास्क के लिए इंस्ट्रक्शंस देता है।

### डिस्ट्रीब्यूटिंग ऑपरेटिंग सिस्टम –

डिस्ट्रीब्यूटिंग ऑपरेटिंग सिस्टम पूरी तरह से नेटवर्किंग पर आधारित होते हैं इसमें यूजर ठीक उसी प्रकार रिमोट रिसोर्सेज का उपयोग करते हैं जिस प्रकार लोकल रिसोर्सेज का। लेकिन एक नेटवर्क में इंटरनेट को एक से अधिक कंप्यूटर आपस में कम्युनिकेशन करते हैं इसलिए डिस्ट्रीब्यूटिंग सिस्टम उसके फंक्शंस की वजह से नेटवर्किंग पर आधारित है डिस्ट्रीब्युटेड सिस्टम्स को समझने के लिए मान लीजिए एक वेबसाइट से दूसरे वेबसाइट पर डाटा और प्रोसेस के माइग्रेशन को डिस्ट्रीब्यूटर ऑपरेटिंग सिस्टम ही नियंत्रित करता है ना कि कंप्यूटर यूजर। डिस्ट्रीब्यूटिंग ऑपरेटिंग सिस्टम्स में डाटा को विभिन्न लोकेशन से स्टोर किया जा सकता है और एक्सेस किया जा सकता है और उसको प्रोसेस किया जा सकता है।

# ऑपरेटिंग सिस्टम मॉडल टेस्ट पेपर

**1. ऑपरेटिंग सिस्टम किस प्रकार का सॉफ्टवेयर है –**

(A) सिस्टम सॉफ्टवेयर (B) एप्लीकेशन सॉफ्टवेयर
(C) यूटिलिटी सॉफ्टवेयर (D) फर्मवेयर सॉफ्टवेयर

**2. ऑपरेटिंग सिस्टम क्या है –**

(A) सिस्टम सॉफ्टवेयर (B) हार्डवेयर यूटिलिटी
(C) फ्रीवेयर सॉफ्टवेयर (D) इनमें से कोई नहीं

**3. ऑपरेटिंग सिस्टम होता है –**

(A) प्रोग्राम लैंग्वेज का सेट (B) हार्डवेयर का सेट
(C) दोनों (D) इनमें से कोई नहीं

**4. ऑपरेटिंग सिस्टम का मुख्य कार्य है –**

(A) सॉफ्टवेयर का प्रबंधन (B) हार्डवेयर का प्रबंधन
(C) यूजर इंटरफेस प्रदान करना (D) उपयुक्त सभी

**5. जी.यू.आई का पूरा नाम है –**

(A) ग्राफिकल यूजर इंटरफेस (B) ग्राफिकल यूटिलिटी इंटरफेस
(C) ग्राफिकल यूजर इंटरेक्शन (D) गार्बेज यूजर इंटरफेस

**6. विंडोज है –**

(A) एक एप्लीकेशन सॉफ्टवेयर (B) एक सिस्टम सॉफ्टवेयर
(C) एक फर्मवेयर सॉफ्टवेयर (D) एक यूटिलिटी सॉफ्टवेयर

**7. विंडोस है –**

(A) सन कंपनी का प्रोडक्ट (B) माइक्रोसॉफ्ट प्रोडक्ट
(C) एप्पल कंपनी का प्रोडक्ट (D) यूनिक्स सब प्रोडक्ट

**8. निम्न में से अत्यधिक यूजर फ्रेंडली ऑपरेटिंग सिस्टम है –**

(A) यूनिक्स ओएस (B) लाइनेक्स ओएस
(C) मैकिनटोश सिस्टम (D) माइक्रोसॉफ्ट विंडोज

**9. दुनिया में सबसे लोकप्रिय, उपयोग किया जाने वाला ओएस है**

(A) यूनिक्स सिस्टम (B) लाइनेक्स सिस्टम
(C) मैकिनटोश सिस्टम (D) माइक्रोसॉफ्ट विंडोज

**10. सिस्टम सॉफ्टवेयर कितने प्रकार के होते हैं –**

(A) एक (B) दो
(C) तीन (D) चार

**11. डिवाइस ड्राइवर किस प्रकार के सॉफ्टवेयर होते हैं –**

(A) सिस्टम सॉफ्टवेयर (B) एप्लीकेशन सॉफ्टवेयर
(C) यूटिलिटी सॉफ्टवेयर (D) फर्मवेयर

**12. ट्रांसलेटर किस प्रकार के सॉफ्टवेयर होते हैं –**

(A) सिस्टम सॉफ्टवेयर (B) एप्लीकेशन सॉफ्टवेयर
(C) यूटिलिटी सॉफ्टवेयर (D) फर्मवेयर

**13. वह कौनसा प्रोग्राम है जो ऑपरेटिंग सिस्टम की कोर में उपस्थित ऑपरेटिंग सिस्टम को कंट्रोल करता है –**

(A) शैल (B) कर्नेल
(C) बॉयोस (D) फर्मवेयर

**14. हम अपने कंप्यूटर को निर्देश किस जरिए से दे सकते हैं –**

(A) कमांड प्रॉम्प्ट के जरिए (B) शैल के माध्यम से
(C) GUI के जरिए (D) उपयुक्त सभी के जरिए

**15. वाक्य को समझिए– ऑपरेटिंग सिस्टम का मैनेजर कर्नेल नाम का प्रोग्राम होता है वह पूरे ऑपरेटिंग सिस्टम पर नियंत्रण एवं प्रबंधन संपादित करता है लेकिन जब कोई यूजर कंप्यूटर पर इंटरेक्ट करता है कंप्यूटर के साथ तब–**

(A) कर्नेल डायरेक्ट यूजर से इंटरेक्ट करता है
(B) कर्नेल शैल एवं कमांड के जरिए इंटरेक्ट करता है
(C) कर्नेल जी.यू.आई के जरिए इंटरेक्ट करता है
(D) उपयुक्त सभी

**16. वह कौन सा प्रोग्राम है जो कंप्यूटर ऑन होने के बाद ऑपरेटिंग सिस्टम में से आकर सबसे पहले मैमोरी में लोड होता है –**

(A) बॉयोस (B) कर्नेल
(C) शैल (D) जी.यू.आई

**17. ऑपरेटिंग सिस्टम में सबसे महत्वपूर्ण प्रोग्राम है–**

(A) कर्नेल (B) जी.यू.आई
(C) शैल (D) बॉयोस

**18. ऑपरेटिंग सिस्टम की सेंट्रल कोर है –**

(A) कर्नेल (B) जी.यू.आई
(C) शैल (D) बॉयोस

**19. ऑपरेटिंग सिस्टम के कार्य निम्न में से कौन कौन से हैं –**

(A) प्रोग्राम्स को एग्जीक्यूट करवाना
(B) इनपुट आउटपुट ऑपरेशन करवाना
(C) सुरक्षा संबंधित प्रोसेस हैंडल करना
(D) फाइल मैनेजमेंट
(E) उपयुक्त सभी

**20. जब कोई प्रोग्राम एग्जीक्यूट होने के लिए जाता है तो उसे कहते हैं –**

(A) प्रोसेस (B) टास्क
(C) प्रोग्राम (D) इनमें से कोई नहीं

**21. सभी प्रोसेस एवं जॉब उपस्थित होती हैं –**

(A) हार्ड डिस्क में (B) रैम में
(C) बॉयोस में (D) प्रोसेसर में

Ⓐ रोम में Ⓑ रैम में
Ⓒ हार्ड डिस्क में Ⓓ प्रोसेसर में

**61.** **आजकल कंप्यूटर को तेज गति से बूट कराने के लिए बॉयोस के अंदर उपस्थित इंस्ट्रक्शंस प्रोग्राम को रैम में कॉपी कर दिया जाता है उक्त घटना कहलाती है –**

Ⓐ शैडोइंग Ⓑ शेड्यूलिंग
Ⓒ बूट लोडर Ⓓ बूट स्टार्टर

**62.** **आजकल कंप्यूटर्स में फ्लैश बॉयोस होती है जिसको यूजर –**

Ⓐ केवल एक बार अपडेट कर सकता है
Ⓑ अपडेट नहीं कर सकता
Ⓒ समय–समय पर अपडेट कर सकता है
Ⓓ इनमें से कोई नहीं

**63.** **कंप्यूटर के शुरू होने पर माइक्रोप्रोसेसर सबसे पहले किन इंस्ट्रक्शन को एग्जीक्यूट करता है –**

Ⓐ बॉयोस इंस्ट्रक्शन को
Ⓑ हार्ड डिस्क इंस्ट्रक्शन को
Ⓒ ऑपरेटिंग सिस्टम की इंस्ट्रक्शन को
Ⓓ खुद की इंस्ट्रक्शंस को

**64.** **ऑपरेटिंग सिस्टम उपस्थित होता है –**

Ⓐ हार्ड डिस्क में Ⓑ रैम में
Ⓒ रोम में Ⓓ बॉयोस में

**65.** **हार्ड डिस्क में उपस्थित ऑपरेटिंग सिस्टम को सबसे पहले किस प्रोग्राम के जरिए लोड करवाया जाता है –**

Ⓐ बूटस्ट्रैप एवं बूट लोडर Ⓑ कर्नेल
Ⓒ शैल Ⓓ शैडोइंग

**66.** **कंप्यूटर के ऑन होने के तुरंत पश्चात बॉयोस के द्वारा सबसे पहली कौन सी प्रक्रिया शुरू होती है –**

Ⓐ पोस्ट Ⓑ बैकअप
Ⓒ ऑपरेटिंग सिस्टम लोडर Ⓓ कर्नेल लोडर

**67.** **बॉयोस के फंक्शन पोस्ट का पूरा नाम क्या है –**

Ⓐ पावर ऑन सेल्फ टेस्ट Ⓑ पोस्ट ऑन सेल्फ टेस्ट
Ⓒ प्रोग्राम ऑन सेल्फ टेस्ट Ⓓ प्राइमरी ऑन सेल्फ टेस्ट

**68.** **बॉयोस के पोस्ट फंक्शन का मुख्य कार्य है –**

Ⓐ समस्त हार्डवेयर डिवाइस को चेक करना
Ⓑ समस्त सॉफ्टवेयर को चेक करना
Ⓒ ऑपरेटिंग सिस्टम को चेक करना
Ⓓ उपयुक्त सभी

**69.** **बैल लैबोरेट्री स्थित है –**

Ⓐ न्यूजर्सी Ⓑ न्यूयॉर्क
Ⓒ वाशिंगटन Ⓓ सिलीकान वैली

**70.** **कभी–कभी कंप्यूटर को शुरू करने पर बीप की आवाज आती है बीप कोड जनरेट किए जाते हैं –**

Ⓐ बॉयोस के द्वारा Ⓑ माइक्रो प्रोसेसर के द्वारा
Ⓒ ऑपरेटिंग सिस्टम के द्वारा Ⓓ हार्ड इसके द्वारा

**71.** **कौन सी प्रोसेस ऑपरेटिंग सिस्टम को लोड करवाने में सहायता करती है –**

Ⓐ बूटस्ट्रैप Ⓑ बूट सीक्वेंस
Ⓒ टास्क मैनेजर Ⓓ कर्नेल

**72.** **कंप्यूटर के विभिन्न हार्डवेयर एवं डिवाइस के खुद के सॉफ्टवेयर होते हैं जो उस हार्डवेयर को संचालित करते हैं उक्त सॉफ्टवेयर कहलाते हैं –**

Ⓐ डिवाइस ड्राइवर Ⓑ डिवाइस मैनेजर
Ⓒ डिवाइस लोडर Ⓓ डिवाइस लिंकर

**73.** **आईबीएम का पूरा नाम क्या है –**

Ⓐ इंटरनेशनल बिजनेस मशीन
Ⓑ इंटरनेशनल बायनरी मशीन
Ⓒ इंटरनेशनल बायनरी मैकेनिक्स
Ⓓ इंटरनेशनल ब्रॉडकास्ट मशीन

**74.** **आईबीएम का हेड क्वार्टर स्थित है –**

Ⓐ न्यूयॉर्क Ⓑ वाशिंगटन
Ⓒ फ्रांस Ⓓ सिलीकान वैली

**75.** **माइक्रोसॉफ्ट का हेड क्वार्टर स्थित है –**

Ⓐ न्यूयॉर्क Ⓑ वाशिंगटन
Ⓒ फ्रांस Ⓓ सिलीकान वैली

**76.** **कंप्यूटर के हार्डवेयर में गड़बड़ी पाए जाने पर बॉयोस यूजर को किस माध्यम से अलर्ट करती है –**

Ⓐ कर्नेल कोड के माध्यम से Ⓑ डिब्बागिंग के माध्यम से
Ⓒ बीप कोड के माध्यम से Ⓓ इनमें से कोई नहीं

**77.** **सन माइक्रोसिस्टम स्थित है –**

Ⓐ कैलिफोर्निया Ⓑ न्यूयॉर्क
Ⓒ वाशिंगटन Ⓓ सिलीकान वैली

**78.** **सन माइक्रोसिस्टम को किस कंपनी ने खरीद कर अपना अंग बना लिया है–**

Ⓐ ओरेकल Ⓑ आईबीएम
Ⓒ माइक्रोसॉफ्ट Ⓓ टैबलेट पैकर्ड हेड

**79.** **एप्पल कंपनी का हेड क्वार्टर स्थित है –**

Ⓐ कैलिफोर्निया Ⓑ न्यूयॉर्क
Ⓒ वाशिंगटन Ⓓ सिलीकान वैली

**80. फेसबुक कंपनी का हेड क्वार्टर स्थित है –**

Ⓐ कैलिफोर्निया Ⓑ न्यूयॉर्क
Ⓒ वाशिंगटन Ⓓ सिलीकान वैली

**81. गूगल कंपनी का हेड क्वार्टर स्थित है –**

Ⓐ कैलिफोर्निया Ⓑ न्यूयॉर्क
Ⓒ वाशिंगटन Ⓓ सिलीकान वैली

**82. गूगल कंपनी के संस्थापक कौन हैं –**

Ⓐ लैरी पेज Ⓑ सर्गी ब्रिन
Ⓒ बिल गेट्स Ⓓ अ एवं ब दोनों

**83. फेसबुक कंपनी के संस्थापक कौन हैं –**

Ⓐ मार्क जुकेरबर्ग Ⓑ लैरी पेज
Ⓒ सर्गी ब्रिन Ⓓ बिल गेट्स

**84.एप्पल कंपनी के संस्थापक कौन हैं –**

Ⓐ स्टीव जॉब्स Ⓑ मार्क जुकेरबर्ग
Ⓒ लैरी पेज Ⓓ बिल गेट्स

**85. माइक्रोसॉफ्ट कंपनी के संस्थापक कौन है –**

Ⓐ स्टीव जॉब्स Ⓑ मार्क जुकेरबर्ग
Ⓒ लैरी पेज Ⓓ बिल गेट्स

**86. इंटरनेट के संस्थापक कौन है –**

Ⓐ टिम बर्नर्स ली Ⓑ स्टीव जॉब्स
Ⓒ मार्क जुकेरबर्ग Ⓓ लैरी पेज सर्गी ब्रिन

**87. अमेजॉन कंपनी के संस्थापक कौन है –**

Ⓐ जेफ्फ बेजोस Ⓑ टिम बर्नर्स ली
Ⓒ स्टीव जॉब्स Ⓓ मार्क जुकेरबर्ग

**88. जब कोई कंप्यूटर प्रोसेस दूसरी प्रोसेस के साथ सूचनाओं का आदान प्रदान करती है इसको कहते हैं–**

Ⓐ इंटरप्रोसेस कम्युनिकेशन Ⓑ मल्टीप्रोसेस
Ⓒ सिस्टम कम्युनिकेशन Ⓓ इंटर्नल कम्युनिकेशन

**89. मल्टी यूजर्स की हर गतिविधि को ऑपरेटिंग सिस्टम ट्रैक करता है उसको कहते है –**

Ⓐ यूजर हिस्ट्री Ⓑ ऑपरेटिंग सिस्टम हिस्ट्री
Ⓒ यूजर सेशन लॉग Ⓓ यूजर हिस्ट्री लॉग

**90. पैरलर प्रोसेसिंग को कहते हैं –**

Ⓐ पैरलर कंप्यूटिंग Ⓑ क्लाउड कंप्यूटिंग
Ⓒ मल्टी कंप्यूटिंग Ⓓ इंटर्नल कंप्यूटिंग

**91.एन.ओ.एस ऑपरेटिंग सिस्टम का पूरा नाम है –**

Ⓐ नैनो ऑपरेटिंग सिस्टम Ⓑ नेटवर्किंग ऑपरेटिंग सिस्टम
Ⓒ नेटवर्किंग ओरेकल सिस्टम Ⓓ नेटवर्किंग ऑपरेटिंग सर्विस

**92.एटीएम का पूरा नाम है –**

Ⓐ ऑटोमैटिक टेलर मशीन Ⓑ ऑटोमैटिक टाइम मशीन
Ⓒ ऑटोमैटिक ट्रांसफर मशीन Ⓓ ऑटोमैटिक ट्रांसमिट मशीन

**93. सी.सी.टी.वी का पूरा नाम है –**

Ⓐ क्लोज्ड सर्किट टेलीविजन Ⓑ क्लाउड सर्किट टेलीविजन
Ⓒ सर्कुलर सर्किट टेलीविजन Ⓓ कॉमन सर्किट टेलीविजन

**94. फायर अलार्म में किस प्रकार का ऑपरेटिंग सिस्टम होता है**

Ⓐ एंबेडेड ओएस Ⓑ नेटवर्क ओएस
Ⓒ मल्टीप्रोसेसर ओएस Ⓓ पैरलर प्रोसेसिंग ओएस

**95.एटीएम मशीन में किस प्रकार का ऑपरेटिंग सिस्टम होता है**

Ⓐ एंबेडेड ओएस Ⓑ नेटवर्क ओएस
Ⓒ मल्टीप्रोसेसर ओएस Ⓓ पैरलर प्रोसेसिंग ओएस

**96. हार्डवेयर से इंटरैक्ट करता है –**

Ⓐ कर्नेल Ⓑ शैल
Ⓒ यूजर Ⓓ प्रोसेसर

**97. विंडोज 95 ऑपरेटिंग सिस्टम निम्न में से किस कंपनी के द्वारा डेवलप किया गया था –**

Ⓐ सन सिस्टम्स Ⓑ माइक्रोसॉफ्ट
Ⓒ एप्पल Ⓓ आईबीएम

**98. विंडोज 95 ऑपरेटिंग सिस्टम में INT FILE है –**

Ⓐ प्रोग्राम फाइल Ⓑ मैसेज फाइल
Ⓒ टेक्स्ट फाइल Ⓓ ऑडियो फाइल

**99. निम्न में से कौनसा एक सिस्टम सॉफ्टवेयर नहीं है –**

Ⓐ ट्रांसलेटर्स Ⓑ यूटिलिटीज
Ⓒ डेटाबेस Ⓓ ऑपरेटिंग सिस्टम

**100. Unix ओएस पर आधारित Mac OS X जिसे OS X के नाम से भी जाना जाता है जो किस कर्नेल पर आधारित है –**

Ⓐ मोनोलिथिक कर्नेल Ⓑ माइक्रो कर्नेल
Ⓒ एक्सओ कर्नेल Ⓓ हाइब्रिड कर्नेल

# Chapter ∞ 5 ∞
# (Digital Security Matters)

**परिचय–**

आज हर कहीं डिजिटल ट्रेंड चल रहा है और आज के जमाने में कंप्यूटर पर हर जगह हर टाइप का काम किया जाता है लेकिन कंप्यूटर के हार्डवेयर और यूजर के बीच इंटरेक्शन बिना ऑपरेटिंग सिस्टम (सिस्टम सॉफ्टवेयर) के संभव नहीं है ऐसा अब तक हम जान चुके है। आजकल प्रत्येक कंपनी या यूनिवर्सिटी या स्कूल या ऑफिस कुछ भी हो उनकी अपनी कुछ वैल्युएबल फाइल्स या इंफॉर्मेशंस होती हैं। जिनको वो बहोत ज्यादा सुरक्षित रखने की कोशिस करती हैं।

जैसे कोई हार्डवेयर कंपनी इंटेल या सैमसंग अपनी न्यू चिप की डिजाईन (जैसे माइक्रोप्रोसेसर या रैम) की इंफॉर्मेशंस को अपने सिस्टम्स में सुरक्षित रखती है। वैसे ही कोई सॉफ्टवेयर कंपनी अपने नए सॉफ्टवेयर के डेवलपमेंट से सम्बंधित इंफॉर्मेशंस को सुरक्षित रखती हैं ये इंफॉर्मेशंस डिजिटल इंफॉर्मेशंस हो सकती है। दूसरी तरफ फाइनेंसियल इंस्टीटूट्स या लीगल फर्म्स इनकी खुद की इंफॉर्मेशंस भी कम्प्यूटर्स में सुरक्षित रहती हैं। या किसी भी सामान्य व्यक्ति का पर्सनल कंप्यूटर हो उसमे उसकी इम्पोर्टेन्ट वैल्युएबल इंफॉर्मेशंस (जैसे – बैंक डिटेल्स, टैक्स रिटर्न, डाक्यूमेंट्स, क्रेडिट कार्ड और सोशल सिक्योरिटी नंबर्स इत्यादि) को सुरक्षित रखता है। किसी भी कंप्यूटर की सारी वैल्युएबल इंफॉर्मेशंस हार्डडिस्क में स्टोर रहती हैं। जैसे–जैसे कंप्यूटर सिस्टम्स में वैल्युएबल इंफॉर्मेशंस ज्यादा स्टोर होने लग जाती है वैसे–वैसे कंप्यूटर सिस्टम की सिक्योरिटी इंक्रीज करना एक अहम मुद्दा बन जाता है।

ऑपरेटिंग सिस्टम के सिक्योरिटी मोडूयल का मैन पर्पज (मेजर कंसर्न) वैल्युएबल इंफॉर्मेशंस को अन–ऑथोरायज्ड एक्सेस से बचाकर रखना हैं। लेकिन आज के जमाने में यही सबसे डिफिकल्ट है। क्युकी कैसा भी सिस्टम हो कहीं ना कहीं उसमे बग्स रहने ही संभावना रहती है जो एक सिक्योरिटी ब्रीच–होल साबित हो सकता है।

हजारों सालों पहले सम्राट अशोक जैसे महान सम्राट जब कोई सीक्रेट मैसेज अपनी सेना या सहयोगियों को भेजते थे तो एक खतरा ये था कि कहीं रास्ते में दुश्मन उस सीक्रेट मैसेज को इंटरसेप्ट ना कर लें। दुश्मनो से अपने सीक्रेट मैसेज को इंटरसेप्ट होने से बचाने के लिए ये लोग एन्क्रिप्शन्स का यूज करते थे। जैसे सीक्रेट मैसेज के प्रत्येक लेटर (अक्षर) को तीन लेटर (अक्षर) लेफ्ट से रिप्लेस करना या राइट साइड में रिप्लेस करना। जैसा कि आजकल हम रीजनिंग में कोड़िंग–डिकोडिंग में करते हैं। ऐसे में जिसको डिकोड करना बताया गया है वो ही डिकोड कर पाता था ऐसा ही कुछ एन्क्रिप्शन मेथड आज के जमाने में होता है। जिसके पास एन्क्रिप्शन ज्ञमल होगी वो ही मैसेज को रीड कर पायेगा।

लेकिन तब क्या होगा जब किसी आउटगोइंग मैसेज को एन्क्रिप्शन से पहले और इनकमिंग मैसेज को एन्क्रिप्शन के बाद में इंटरसेप्ट कर लिया जाए। किसी के कंप्यूटर की सिक्योरिटी को ब्रेक करना हमेशा आसान काम नहीं होता लेकिन एक 2048 बिट (size of an SSL certificate) की एन्क्रिप्शन ***Key*** को ब्रेक करने की अपेक्षा कंप्यूटर की सिक्योरिटी को ब्रेक करना आसान होगा। ऐसा कंप्यूटर के ऑपरेटिंग सिस्टम में बग्स होने के कारण हो सकता है।

**Vulnerability वल्नरेबिलिटी –**

जब कोई बग, सिक्योरिटी बग हो तो उसको वल्नरेबिलिटी कह देते हैं। और ये अक्सर सॉफ्टवेयर में पाई जाती है। हैकर्स उस बग में सही इनपुट ट्रिगर कर सॉफ्टवेयर में सेंध मार देते हैं। जिसको सिक्योरिटी की भाषा में **एक्सप्लॉइट** कहते हैं। कई बार सही एक्सप्लॉइट्स की वजह से अटैकर्स पूरे कंप्यूटर पर अपना फूल कन्ट्रोल कर लेते हैं।

जैसे अभी कुछ सालों पहले रैंसमवेयर नाम के मिलिसियस सॉफ्टवेयर (मॉलवेयर) ने दुनिया के 150 देशों के लगभग दो लाख तीस हजार कम्प्यूटर्स पर अपना कब्जा कर रिश्वत की पेशकश की थी। अटैकर्स एक्सप्लॉइट्स को मैनुअली या ऑटोमेटिकली दोनों तरीकों से लांच कर सकते हैं। इसको समझने के लिए पहले हमको वायरस और वॉर्म में फर्क समझना पड़ेगा।

**कंप्यूटर वायरस –**

कुछ लोगों का मानना हैं कि 'वायरस को फैलने के लिए कुछ ना कुछ यूजर इंटरेक्शन चाहिए' होता है जैसे यूजर ने किसी वेबसाइट लिंक या अटैचमेंट पर क्लिक किया तो उसका सिस्टम इन्फेक्ट हो सकता है।

**कंप्यूटर वोर्म्स –**

वोर्म्स खुद अपना वंश बढ़ाते रहते है यानि वो सेल्फ प्रोपेल्लेड होते हैं इनको फैलने के लिए किसी भी प्रकार का कोई भी यूजर इंटरेक्शन नहीं चाहिए होता है।

**ट्रोजन हॉर्स –**

फ्री या फोकट शब्द तो आपने सुना ही होगा ? जब भी हम इंसानो को कुछ पता चलता है कि कुछ फ्री मिल रहा है चाहे वो जहर ही क्यों ना हो ऐसे में इंसान अपनी रेस्पेक्ट उस फ्री चीज के प्रति जरूर शो करता है। और इंटरनेट की बात करें तो कोई भी सॉफ्टवेयर हो और वो हमको चाहिए उसके लिए हम उसको फ्री डाउनलोड XYZ सॉफ्टवेयर के साथ ही गूगल पर फाइंड करते है।

पिक्चर में दूसरी तरफ कुछ अटैकर्स (ताऊ लोग) क्या करते है वो पॉपुलर और महंगे सॉफ्टवेयर के पैकेज में वोर्म्स डालकर उसको इन्फेक्टेड कर रीपैकेज कर, फ्री में इंटरनेट पर कई वेबसायट्स पर डाल देते हैं। और जब हम जैसे सामान्य यूजर्स को जो महंगे सॉफ्टवेयर चाहिए वो फ्री में मिल जाते हैं तो उनकी खुशी से आँखे भर आती हैं और बिना देरी किये वो अपने कंप्यूटर में उस इन्फेक्टेड सॉफ्टवेयर को इनस्टॉल कर लेते हैं।

ऐसे में जब हम जैसे सामान्य यूजर्स उस फ्री सॉफ्टवेयर को इनस्टॉल करते हैं तो ऐसे में फ्री सॉफ्टवेयर के साथ–साथ कुछ इन्फेक्टेड एडिशनल फंक्शनेलिटी भी इनस्टॉल हो जाती है। जिससे कंप्यूटर का कन्ट्रोल अटैकर्स (ताऊ लोगों) के पास चला जाता है ऐसे सॉफ्टवेयर **ट्रोजन हॉर्स** (ट्रोजन गांव के घोड) कहलाते हैं।

**ऑपरेटिंग सिस्टम सिक्योरिटी और नेटवर्किंग सिक्योरिटी –**

ऑपरेटिंग सिस्टम सिक्योरिटी और नेटवर्किंग सिक्योरिटी दोनों एक सिक्के के दो पहलू के समान हैं। जैसे वायरस इंटरनेट के माध्यम से आते हैं लेकिन अफेक्ट ऑपरेटिंग सिस्टम को करते हैं।

**सिक्योरिटी एनवायरनमेंट–**

कुछ लोग सिक्योरिटी शब्द को और कुछ लोग प्रोटेक्शन शब्द को यूज करते रहते हैं। हम सिक्योरिटी शब्द का यूज करेंगे।

**Threats थ्रेट्स –**

कई सिक्योरिटी जर्नल्स किसी इनफार्मेशन सिस्टम की सिक्योरिटी को तीन कंपोनेंट्स में बांटते हैं। जिनको सिक्योरिटी साइंस में सीआईए (CIA) कह देते हैं। इन तीनो की कोर सिक्योरिटी प्रॉपर्टीज होती हैं जिनको हमें अटैकर्स (ताऊ लोगों) से बचाकर रखना पड़ता है।

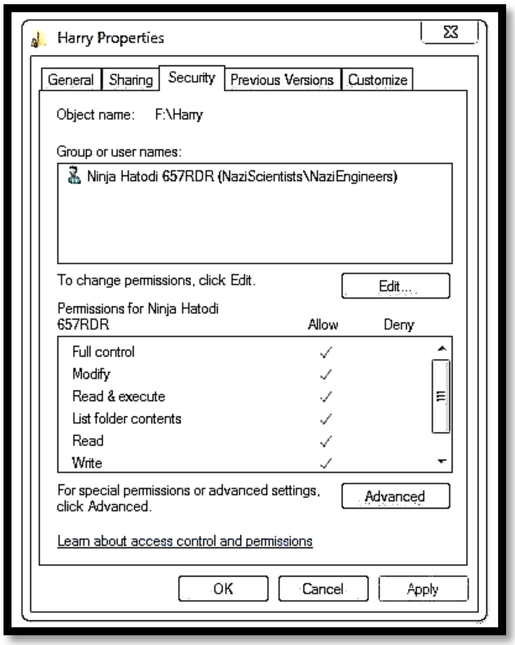
- **कॉन्फिडेंटिअलिटी – गोपनीयता**
- **इंटीग्रिटी – अखंडता**
- **अवेलेबिलिटी – उपलब्धता**

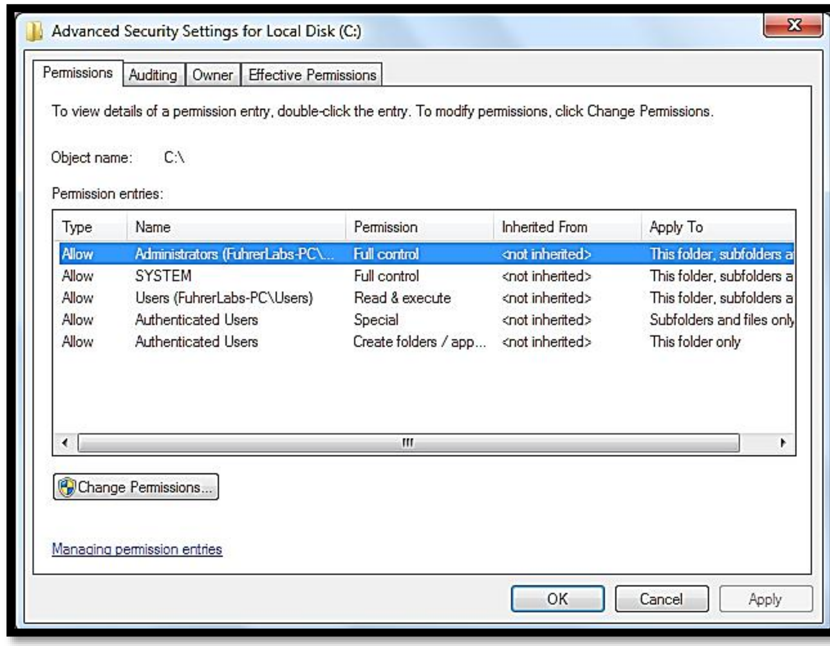
| Goal | Threat |
|---|---|
| Confidentiality | Exposure of data |
| Integrity | Tampering with data |
| Availability | Denial of service |

Security goals and threats.

**1. Confidentiality - कॉन्फिडेंटिअलिटी – गोपनीयता –**

ये सीक्रेट डेटा से सम्बंधित होती है यानी सीक्रेट डेटा सीक्रेट ही रहे बोले तो सीक्रेट डेटा रिमेन सीक्रेट। या उस डेटा का ओनर चाहता है कि वो डेटा कुछ सर्टेन लोगों के द्वारा ही एक्सेस किया जाए और किसी भी के द्वारा नहीं। ऐसे में सिस्टम जिम्मेदार रहेगा कि जब डेटा का रिलीज हो तब अनऑथोरायज्ड लोगों तक किसी भी हालत में नहीं पहुंचना चाहिए। सिर्फ डेटा ओनर ही डिसाइड कर सकता है कि उस डेटा में से कौन क्या देख सकता है।




**2. Integrity इंटीग्रिटी – अखंडता –**

इसका मतलब ये है कि अनऑथोरायज्ड यूजर्स, बिना ओनर परमिशन के किसी भी डेटा को मोडिफाई नहीं कर पाए। डाटा मॉडिफिकेशन में डेटा के मोडिफाई करने के साथ–साथ डेटा को रिमूव करना और एक्सिस्टिंग डेटा में फाल्स डेटा ऐड करने से भी है। एक सिस्टम की ये जिम्मेदारी है कि सिस्टम में डाला गया डेटा, जब तक ओनर ना चाहे तब तक अनचेंज्ड रहना चाहिए ।

**3. Availability अवेलेबिलिटी – उपलब्धता –**

इसका मतलब ये है कि कोई भी सिस्टम को डिस्टर्ब करके अनुपयोगी (अनयूजेबल) नहीं बना सके। आजकल (डिनायल ऑफ सर्विस) डॉस अटैक बढ़ते जा रहे हैं।

**डॉस अटैक (डिनायल ऑफ सर्विस) –**

इस अटैक में इंटरनेट सर्वर पर एक साथ हजारों रिक्वेस्ट (रिक्वेस्ट्स की बाढ़) भेजी जाती हैं जिसको आप अनयूजेवल ट्रैफिक कह सकते हैं। जब एक साथ इतनी रिक्वेस्ट भेजी जाती हैं तो ये सीपीयू के टाइम को खा जाती हैं सीपीयू रिक्वेस्ट्स को एक्सामिन और डिस्कार्ड करने में ही पागल हो जाता है। मान लीजिये सीपीयू किसी वेबपेज को रीड करने की इनकमिंग रिक्वेस्ट को प्रोसेस करने में 100 म्यूसेकण्ड्स लेता है और ऐसे में कोई 10000 या इससे अधिक रिक्वेस्ट प्रति सेकंड सर्वर को भेजे तो ये सर्वर की कहानी खत्म कर देगा।

जैसे–जैसे समय बदलता गया कंप्यूटर सिक्योरिटी एक्सपर्ट्स ने सोचा ये तीन फंडामेंटल प्रॉपर्टीज काफी नहीं है तो उन्होंने कुछ एडिशनल प्रॉपर्टीज और जोड़ दीं। जैसे–

- **ऑथेंटिसिटी**
- **अकॉउंटेबिलिटी**
- **नॉन–रिपुड़ीऐबिलिटी**
- **प्राइवेसी इत्यादि।**

**ब्लैक हैट हैकर्स (काली टोपी वाले ताऊ लोग)** –

ये इंटरनेशनल हैकर्स की एक कम्युनिटी या गैंग है। ये हाईली एडवांस्ड टूल और सर्विस से अटैक करते हैं।

**वाइट हैट क्रैकर्स (सफैद टोपी वाले चाचा लोग)** –

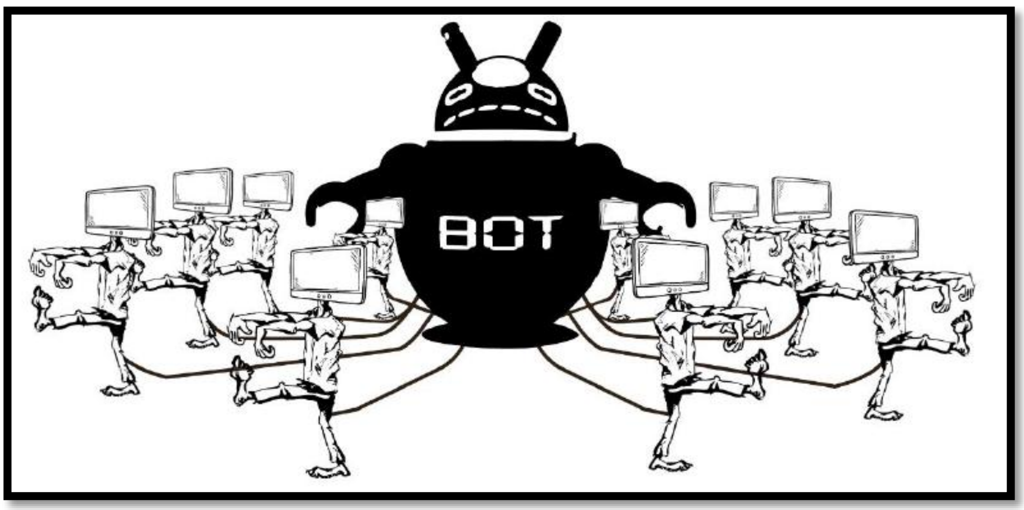
ये भी इंटरनेशनल हैकर्स की एक कम्युनिटी या गैंग है। ये हाईली एडवांस्ड टूल और सर्विस से अटैक को रोकने की सर्विस देते हैं। और सबसे ज्यादा अटैक टूल भी इन्होने ही डेवलप किये हैं। क्युकी इनके टूल्स से कंप्यूटर सिस्टम्स और नेटवर्क सिक्योरिटी को टेस्ट किया जाता है।

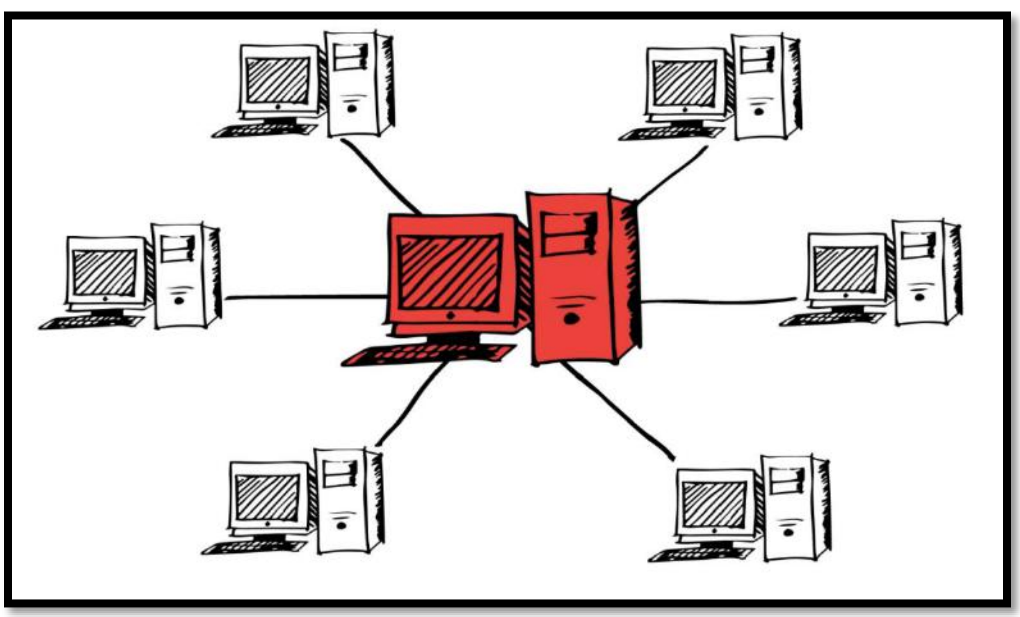
**बोट नेट्स –**

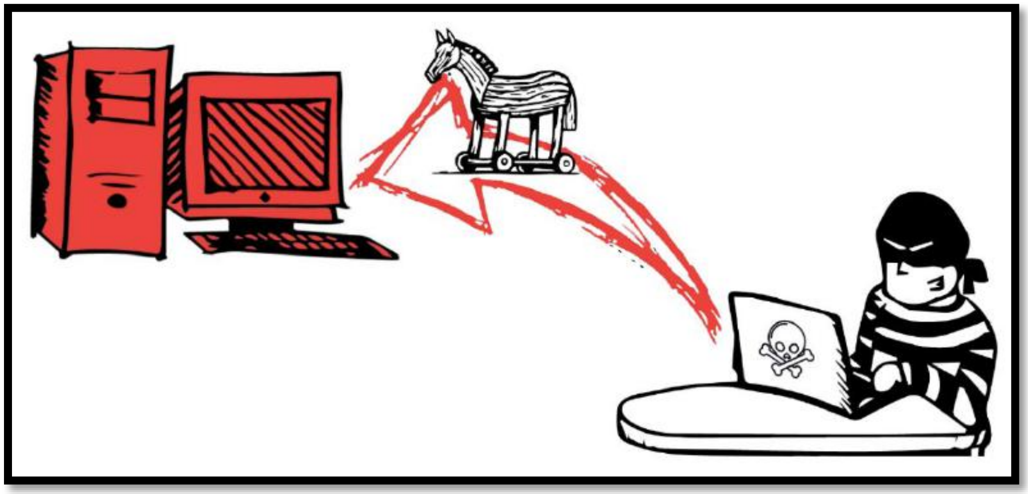
ये एक इंटरनेट पर इंटरकनेक्टेड साइबर इंफ्रास्ट्रक्चर है जहाँ पर साइबर क्रिमिनल एक्टिविटीज चलती हैं। एक बोटनेट इंफ्रास्ट्रक्चर में लाखों या करोड़ों हैक्ड या कम्प्रोमायज्ड कम्प्यूटर्स होते हैं। इस नेटवर्क से आपका कंप्यूटर भी जुड़ सकता है और आपको कभी पता भी नहीं चलेगा। इस तरीके से बॉटनेट्स में कंप्यूटर जुड़ते जाते हैं।

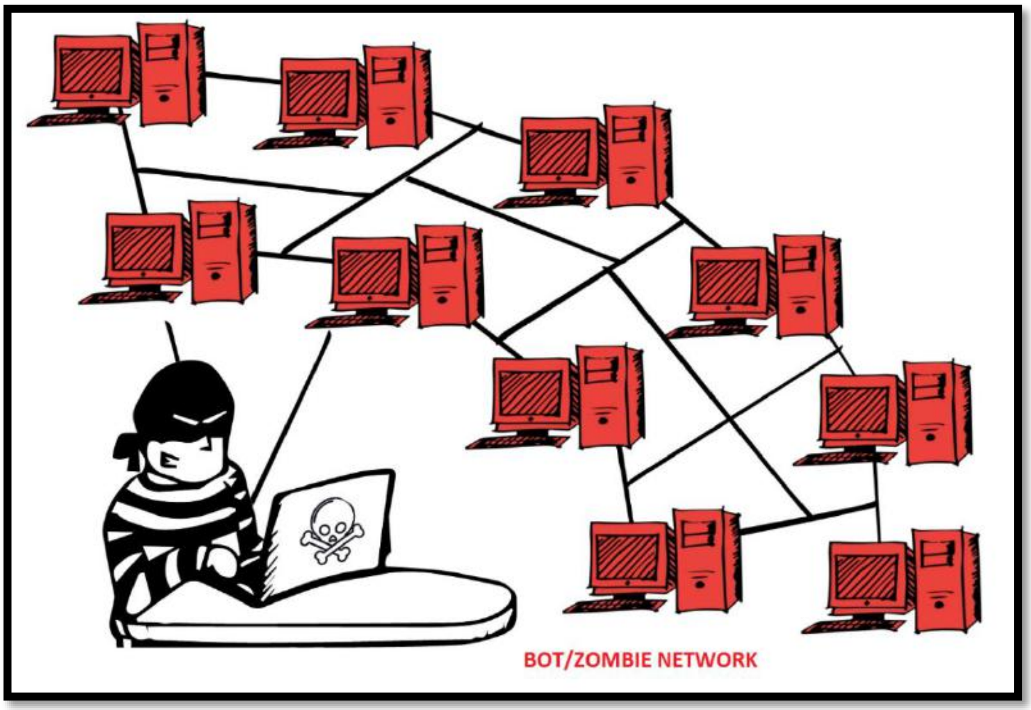
यहाँ पर हैकर्स, आम आदमी के कंप्यूटर सिस्टम पर कन्ट्रोल करने के लिए सबसे पहले इंटरनेट पर महंगे और फ्री सॉफ्टवेयरर्स को क्रैकड वर्शन के नाम से ऑफर करते हैं और जब आप उसको इनस्टॉल करते हैं तो अटैकर को आपके कंप्यूटर का फुल कन्ट्रोल मिल जाता है जैसे किसी स्ट्रेंजर को अपनी घर की चाबी देना।

जब आपका कंप्यूटर अटैकर के कन्ट्रोल में हो जाता है तो वो बॉटनेट्स का एक मेंबर बन जाता है जिसको **बोट या जोंबी** कहते हैं। और सबसे बड़ी बात कंप्यूटर यूजर को ये सब कुछ दिखता नहीं। कभी–कभी ये अटैक कंप्यूटर से बाहर आकर भी नुक्सान पंहुचा सकते हैं। ऑस्ट्रेलिया के क्वींस लैंड में एक सीवेज सिस्टम कंपनी ने एक एम्प्लोयी को नौकरी से बाहर निकाल दिया था तो उसका दिमाग ठनका और उसने सीवेज सिस्टम के सर्वर को हैक कर, सर्वर को अपने कन्ट्रोल में ले लिया और उसने मिलियंस लीटर सीवेज को शहर के पार्क्स और नदियों और कॉस्ट्ल्स में फैला दिया था।








**ऑपरेटिंग सिस्टम सिक्योरिटी –**

मोस्ट ऑफ टाइम्स किसी कंप्यूटर सिस्टम की सिक्योरिटी को **कम्प्रोमायस** किया जा सकता है। इसका मुख्य्य कारण हम इंडियन लोग जब कोरोना से नहीं डरे तो कंप्यूटर और साइबर सिक्योरिटी तो बहोत दूर की बात है। हम लोगो में जागरूकता की कमी और लेजीपन बहोत ज्यादा है।

क्युकि ज्यादातर लोगो के पासवर्ड या तो पासवर्ड या उसकी फेवरिट मूवी नेम या फेवरिट हीरो नेम या विहिकल नंबर या खुद या फैमिली मेंबर का फर्स्ट या लास्ट नेम या इनमे से एक कॉम्बिनेशन को बनाते हैं दूसरी तरफ एटीएम पिन की बात करें तो वो चार जीरो से चार नायन या बाइक स्कूटर या कार नंबर या मोबाइल नंबर में से चार नंबर हो सकते हैं। ऐसे में वो अपने आपको बहोत स्मार्ट और जागरूक समझते हैं। और कुछ टेम्परेरी ताऊ लोग कुछ ज्यादा ही कॉम्पलीकेट पासवर्ड या पिन बनाते हैं और फिर गजनी बन जाते हैं। कुछ गजनी लोग अपने पासवर्ड या पिन को डायरी वगेराह में लिख के रखते हैं जहां से ऑफिस स्टाफ या सेक्रेटरी या विजिटर या सफाई कर्मचारी या विजिटर के हाथ लग सकता हैं। ऐसे में आप ही बताओ कंप्यूटर क्या सुसायड करें ? और कुछ लापरवाह लोग भी होते हैं जो अपनी यूएसबी पेन ड्राइव को खो देते हैं जिसमें सेंसिटिव इनफार्मेशन हो सकती है। और कुछ लोग अपने कंप्यूटर या हार्ड डिस्क में से डाटा को डिलीट करने के बाद उनको को बेच देते जहां से हैकर्स या अटैकर्स डाटा को वापस निकाल लेते हैं।

**पॉसिबिलिटी ऑफ बिल्डिंग सिक्योर सिस्टम –**

अगर कोई सॉफ्टवेयर ज्यादा बड़ा या काम्प्लेक्स या कॉम्प्लिकेट नहीं है तो सॉफ्टवेयर को बग फ्री बनाया जा सकता है और उसको अच्छे से वेरिफाई कर चेक भी किया जा सकता है कि वो बग फ्री है। लेकिन आज के टाइम में सॉफ्टवेयर बहोत काम्प्लेक्स या कॉम्प्लिकेट होते जा रहे हैं। और ऐसे में कही ना कहीं उसमें बग का रह जाना संभव है।

**सिक्योरिटी Vs फीचर्स ( ऑथर व्यू ) –**

सिक्योरिटी का सबसे बड़ा दुश्मन फीचर्स हैं। सिक्योर सिस्टम तभी पॉसिबल है जब वो सिंपल और लिमिटेड फीचर्स वाला हो। क्युकि आजकल कोई क्लाइंट सॉफ्टवेयर बनवाने से पहले बहोत सारे फीचर्स को लिस्टेड करता है फिर सॉफ्टवेयर कंपनी के टेक एक्सपर्ट्स सिस्टम आर्किटेक्ट या डिजाइनर पर उनको डिजाईन करवाने में लग जाते हैं इसका एक ही नतीजा होगा वो है सॉफ्टवेयर ज्यादा काम्प्लेक्स और कोड लाइन्स ज्यादा, साथ ही इनाम के रूप में ज्यादा बग्स और सिक्योरिटी एरर्स। मैंने कई सिक्योरिटी और सॉफ्टवेयर कॉन्फ्रेन्सेस में कम्पनीज को ये बात कई बार गहराई से बताई भी है।

**एक्साम्प्ल –** जैसे पहले मेल भेजते थे तो ई–मेल में केवल ASCII टेक्स्ट हुआ करता था। और कोई भी सिक्योरिटी इशू नहीं था नाही ईमेल के जरिये वायरस इत्यादि फैलते थे। लेकिन जैसे ही ई–मेल सिस्टम में अटैचमेंट फीचर आये तो वर्ड डाक्यूमेंट्स जिनमें मैक्रो प्रोग्राम्स होते हैं साथ ही सॉफ्टवेयर्स या एमपीथ्री सहित अन्य फाइल्स को भेजने का ऑप्शन मिलता है। ऐसे में किसी और के द्वारा भेजा गया विदेशी डॉक्यूमेंट सुरक्षित है इसकी क्या गारंटी है। तो ये ज्यादा फीचर्स का ही कमाल है।

**एक्साम्प्ल –** पहले वेब पेजेज में एचटीएमएल यूज होती थी और कोई सिक्योरिटी प्रॉब्लम भी नहीं थी लेकिन हमने ज्यादा फीचर्स की डिमांड की और फिर जावा स्क्रिप्टिंग और ऐप्लेट्स जैसे प्रोग्राम आये ऐसे में यूजर्स को कंटेंट देखने के लिए उनको रन करवाना होता है तो पॉप अप आता है और बर्बाद करके चला जाता है।

**टिप –** आजकल अटैकर्स और हैकर्स आपको एक Fake Mail या मैसेज भेज सकते हैं जिसमें आपको बताया जाएगा कि में विदेश में हु और मेरे पास इतना मिलियन पैसा है और में आपके साथ बिजनेस करना चाहता हूँ या फिर आपको बताया जाएगा आपने लॉटरी में इतने करोड़ जीते हैं और हम आपको ये पैसा भेजना चाहते है। बस इतना पढ़ते ही तो हम में से आधे लोग खुशी से पागल हो जाते हैं। और वो इस बात को सीक्रेट के रूप में रखते हैं और रात को सपने देखतें है एक ऑडी या फेंटम ले लूंगा और एक आलिशान मकान और एक बिजनेस जरूर ओपन करूँगा। और अगले दिन वो बर्बाद हो जाता है। इसलिए याद रखें फोकट में जहर और पानी भी नहीं मिलता तो तुमको करोड़ों कहाँ से मिल जाएंगे। सावधान रहें और स्मार्ट और अनुभवी बने।

**Trusted Computing Base (टीसीबी) —**

ये ऐसे सिस्टम्स होते है जिसमें सिस्टम सिक्योरिटी को कोम्प्रोमाईज नहीं किया जा सकता। कुछ डिपार्टमेंट्स के लिए सिक्योरिटी सबसे इम्पोर्टेन्ट फीचर होता है उनको नए या बहोत सारे फीचर्स नहीं चाहिए उनको सिर्फ और सिर्फ टाइट सिक्योरिटी सिस्टम्स चाहिए होते हैं। जैसे मिलिट्री या सीक्रेट सर्विस डिपार्टमेंट्स। ऐसे सिस्टम्स में सिक्योरिटी को प्राइम ऑब्जेक्ट मानकर सिक्योरिटी रूल्स बनाकर उनके आधार पर हार्डवेयर और सॉफ्टवेयर को चूज किया जाता है। और कोई सिस्टम लिमिटेड वर्किंग स्पेसिफिकेशन्स के साथ डिजाइन किया जाए तो उसकी सिक्योरिटी को कोम्प्रोमाईज करना आसान नहीं है।

टीसीबी में आमतौर पर अधिकांश हार्डवेयर होते हैं (i/o) उपकरणों को छोड़कर जो सिक्योरिटी को प्रभावित ना करें। इनमें ऑपरेटिंग सिस्टम की कर्नेल का कुछ पोरशन और साथ में ऐसे यूजर प्रोग्राम जिनमें सुपर यूजर पॉवर है जैसे– यूनिक्स में SETUID रुट प्रोग्राम्स के अलावा, ऑपरेटिंग सिस्टम के मुख्य फंक्शन्स जो टीसीबी का हिस्सा होना चाहिए जैसे– प्रोसेस क्रिएशन, प्रोसेस स्विचिंग, मैमोरी मैनेजमेंट, फाइल मैनेजमेंट, आई/ओ मैनेजमेंट। डिजाइनिंग के दौरान टीसीबी प्लेटफार्म मोडूयल बाकी के ऑपरेटिंग सिस्टम से सेपरेट रहता है जिससे उसका साइज कम रहता है और साथ ही उसकी करेक्टनेस को आसानी से वेरिफाई किया जा सकता है।

**रेफ्रेन्स मॉनिटर –**

रेफ्रेन्स मॉनिटर, टीसीबी प्लेटफार्म मोडूयल का एक इम्पोर्टेन्ट पार्ट होता है। ये रेफ्रेन्स मॉनिटर ही सभी सिस्टम कॉल्स को एक्सेप्ट करता है जैसे– सिक्योरिटी कॉल्स, ओपनिंग फाइल्स, और रेफ्रेन्स मॉनिटर डिसीजन लेता है कि ये कॉल प्रोसेस्ड करनी है कि नहीं ? रेफ्रेन्स मॉनिटर ही सारे सिक्योरिटी डिसीजन्स को एक जगह पर अलाउ करता है जिससे सिस्टम बाईपासिंग की पॉसिबिलिटी ना रहे लेकिन जनरल ऑपरेटिंग सिस्टम्स इस तरीके से डिजाइन नहीं किये जाते। इसलिए वो इनसिक्योर होते हैं।

**1. जब किसी सॉफ्टवेयर में कोई एरर या फाल्ट रह गया हो तो उसे कहते हैं –**

Ⓐ एरर इन सॉफ्टवेयर Ⓑ फॉल्टी सॉफ्टवेयर
Ⓒ सॉफ्टवेयर बग Ⓓ वल्नरेबिलिटी

**2. जब सॉफ्टवेयर में कोई सिक्योरिटी बग हो तो उसे कहते हैं –**

Ⓐ एरर इन सॉफ्टवेयर Ⓑ फॉल्टी सॉफ्टवेयर
Ⓒ सॉफ्टवेयर बग Ⓓ वल्नरेबिलिटी

**3. हैकर्स सिक्योरिटी बग में सही इनपुट ट्रिगर कर उस सॉफ्टवेयर में सेंध मार देते हैं। जिसको सिक्योरिटी की भाषा में कहते हैं –**

Ⓐ क्रेकिंग Ⓑ एक्सप्लॉइट
Ⓒ वल्नरेबिलिटी Ⓓ बग

**4. अटैकर्स एक्सप्लॉइट्स को लांच कर सकते हैं –**

Ⓐ मैनुअली Ⓑ ऑटोमेटिकली
Ⓒ दोनों तरीकों से Ⓓ दोनों में से कोई नहीं

**5. डॉस अटैक (DOS-ATTACK) का पूरा नाम है –**

Ⓐ डिनायल ऑफ सिक्योरिटी Ⓑ डिनायल ऑफ सिस्टम
Ⓒ डिस्ट्रीब्यूशन ऑफ सर्विस Ⓓ डिनायल ऑफ सर्विस

**6. डी–डॉस अटैक या (डी.डी.ओ.एस अटैक) का पूरा नाम है –**

Ⓐ डिवायड डिनायल ऑफ सिक्योरिटी
Ⓑ डायरेक्ट डिनायल ऑफ सिस्टम
Ⓒ डिमांड डिनायल ऑफ सर्विस
Ⓓ डिस्ट्रीब्यूटिड डिनायल ऑफ सर्विस

**7. कंप्यूटर वायरस एक कंप्यूटर से दूसरे कंप्यूटर में किस माध्यम से सर्वाधिक पहुंच करते हैं –**

Ⓐ इंटरनेट Ⓑ इंट्रानेट
Ⓒ एक्सट्रानेट Ⓓ पेन ड्राइव

**8. (थ्रेट्स) कई सिक्योरिटी जर्नल्स किसी इनफार्मेशन सिस्टम की सिक्योरिटी को तीन कंपोनेंट्स में बांटते हैं। जिनको सिक्योरिटी साइंस में सीआईए CIA कहते हैं। निम्न में से कौनसा कॉम्पोनेन्ट CIA से सम्बंधित नहीं है –**

Ⓐ कॉन्फिडेंटिअलिटी Ⓑ क्रेडिबिलिटी
Ⓒ इंटीग्रिटी Ⓓ अवेलेबिलिटी

**9. कॉन्फिडेंटिअलिटी में किस चीज का खतरा रहता है –**

Ⓐ डिनायल ऑफ सर्विस Ⓑ टेम्परिंग विथ डाटा
Ⓒ एक्सपोजर ऑफ डाटा Ⓓ उपयुक्त सभी

**10. डाटा इंटीग्रिटी में किस चीज का खतरा रहता है –**

Ⓐ डिनायल ऑफ सर्विस Ⓑ टेम्परिंग विथ डाटा
Ⓒ एक्सपोजर ऑफ डाटा Ⓓ उपयुक्त सभी

**11. डाटा अवेलेबिलिटी में किस चीज का खतरा रहता है –**

Ⓐ डिनायल ऑफ सर्विस Ⓑ टेम्परिंग विथ डाटा
Ⓒ एक्सपोजर ऑफ डाटा Ⓓ उपयुक्त सभी

**12. रैंसमवेयर नाम के मिलिसियस सॉफ्टवेयर (मॉलवेयर) ने दुनिया के कितने देशों को अपने कब्जे में कर लिया था –**

Ⓐ 80 Ⓑ 120
Ⓒ 150 Ⓓ 180

**13. ऐसा इंटरकनेक्टेड साइबर इंफ्रास्ट्रक्चर जिसमें लाखों या करोड़ों हैक्ड या कम्प्रोमायज्ड कम्प्यूटर्स होते हैं कहलाता है –**

Ⓐ हैकिंग नेटवर्क Ⓑ डार्क वेब
Ⓒ साइबर नेटवर्क Ⓓ बोट नेट्स

**14. मान लीजिये कोई कंप्यूटर, अटैकर के कन्ट्रोल में हो जाता है और वो बॉटनेट्स का एक मेंबर बन जाता है ऐसे में वो कंप्यूटर कहलायेगा –**

Ⓐ ट्रोजन हॉर्स Ⓑ बोट
Ⓒ जोंबी Ⓓ बोट या जोंबी

**15. मोस्ट ऑफ टाइम्स किसी कंप्यूटर सिस्टम की सिक्योरिटी को कम्प्रोमायस किया जा सकता है। इसका कारण हो सकता है –**

Ⓐ साइबर सिक्योरिटी के प्रति लोगों में जागरूकता की कमी
Ⓑ सिंपल और जनरल पासवर्ड रखना
Ⓒ पासवर्ड्स को लिखकर रखना / आई.टी शिक्षा की कमी
Ⓓ उपयुक्त सभी

**16. पॉसिबिलिटी ऑफ बिल्डिंग सिक्योर सिस्टम थ्योरी के बारे में सत्य कथन है –**

Ⓐ अगर सॉफ्टवेयर ज्यादा बड़ा या काम्प्लेक्स या कॉम्प्लिकेट नहीं है तो सॉफ्टवेयर को बग फ्री बनाया जा सकता है।
Ⓑ अगर कोई सॉफ्टवेयर ज्यादा काम्प्लेक्स या कॉम्प्लिकेट नहीं है तो उसको अच्छे से वेरिफाई कर चेक भी किया जा सकता है कि वो बग फ्री है।
Ⓒ आज के टाइम में सॉफ्टवेयरर्स बहोत काम्प्लेक्स या कॉम्प्लिकेट होते जा रहे हैं। और ऐसे में कही ना कहीं उसमें बग का रह जाना संभव है।
Ⓓ सॉफ्टवेयर में जितने कम फीचर्स होंगे उतने कम बग्स होंगे।
Ⓔ उपयुक्त सभी

**17. (टीसीबी) सिस्टम्स का पूरा नाम है –**

Ⓐ ट्रस्टेड कंप्यूटर बुक Ⓑ ट्रस्टेड कंप्यूटर बग्स
Ⓒ ट्रस्टेड कंप्यूटिंग बेस Ⓓ ट्रस्टेड कमांड बेस

**81. निम्न में से कौनसा कंप्यूटर वायरस डिस्क के मास्टर बूट रिकॉर्ड को इन्फेक्ट करता है –**

Ⓐ मैक्रो वायरस Ⓑ मैमोरी रेजिडेंट वायरस
Ⓒ डिवाइस ड्राइवर वायरस Ⓓ बूट सेक्टर वायरस

**82. सामान्यतः "कंप्यूटर वायरस" कितने प्रकार के होते है –**

Ⓐ 3 Ⓑ 5
Ⓒ 7 Ⓓ 10

**83. कौनसा कंप्यूटर वायरस खुद के टाइप और सिग्नेचर को चेंज करता रहता है जिससे उसको आइडेंटिफाई करना कठिन हो जाता है**

Ⓐ बूट सेक्टर वायरस Ⓑ मैमोरी रेजिडेंट वायरस
Ⓒ डिवाइस ड्राइवर वायरस Ⓓ पोलीमॉर्फिक वायरस

**84. निम्न में से कौनसा कंप्यूटर वायरस फाइल्स को इन्फेक्ट करने के तुरंत बाद डिलीट कर देता है –**

Ⓐ ओवर रायट वायरस Ⓑ बूट सेक्टर वायरस
Ⓒ डिवाइस ड्राइवर वायरस Ⓓ पोलीमॉर्फिक वायरस

**85. निम्न में से कौनसा वायरस "कैविटी वायरस" कहलाता है जो मेमोरी के एम्प्टी खाली स्पेस को फिल–अप करके मेमोरी वेस्टेज कर देता है –**

Ⓐ ओवर रायट वायरस Ⓑ बूट सेक्टर वायरस
Ⓒ स्पेस फिलर वायरस Ⓓ पोलीमॉर्फिक वायरस

**86. कंप्यूटर वायरस या वोर्म्स किस कैटेगरी में आते है –**

Ⓐ सॉफ्टवेयर Ⓑ फर्मवेयर
Ⓒ मॉलवेयर Ⓓ हार्डवेयर

**87. आईडेन्टिटी थेफ्ट कहते हैं –**

Ⓐ सोशल सिक्योरिटी नंबर्स को चुराना
Ⓑ बैंक अकाउंट डिटेल्स को चुराना
Ⓒ क्रेडिट कार्ड नंबर्स को चुराना
Ⓓ उपयुक्त सभी

**88. स्पैम को रोका जा सकता है –**

Ⓐ स्पैम भेजने वाले ईमेल ऐड्रेस्सेस को ब्लॉक करके
Ⓑ कीवर्ड्स के आधार पर ईमेल फिल्टर यूज करके
Ⓒ स्पैमर के सब्सक्रिप्शन को अनसब्सक्राइब करके
Ⓓ उपयुक्त सभी से

**89. मैक्रो वायरस के बारे में सत्य कथन है –**

Ⓐ इनको फैलने के लिए OS पर निर्भर रहना पड़ता है
Ⓑ ये अन्य वायरस से साइज में बड़े होते हैं
Ⓒ फैलने के लिए पर्टिकुलर एप्लीकेशन पर निर्भर रहना पड़ता है
Ⓓ लो–लेवल लैंग्वेज में रिटन होने से अनडिटेक्टेड रहते हैं

**90. निम्न में से कौनसा कथन कंप्यूटर वायरस और वोर्म्स को डिफरेंसिएट करता है –**

Ⓐ वायरस केवल सिंगल मशीन को इन्फेक्ट करता है
Ⓑ वोर्म्स मल्टीपल मशीन्स को इन्फेक्ट करते हैं
Ⓒ किसी मशीन को इन्फेक्ट करने के लिए वायरस को यूजर इंटरेक्शन की आवश्यकता होती है
Ⓓ किसी मशीन को इन्फेक्ट करने के लिए वोर्म्स को यूजर इंटरेक्शन की आवश्यकता होती है

**91. निम्न में से कौनसा एक एंटी वायरस सॉफ्टवेयर नहीं है –**

Ⓐ BitDefender Ⓑ नॉर्टन NorTon
Ⓒ अविरा Avira Ⓓ Allen Musk

**92. निम्न में से कौनसा एक एंटी वायरस सॉफ्टवेयर नहीं है –**

Ⓐ एफ सिक्योर Ⓑ कैस्पर स्काई
Ⓒ ट्रेंड माइक्रो Ⓓ जुपिटर प्लस

**93. निम्न में से कौनसा एक एंटी वायरस सॉफ्टवेयर नहीं है –**

Ⓐ मैकेफी McAfee Ⓑ ईसेट नोड 32
Ⓒ बुल गार्ड Ⓓ पांडा मस्टंग

**94. निम्न में से कौनसा एक एंटी वायरस सॉफ्टवेयर नहीं है –**

Ⓐ मॉलवेयर बायट्स Ⓑ वायरस बुलेट
Ⓒ सोफोस होम Ⓓ ऐवीजी AVG

**95. निम्नलिखित में से कौनसा एक प्रकार का ऐसा कंप्यूटर प्रोग्राम है जो या तो होने का दिखावा करता है, या होने के रूप में डिस्क्रायब किया गया होता है, या वो उपयोगी या वांछनीय सुविधाओं का एक सेट होने का इलूजन पैदा करता है लेकिन वास्तव में वो एक हानिकारक कोड होता हैं कहलाता है –**

Ⓐ Viruses Ⓑ Worm
Ⓒ Adware Ⓓ Trojans Bot

**96. निम्न में से कौनसा प्रोग्राम या सॉफ्टवेयर खुद को सेल्फ रिप्लीकेट करने के साथ साथ फाइल और सिस्टम को डेमेज करता है –**

Ⓐ ट्रोजन हॉर्स Ⓑ बोट्स
Ⓒ वायरस Ⓓ वोर्म्स

**97. निम्न में कौनसा सॉफ्टवेयर टर्म– वायरस, वोर्म्स, ट्रोजन हॉर्स, ऐडवेयर, इन सब के लिए सयुक्त रूप से लिया जाता है –**

Ⓐ की–लॉगर Ⓑ बैकडोर्स
Ⓒ मॉलवेयर Ⓓ बोट्स
Ⓔ स्पाईवेयर

**98. निम्न में से कौनसा मॉलीसियस सॉफ्टवेयर आपके कंप्यूटर में डाउनलोड होने के बाद आपकी हार्ड डिस्क को स्कैन करके उसमे से personal इंफॉर्मेशंस और इंटरनेट ब्राउजिंग हैबिट्स सहित आपकी जासूसी करता है –**

Ⓐ की–लॉगर
Ⓑ बैकडोर्स
Ⓒ मॉलवेयर
Ⓓ बोट्स
Ⓔ स्पाईवेयर

**99. निम्न में से कौनसा मॉलीसियस सॉफ्टवेयर अटैकर के द्वारा आपके कंप्यूटर का रुट या एडमिनिस्ट्रेटिव एक्सेस गेन करने के लिए डिजाइन किया जाता है –**

Ⓐ बैकडोर्स
Ⓑ मॉलवेयर
Ⓒ स्पाईवेयर
Ⓓ रुटकिट्स (Rootkits)
Ⓔ एंटीवेयर

**100. निम्न में से किस अटैक के जरिये नेटवर्क में से ट्रांसमिट होने वाले स्माल पैकेट्स को कैप्चर कर पैकेट्स के डेटा को रीड किया जाता है – The attack that focuses on capturing small packets from the network transmitted by other computers and reading the data content in search of any type of information is ____**

Ⓐ ईव्स–ड्रॉप्पिंग (Eavesdropping)
Ⓑ डिनायल ऑफ सर्विस (Denial of service)
Ⓒ एक्सप्लॉइट्स (Exploits)
Ⓓ पिशिंग (Phishing)

# CHAPTER ∞ C4 ∞
# (Control Statements (Looping))

**परिचय –**

कभी–कभी हम चाहते हैं कि हमारे कोड का कुछ पार्ट एक से अधिक बार Execute हो। इसके लिए हम या तो अपने प्रोग्राम में कोड रिपीट कर सकते हैं या रिपीट करने के बजाय Loop का उपयोग कर सकते हैं। लेकिन मान लीजिये हमें कोड के कुछ पार्ट को सौ या सौ से अधिक बार Execute करने की आवश्यकता है तो कोड को रिपीट करना व्यावहारिक नहीं है। बल्कि इसके बजाये हम अपने रिपीट किये जाने वाले कोड का उपयोग Loop के अंदर कर सकते हैं।

```
while(condition)
{
    code or statement(s);
}
```

ध्यान दीजिये Loop, प्रोग्राम के कुछ हिस्से को Specified number of time तक या जब तक कोई Particular Condition को सेटिस्फाई नहीं किया जा रहा है। तब तक रिपीट करता रहता है। या दूसरे शब्दों में आप कह सकते हैं, Looping किसी सिंगल स्टेटमेंट या स्टेटमेंट्स के ग्रुप को तब तक रिपीट करने की प्रोसेस है जब तक कि लूप को समाप्त करने के लिए कुछ कंडीशन सेटिस्फाई न हो जाए। C प्रोग्रामिंग लैंग्वेज में कुछ लूप कमांड होते हैं। हम इन कमांड्स को आगे देखेंगे।

**लूप के चार भाग होते हैं –**

1- Initialization.
2- Conditions.
3- Statements.
4- Incrementation or Decrementation.

**Type of loops -**

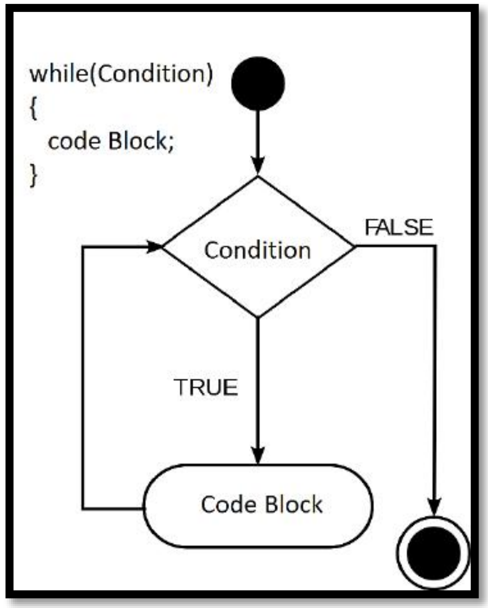
- **Entry control loop:**
  ऐसे Loops जिसमें स्टेटमेंट के Execution से पहले Condition को टेस्ट किया जाता है। और अगर शुरुआत में ही कंडीशन False है तो Loop एक बार भी रन नहीं होगा। e.g. for loop, while loop.

- **Exit control loops:**
  ऐसे Loops जिनमें स्टेटमेंट के Execution के बाद Condition को टेस्ट किया जाता है। इस प्रकार यदि शुरुआत में ही कंडीशन False है तो लूप at least एक बार रन होगा। e.g. do while loop

**1. while loop –** C में तीन Loops हैं, 'while' Loop उन सभी में सबसे सरल लूप है। इसको एक Condition (कोष्ठकों में, **if** स्टेटमेंट की तरह) दिया जाता है जिसका यह मूल्यांकन (Evaluate) करता है। यदि मूल्यांकन (Evaluation) के दौरान कंडीशन True (non zero) पाई जाती है तो Loop की बॉडी (Code Block) Execute हो जाती है। इसके बाद कंडीशन को फिर से Evaluate (मूल्यांकन) किया जाता है, यदि अभी भी कंडीशन True है, तो लूप की बॉडी (Code Block) फिर से Execute हो जाती है। और यह प्रोसेस तब तक जारी (Continues) रहेगी जब तक कि कंडीशन फाइनली False नहीं हो जाती। फिर Execution लूप के बाद आने वाले पहले स्टेटमेंट पर जाता है।

'while' लूप एक कंडीशन या कमांड के एक ब्लॉक से बना होता है जो एक लूप में चलना चाहिए। जैसा कि हमने पहले बताया है, कमांड्स का ब्लॉक दो ओपनिंग और क्लोजिंग ब्रेसेस में संलग्न कमांड्स की एक सीरीज होता है। 'while' लूप में, कंडीशन का मूल्यांकन किया जाता है और यदि यह True है तो लूप का स्टेटमेंट Execute किया जाता है।

Statement के Execution के बाद और (increment or decrement), लूप की कंडीशन को फिर से टेस्ट किया जाता है। स्टेटमेंट्स के बार–बार एक्सिक्यूशन की प्रोसेस, (increment or decrement), और कंडीशन की टेस्टिंग, तब तक जारी (Continuous) रहती है जब तक कि कंडीशन लास्ट में False न हो जाए। और लूप का कन्ट्रोल फिर अगले स्टेटमेंट में ट्रांसफर हो जाता है। ध्यान दीजिये यदि लूप की बॉडी में एक से अधिक स्टेटमेंट की आवश्यकता है, स्टार्ट और एन्ड ठतंबमे का उपयोग किया जाना चाहिए। इस लूप का उपयोग इस प्रकार किया जाता है –




| **Syntax** | **Free Syntax** |
|---|---|
| <pre>initialize;<br>while(condition)<br>{<br>statement;<br>increment or decrement;<br>}</pre> | <pre>while( condition )<br>        command;<br>while( condition )<br>      {<br>        block of commands<br>      }</pre> |

Loop condition एक बूलियन एक्सप्रेशन है। एक बूलियन एक्सप्रेशन एक logical statement होता है जो या तो सही होता है या फिर गलत होता है। यदि logical स्टेटमेंट Valid होता है तो इसका मान 1 होता है और यदि Valid नहीं है तो इसका मान 0 होता है। उदाहरण के लिए Boolean statement (3>4) यहाँ Invalid है इसलिए इसका मान 0 है। जबकि स्टेटमेंट (10==10) एक Valid लॉजिकल स्टेटमेंट है इसलिए इसका मान 1 है।

**Semicolon Warning! -** while के बाद Semicolon लगाने से बचें। जिस प्रकार हम if स्टेटमेंट के बाद Semicolon नहीं लगाते। इसलिए Loops में भी आप इस बात को ध्यान रखिये।

```
#include<stdio.h>
#include<conio.h>
main()
{
int i=0;
while( i<100 )
{
printf("\n i=%d",i);
i=i+1;
}
system("pause");
}
```

उपरोक्त उदाहरण में i=i+1 का अर्थ है कि i में 1 जोड़ें और फिर इसे i को असाइन करें और सिम्पली इसकी वैल्यू को इंक्रीज कीजिये। जैसा कि हमने पहले देखा, C प्रोग्रामिंग लैंग्वेज में एक स्पेशल ++ ऑपरेटर होता है जो यही काम करता है। हम i=i+1 की जगह पर i++ का उपयोग कर सकते हैं।

**Write a program** – To accept scores of a person and calculate sum of them and their average and print them.

```
#include<stdio.h>
#include<conio.h>
main()
{
   int count=0;
   float num=0,sum=0,avg=0;
     printf("Enter score (-1 to stop): ");
     scanf("%f",&num);
while(num>=0)
{
   sum=sum+num;
   count++;
   printf("Enter score (-1 to stop): ");
   scanf("%f",&num);
}
   avg=sum/count;
   printf("\nAverage=%f",avg);
   printf("\nSum=%f\n",sum);
   system("pause");
}
```

Here are the output results of a sample run:

**Enter score (-1 to stop):** 12
**Enter score (-1 to stop):** 1480
**Enter score (-1 to stop):** -1
**Average=**13.000000
**Sum=**26.000000

**Print the sequence 1, 2, 3, 4, 5,...N**

```
int i,n;
        printf("Enter the value of N ");
        scanf("%d",&n);
        i=1;
while(i<=n)
       {
          printf("%d,",i);
           i++;
        }
   printf("\b ");  /*to remove the comma (,) printed at the last */
```

इस उदाहरण में हमें पहले नंबर मिलता है और फिर Loop में Entry करते हैं। जब तक यूजर 0 से छोटी वैल्यू एंटर नहीं करता तब तक हम Loop के अंदर ही रहेंगे। यदि यूजर 0 से कम वैल्यू एंटर करता है तो हम इसे STOP receiving scores के रूप में समझेंगे।

**do while loop -**

C प्रोग्रामिंग में do while loop, while लूप का "उल्टा" वर्जन है। "while" में कंडीशन पहले चैक होती है और कंडीशन True होने पर ही while loop के अंदर का Statement रन होता है।

जबकि do while लूप में कंडीशन बाद में चैक होती है। इसका मतलब है कि do while loop, में Condition तक पहुंचने से पहले Body को कम से कम एक बार Execute करने की गारंटी होती है। यदि कंडीशन False है तो लूप की body को फिर कभी भी Execute नहीं किया जाता।

**It is an exit control loop. The loop is used as follows:**

```
initialize;
do
{
statement;
increment or decrement;
}
while(condition);
```

यहां Statement के execution के बाद ही condition को Test किया जाता है और फिर increment या decrement किया जाता है। इस प्रकार के लूप में पहले body को evaluate किया जाता है और उसके बाद condition को Test करते हुए प्रोग्राम आगे बढ़ता है। इस प्रकार यदि शुरुआत में कंडीशन False है तो भी स्टेटमेंट कम से कम एक बार Execute होगा। ध्यान रहे यह एक "Exit control loop" है।

**Drill Note-** Most Important thing , do while guarantees execution at least once.

```
int j = 5;
      printf("start\n");
do
      printf("j = %i\n", j--);
while(j > 0);
      printf("stop\n");
```

**output –**

```
start
j = 5
j = 4
j = 3
j = 2
j = 1
stop
```

**WAP to Print the sequence 1, 2, 3, 4, 5,…………..N**

```
int i,n;
      printf("Enter the value of N ");
      scanf("%d",&n);
      i=1;
do
   {
      printf("%d,",i);
      i++;
   }
while(i<=n);
      printf("\b "); /*to remove the comma (,) printed at the last */
```

**for loop -**

"for loop" कुछ कुछ while लूप के समान है लेकिन यह उससे अधिक Complex (जटिल) लूप है। "for loop" एक कंट्रोल स्टेटमेंट से बनाया गया है जो यह निर्धारित करता है कि Loop (कमांड सेक्शन) कितनी बार चलेगा। कमांड सेक्शन या तो एक सिंगल कमांड या कमांड्स का एक ब्लॉक हो सकता है।

***for loop*** is also an **entry control loop** which provides a more concise loop control structure. The loop is used as:

```
for (initialize; Test condition; increment or decrement)
  {
   statement;
  }
```

# Chapter ∞ 25 ∞
# (A.I, M.L, and Initiatives)

## परिचय–

**Brief History of AI -**

John McCarthy को आर्टिफिशियल इंटेलिजेंस का फादर कहा जाता है। हालांकि फर्स्ट आर्टिफिशियल इंटेलिजेंस क्रिएट करने का सम्मान Alan Mathison Turing को दिया जाता है।

**आर्टिफिशियल इंटेलिजेंस का इतिहास –**

- ☞ वर्ष 1955 में, Allen Newell और Herbert A. Simon ने सबसे पहले आर्टिफिशियल इंटेलिजेंस प्रोग्राम को विकसित (develop) किया था। इन्होने इस प्रोग्राम का नाम लॉजिक थ्योरिस्ट (Logic Theorist) रखा।
- ☞ "आर्टिफिशियल इंटेलिजेंस" शब्द का अविष्कार 1956 में John McCarthy ने किया था John McCarthy को आर्टिफिशियल इंटेलिजेंस का जनक भी कहा जाता है।
- ☞ वर्ष 1966 में, वैज्ञानिकों ने ऐसे Algorithm को विकसित करना शुरू किया जो गणितीय समस्याओं को हल कर सकते थे। इसी वर्ष में Joseph Weizenbaum के द्वारा पहले चैटबॉट (chatbot) का आविष्कार किया गया जिसका नाम एलिसा (Elisa) रखा गया।
- ☞ वर्ष 1972 में, जापान के द्वारा पहला बुद्धिमान ह्यूमनॉइड रोबोट (intelligent humanoid robot) बनाया गया जिसका WABOT-1 नाम था।
- ☞ वर्ष 1980 में, आर्टिफिशियल इंटेलिजेंस Expert system के साथ आया। एक्सपर्ट सिस्टम इन्सान की तरह सोचने और निर्णय लेने में सक्षम था।
- ☞ अमेरिका ने वर्ष 1980 में ही आर्टिफिशियल इंटेलिजेंस का पहला राष्ट्रीय सम्मेलन (National Conference) स्टैनफोर्ड विश्वविद्यालय (Stanford University) में आयोजित किया था।
- ☞ वर्ष 1997 में, IBM Deep Blue ने विश्व शतरंज चौंपियन को हराया। IBM Deep Blue पहला कंप्यूटर था जिसने आर्टिफिशियल इंटेलिजेंस का इस्तेमाल करके किसी विश्व शतरंज चौंपियन को हराया हो।
- ☞ वर्ष 2002 में पहली बार आर्टिफिशियल इंटेलिजेंस ने वैक्यूम क्लीनर Vacuum Cleaners के रूप में आया जिसका नाम Roomba था।
- ☞ वर्ष 2006 में AI बिजनेस की दुनिया में आया। फेसबुक, ट्विटर और नेटफ्लिक्स जैसी बड़ी कंपनियों के आर्टिफीसियल इंटेलिजेंस का उपयोग करना शुरू कर दिया।
- ☞ वर्ष 2011 तक AI तकनीक काफी एडवांस और मॉडर्न हो चुकी थी। क्योंकि इस समय वह पहेलियों को सुलझाने और कठिन समस्याओं को समझने में समर्थ हो चुकी थी। इसके अलावा आर्टिफिशियल इंटेलिजेंस मुश्किल सवालों को जल्दी हल करने में सक्षम हो चूका था।
- ☞ वर्ष 2012 में, गूगल ने Google Now नाम के App को विकसित किया जो यूजर को भविष्यवाणी के रूप में जानकारी दे सकता था।
- ☞ वर्ष 2014 में, AI तकनीक से चैटबॉट को Develop किया गया जो लोगों की समस्याओ को सुनकर उनका समाधान करते थे।
- ☞ वर्ष 2020 में, Baidu ने LinearFold AI अल्गोरिथम को रिलीज़ किया, इस अल्गोरिथम का इस्तेमाल कोरोना की वैक्सीन बनाने के लिए किया गया।

### भारत में आर्टिफिशियल इंटेलिजेंस (A.I) और मशीन लर्निंग (M.L)

पिछले कुछ वर्षों से विभिन्न कारणों और मुद्दों को लेकर कृत्रिम बुद्धिमत्ता (आर्टिफिशियल इंटेलिजेंस) के बारे में बराबर चर्चा बनी हुई है। आर्टिफिशियल इंटेलिजेंस कंप्यूटर विज्ञान की एक ऐसी शाखा है, जिसका काम बुद्धिमान मशीन बनाना है। हाल ही में सरकार के थिंक टैंक नीति आयोग और गूगल के बीच इस बात पर सहमति बनी है कि दोनों भारत की उदीयमान कृत्रिम बुद्धिमत्ता (Artificial Intelliegence) और मशीन लर्निंग (Machine Learning) के पारिस्थितिकी तंत्र को बढ़ावा देने के उद्देश्य से कई पहलों पर मिलकर एक साथ काम करेंगे,

प्रिय दोस्तों, अब तक हमारे नोट्स में से अन्य परीक्षाओं में आये हुए प्रश्नों के परिणाम देखने के लिए क्लिक करें -

RAS PRE. - https://www.youtube.com/watch?v=p3_i-3qfDy8&t=1253s

Rajasthan CET Gradu. Level - https://youtu.be/gPqDNlc6UR0

Rajasthan CET 12th Level - https://youtu.be/oCa-CoTFu4A

VDO PRE. - https://www.youtube.com/watch?v=gXdAk856W18&t=202s

Patwari - https://www.youtube.com/watch?v=X6mKGdtXyu4&t=2s

PTI 3rd grade - https://www.youtube.com/watch?v=iA_MemKKgEk&t=5s

SSC GD – 2021 - https://youtu.be/2gzzfJyt6vI

| EXAM (परीक्षा) | DATE | हमारे नोट्स में से आये हुए प्रश्नों की संख्या |
|---|---|---|
| RAS PRE. 2021 | 27 अक्तूबर | 74 प्रश्न आये |
| SSC GD 2021 | 16 नवम्बर | 68 (100 में से) |
| SSC GD 2021 | 30 नवम्बर | 66 (100 में से) |
| SSC GD 2021 | 08 दिसम्बर | 67 (100 में से) |
| राजस्थान S.I. 2021 | 14 सितम्बर | 119 (200 में से) |
| राजस्थान S.I. 2021 | 15 सितम्बर | 126 (200 में से) |
| RAJASTHAN PATWARI 2021 | 23 अक्तूबर (1st शिफ्ट) | 79 (150 में से) |
| RAJASTHAN PATWARI 2021 | 23 अक्तूबर (2nd शिफ्ट) | 103 (150 में से) |

| | | |
|---|---|---|
| RAJASTHAN PATWARI 2021 | 24 अक्तूबर (2nd शिफ्ट) | 91 (150 में से) |
| RAJASTHAN VDO 2021 | 27 दिसंबर (1st शिफ्ट) | 59 (100 में से) |
| RAJASTHAN VDO 2021 | 27 दिसंबर (2nd शिफ्ट) | 61 (100 में से) |
| RAJASTHAN VDO 2021 | 28 दिसंबर (2nd शिफ्ट) | 57 (100 में से) |
| U.P. SI 2021 | 14 नवम्बर 2021 1st शिफ्ट | 91 (160 में से) |
| U.P. SI 2021 | 21नवम्बर2021 (1st शिफ्ट) | 89 (160 में से) |
| Raj. CET Graduation level | 07 January 2023 (1st शिफ्ट) | 96 (150 में से ) |
| Raj. CET 12th level | 04 February 2023 (1st शिफ्ट) | 98 (150 में से ) |


**& Many More Exams like UPSC, SSC, Bank Etc.**

**नोट्स खरीदने के लिए इन लिंक पर क्लिक करें**

**Whatsapp - https://wa.link/j53pta**

**Online order - https://bit.ly/3xjbC7f**

**Call करें - 9887809083**