B.L.D.E. Association's

S.B.ARTS & K.C.P SCIENCE COLLEGE
VIJAYAPUR

M.Sc. Computer Science

A REPORT ON

Bridge Course

Entitled

"Technical, Theoretic and Practical approaches in Higher Education"

For
M.Sc (CS)-I Sem Students

2021-2022

Resource Person

Prof(Smt) R.D.Joshi
Prof S.V. Vambase
Prof Pavankumar Mahindrakar

B.L.D.E. Association's

S.B.ARTS & K.C.P SCIENCE COLLEGE
VIJAYAPUR

M.Sc(CS) Programme

NOTICE

It is here by informed to all the M.Sc(CS) I Semester students that there will be Bridge Course from 21/12/2021-24/12/2021 .So all of you should attend and get the benefit.

R. D. Joshi
Co-ordinator
M.Sc. (C.S.) Programme
S.B.Arts & K.C.P.Science College,
Vijayapur.

IQAC, Co-ordinator
S.B.Arts & K.C.P.Science College,
Vijayapur.

Principal
S.B.Arts & K.C.P.Science College,
Vijayapur.

B.L.D.E. Association's

S.B.ARTS & K.C.P SCIENCE COLLEGE
VIJAYAPUR

M.Sc(CS) Programme

About Bridge Course

The Bridge Course is aimed to act as a buffer for the new entrants, with an objective to provide adequate time for the transition to hard-core engineering courses. During this interaction of bridge course week with the faculty and their classmates, the students will be equipped with the knowledge and the confidence needed to take on bigger challenges as future engineers of this country.

Objectives: To act as a buffer for the new entrants.

- To provide adequate time for the transition to hard-core engineering courses.
- Focus on fostering a strong sense of ethical judgment and moral fortitude.
- Applications based self-learning and intermingling of a large cross section of students from vastly

  varying backgrounds.
- A breather, to prepare themselves before courses for first year engineering commence.
- The students will be equipped with the knowledge and the confidence needed to take on bigger challenges.
- Nurture a deeper understanding of the local and global world and our place in at as concerned citizens of the world.
- Interactive and Active Learning by Doing have been weaved into the Bridge Course.
- Active learning with the help of other students.

B.L.D.E. Association's

S.B.ARTS & K.C.P SCIENCE COLLEGE
VIJAYAPUR

M.Sc(CS) Programme

Syllabus
Subject: Introduction of Operating System and its services

UNIT 1: Overview: Definition of Operating System, Functions of an Operating System.
UNIT 2: Types of Operating System: Batch operating system Time-sharing operating systems, Distributed operating System , Network operating System and Real Time operating System
UNIT 3: Services: Services , Batch Processing. Processes : Process, Program, Process Life Cycle.
UNIT 4: Linux : Components of Linux System, Basic Features , Architecture and Basic Commands.

Content

UNIT 1: Overview

An Operating System (OS) is an interface between a computer user and computer hardware. An operating system is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.Some popular Operating Systems include Linux Operating System, Windows Operating System, VMS, OS/400, AIX, z/OS, etc.

Definition

An operating system is a program that acts as an interface between the user and the computer hardware and controls the execution of all kinds of programs.

Following are some of important functions of an operating System.

- Memory Management
- Processor Management
- Device Management
- File Management
- Security
- Control over system performance
- Job accounting
- Error detecting aids
- Coordination between other software and users

UNIT 2:Types of Operating System

Operating systems are there from the very first computer generation and they keep evolving with time. In this chapter, we will discuss some of the important types of operating systems which are most commonly used.

Batch operating system,Time-sharing operating systems,Distributed operating System,Network operating System and Real Time operating System.

UNIT3:Operating System - Services

An Operating System provides services to both the users and to the programs.

- It provides programs an environment to execute.
- It provides users the services to execute the programs in a convenient manner.

Following are a few common services provided by an operating system −

- Program execution
- I/O operations
- File System manipulation
- Communication
- Error Detection
- Resource Allocation
- Protection

Batch processing

Batch processing is a technique in which an Operating System collects the programs and data together in a batch before processing starts. An operating system does the following activities related to batch processing −

- The OS defines a job which has predefined sequence of commands, programs and data as a single unit.
- The OS keeps a number a jobs in memory and executes them without any manual information.
- Jobs are processed in the order of submission, i.e., first come first served fashion.
- When a job completes its execution, its memory is released and the output for the job gets copied into an output spool for later printing or processing.

UNIT 4:Operating System - Processes

A process is basically a program in execution. The execution of a process must progress in a sequential fashion.A process is defined as an entity which represents the basic unit of work to be implemented in the system.To put it in simple terms, we write our computer programs in a text file and when we execute this program, it becomes a process which performs all the tasks mentioned in the program.

When a program is loaded into the memory and it becomes a process, it can be divided into four sections ─ stack, heap, text and data. The following image shows a simplified layout of a process inside main memory −

Program: A program is a piece of code which may be a single line or millions of lines. A computer program is usually written by a computer programmer in a programming language. For example, here is a simple program written in C programming language −

```
#include <stdio.h>

int main() {
   printf("Hello, World! \n");
   return 0;
}
```

A computer program is a collection of instructions that performs a specific task when executed by a computer. When we compare a program with a process, we can conclude that a process is a dynamic instance of a computer program.A part of a computer program that performs a well-defined task is known as an **algorithm**. A collection of computer programs, libraries and related data are referred to as a **software**.

Process Life Cycle

When a process executes, it passes through different states. These stages may differ in different operating systems, and the names of these states are also not standardized.

B.L.D.E. Association's

S.B.ARTS & K.C.P SCIENCE COLLEGE
VIJAYAPUR

M.Sc(CS) Programme

Basic computer organisation and design

Computer organization refers to the operational unit and their interconnection that realise the architectural specification. Computer organization deals with how different part of a computer are organised and how various operations are performed between different part to do a specific task.The organization of the computer is defined by its internal registers ,timing and control structure ,and set of instruction that is uses.

Instruction & Instruction Codes

Instruction:-

Computer instructions are a set of machine language instructions that a particular processor understands and executes. A computer performs tasks on the basis of the instruction provided.
An instruction comprises of groups called fields. These fields include:

- The Operation code (Opcode) field which specifies the operation to be performed.
- The Address field which contains the location of the operand, i.e., register or memory location.
- The Mode field which specifies how the operand will be located.

A basic computer has three instruction code formats which are:

1. Memory – reference instruction
2. Register – reference instruction
3. Input-Output instruction

Memory – reference instruction

In Memory-reference instruction, 12 bits of memory is used to specify an address and one bit to specify the addressing mode 'I'.

Register – reference instruction

The Register-reference instructions are represented by the Opcode 111 with a 0 in the leftmost bit (bit 15) of the instruction.
A Register-reference instruction specifies an operation on or a test of the AC (Accumulator) register.

Input-Output instruction

Just like the Register-reference instruction, an Input-Output instruction does not need a reference to memory and is recognized by the operation code 111 with a 1 in the leftmost bit of the instruction. The remaining 12 bits are used to specify the type of the input-output operation or test performed.

- The three operation code bits in positions 12 through 14 should be equal to 111. Otherwise, the instruction is a memory-reference type, and the bit in position 15 is taken as the addressing mode I.

- When the three operation code bits are equal to 111, control unit inspects the bit in position 15. If the bit is 0, the instruction is a register-reference type. Otherwise, the instruction is an input-output type having bit 1 at position 15.

Instruction Set Completeness

A set of instructions is said to be complete if the computer includes a sufficient number of instructions in each of the following categories:

- Arithmetic, logical and shift instructions
- A set of instructions for moving information to and from memory and processor registers.
- Instructions which controls the program together with instructions that check status conditions.
- Input and Output instructions

Arithmetic, logic and shift instructions provide computational capabilities for processing the type of data the user may wish to employ.
A huge amount of binary information is stored in the memory unit, but all computations are done in processor registers. Therefore, one must possess the capability of moving information between these two units.
Program control instructions such as branch instructions are used change the sequence in which the program is executed.
Input and Output instructions act as an interface between the computer and the user. Programs and data must be transferred into memory, and the results of computations must be transferred back to the user.

Instruction Cycle

A program residing in the memory unit of a computer consists of a sequence of instructions. These instructions are executed by the processor by going through a cycle for each instruction.

In a basic computer, each instruction cycle consists of the following phases:

1. Fetch instruction from memory.
2. Decode the instruction.
3. Read the effective address from memory.
4. Execute the instruction.

Input-Output Configuration

In computer architecture, input-output devices act as an interface between the machine and the user.Instructions and data stored in the memory must come from some input device. The results are displayed to the user through some output device.The following block diagram shows the input-output configuration for a basic computer.

- The input-output terminals send and receive information.
- The amount of information transferred will always have eight bits of an alphanumeric code.
- The information generated through the keyboard is shifted into an input register ‘INPR’.
- The information for the printer is stored in the output register ‘OUTR’.

- Registers INPR and OUTR communicate with a communication interface serially and with the AC in parallel.
- The transmitter interface receives information from the keyboard and transmits it to INPR.
- The receiver interface receives information from OUTR and sends it to the printer serially.

Design of a Basic Computer

A basic computer consists of the following hardware components.

1. A memory unit with 4096 words of 16 bits each
2. Registers: AC (Accumulator), DR (Data register), AR (Address register), IR (Instruction register), PC (Program counter), TR (Temporary register), SC (Sequence Counter), INPR (Input register), and OUTR (Output register).
3. Flip-Flops: I, S, E, R, IEN, FGI and FGO

Linux

Linux is one of popular version of UNIX operating System. It is open source as its source code is freely available. It is free to use. Linux was designed considering UNIX compatibility. Its functionality list is quite similar to that of UNIX.

Components of Linux System

Linux Operating System has primarily three components

- **Kernel** − Kernel is the core part of Linux. It is responsible for all major activities of this operating system. It consists of various modules and it interacts directly with the underlying hardware. Kernel provides the required abstraction to hide low level hardware details to system or application programs.
- **System Library** − System libraries are special functions or programs using which application programs or system utilities accesses Kernel's features. These libraries implement most of the functionalities of the operating system and do not requires kernel module's code access rights.
- **System Utility** − System Utility programs are responsible to do specialized, individual level tasks.

Basic Features

Following are some of the important features of Linux Operating System.

- **Portable** − Portability means software can works on different types of hardware in same way. Linux kernel and application programs supports their installation on any kind of hardware platform.
- **Open Source** − Linux source code is freely available and it is community based development project. Multiple teams work in collaboration to enhance the capability of Linux operating system and it is continuously evolving.

- **Multi-User** − Linux is a multiuser system means multiple users can access system resources like memory/ ram/ application programs at same time.
- **Multiprogramming** − Linux is a multiprogramming system means multiple applications can run at same time.
- **Hierarchical File System** − Linux provides a standard file structure in which system files/ user files are arranged.
- **Shell** − Linux provides a special interpreter program which can be used to execute commands of the operating system. It can be used to do various types of operations, call application programs. etc.
- **Security** − Linux provides user security using authentication features like password protection/ controlled access to specific files/ encryption of data.

Architecture

The architecture of a Linux System consists of the following layers −

- **Hardware layer** − Hardware consists of all peripheral devices (RAM/ HDD/ CPU etc).
- **Kernel** − It is the core component of Operating System, interacts directly with hardware, provides low level services to upper layer components.
- **Shell** − An interface to kernel, hiding complexity of kernel's functions from users. The shell takes commands from the user and executes kernel's functions.
- **Utilities** − Utility programs that provide the user most of the functionalities of an operating systems.

B.L.D.E. Association's
S.B.ARTS & K.C.P SCIENCE COLLEGE
VIJAYAPUR
M.Sc(CS) Programme

Syllabus
Mapping of Computer Hardware with logic gates

UNIT 1: Processor Architecture: Introduction to Processor, architecture and applications.
UNIT 2: Components: components-ALU, Coprocessor, clock signal, synchronzation of modules.
UNIT 3: Functionality: Multiplexer, Demultiplexer, Address bus and Data bus, DMA.
UNIT 4: Mapping: significance of logic gate in design of memory and processor.

Content

UNIT 1:

History, processor: 4-bit, 8-bit, 16-bit 32 bit and 64-bit, Heat sink, Intel and AMD processor comparison.

UNIT 2:

Arithmetic and Logic operations, Math coprocessor, DMA, clock signal,

UNIT 3:

Significance of Multiplexer, Demultiplexer and its applications. Address bus, data bus and multiplexed Address/Data bus. Direct Memory access.

UNIT 4:

The logic gates as building block for - processor, memory, clock, multiplexer, demultiplexer, encoder, decoder, CPU register

Syllabus
Data Structure

Unit 1: Data, Definition, Data types, Built-in Data Type, Derived Data Type, Basic types of Data Structures, Algorithm, Complexity
Unit 2: Basic Operations, Classification Data Structures, Arrays Linked List
Unit 3: Stack
Unit 4: Queue, Trees, Heap, Dictionary, Graph
Unit 5: Real life applications of data structures

Content

Data Definition defines a particular data with the following characteristics.

- Atomic: Definition should define single concept
- Traceable: Definition should be able to mapped to some data element.
- Accurate: Definition should be unambiguous.
- Clear and Concise: Definition should be understandable.

Data Object

Data Object represents an object having a data.

Data Type

Data type is a way to classify various types of data such as integer, string, etc. which determines the values that can be used with the corresponding type of data, the type of operations that can be performed on the corresponding type of data. There are two Data type_

- Built-in Data Type
- Derived Data Type

Built-in Data Type

Those data types for which a language has built-in support are known as Built-in Data types. For example, most of the languages provide the following built-in data types.

- Integers
- Boolean (true, false)
- Floating (Decimal numbers)
- Character and Strings

Derived Data Type

Those data types which are implementation independent as they can be implemented in one or the other way are known as derived data types. These data types are normally built by the combination of primary or built-in data type and associated operations on them. For example
List, Array, Stack and Queue.

Basic Operations

The data in the data structures are processed by certain operations. The particular data structure chosen largely depends on the frequency of the operation that needs to be performed on the data structure.

- Traversing
- Searching
- Insertion
- Deletion
- Sorting
- Merging

Before introducing data structures we should understand that computers do store, retrieve, and process a large amount of data. If the data is stored in well organized way on storage media and in computer's memory then it can be accessed quickly for processing that further reduces the latency and the user is provided fast response.

Classification Data Structures

Data structures can be broadly classified in two categories - *linear structures* and *hierarchical structures*. Arrays, linked lists, stacks, and queues are linear structures, while trees, graphs, heaps etc. are hierarchical structures.

Arrays

Linked List

Stack
Queue
Trees
Heap
Dictionary
Hash Table
Graph

Introduction to Data Structures

Data Structure is a way of collecting and organising data in such a way that we can perform operations on these data in an effective way. Data Structures is about rendering data elements in terms of some relationship, for better organization and storage. For example, we have data player's name "Virat" and age 26. Here "Virat" is of **String** data type and 26 is of **integer** data type.

Basic types of Data Structures

As we discussed above, anything that can store data can be called as a data strucure, hence Integer, Float, Boolean, Char etc, all are data structures. They are known as Primitive Data Structures.

- Linked List
- Tree
- Graph
- Stack, Queue etc.

R. D. Joshi
Co-ordinator
M.Sc. (C.S.) Programme
S.B.Arts & K.C.P.Science College,
Vijayapur.

IQAC, Co-ordinator
S.B.Arts & K.C.P.Science College,
Vijayapur.

Principal
S.B.Arts & K.C.P.Science College,
Vijayapur.