



**Mahatma Gandhi Mission**

## **Jawaharlal Nehru Engineering College**

**Aurangabad, Maharashtra**

Affiliated to Dr. B. A. Technological University, Lonere

NAAC 'A' Grade, ISO 9001:2015, 14001:2015 Certified, AICTE Approved.

**Second Year B. Tech**

**Department of Information Technology**

**Lab Book**

**BTITL307: Object Oriented Paradigm with C++**

**Name:** \_\_\_\_\_

**Class:** \_\_\_\_\_ **Roll No:** \_\_\_\_\_ **Year:** \_\_\_\_\_

**Exam No.:** \_\_\_\_\_



**Mahatma Gandhi Mission**

## **Jawaharlal Nehru Engineering College**

**Aurangabad, Maharashtra**

Affiliated to Dr. B. A. Technological University, Lonere

NAAC 'A' Grade, ISO 9001:2015, 14001:2015 Certified, AICTE Approved.

**Second Year B. Tech**

**Department of Information Technology**

**Lab Book**

**BTITL307: Object Oriented Paradigm with C++**

Prepared by  
**Prof. P.Kaur**  
Assistant Professor  
Lab Incharge

Reviewed by  
**Dr. S. C. Tamane**  
Professor  
Head of Department

Approved by  
**Dr. H. H. Shinde**  
Principal

### **Vision of Information Technology Department:**

To develop expertise of budding technocrats by imparting technical knowledge and human value based education.

### **Mission of Information Technology Department:**

- A. Equipping the students with technical skills, soft skills and professional attitude.
- B. Providing the state of art facilities to the students to excel as competent professionals, entrepreneurs and researchers.

### **Programme Educational Objectives:**

- PEO1. The graduates will utilize their **expertise** in IT industry and solve industry technological problems.
- PEO2. Graduates should excel in **engineering positions** in industry and other organizations that emphasize design & implementation of IT applications.
- PEO3. Graduates will be **innovators & professionals** in technology development, deployment & system implementation.
- PEO4. Graduates will be pioneers in engineering, engineering management, research and **higher education**.
- PEO5. Graduates will be good citizens & cultured human being with full appreciation of importance of IT **professional ethical & social** responsibilities.

### **Program specific outcomes**

- PSO1. An ability to design, develop and implement computer programs in the areas related to Algorithms, Multimedia, Website Design, System Software, DBMS and Networking.
- PSO2. Develop software systems that would perform tasks related to Research, Education and Training and/or E governance.
- PSO3. Design, develop, test and maintain application software that would perform tasks related to information management and mobiles by utilizing new technologies to an individual or organizations.

**Lab outcomes:** After the completion of this course students will be able to,

LO1: Set up the environment to write, compile and execute C++ programs.

LO2: Learn and develop structures to represent objects and the methods to perform operations for a problem.

LO3: Apply object-oriented programming paradigms (standards and principles) to write program for a given problem.

LO4: Understand and demonstrate usage of the object-oriented concepts like inheritance, polymorphism, overloading, files etc.

LO5: Develop debug and correct the errors of a program to produce desired output.

**Mandatory instructions for students:**

1. Students should report to the concerned labs as per the given timetable.
2. Students should make an entry in the log book whenever they enter the labs during practical or for their own personal work.
3. When the experiment is completed, students should shut down the computers and make the counter entry in the logbook.
4. Any damage to the lab computers will be viewed seriously.
5. Students should not leave the lab without concerned faculty's permission.



Mahatma Gandhi Mission

## Jawaharlal Nehru Engineering College

N-6, CIDCO Aurangabad – 431003

### INDEX OF EXPERIMENTS

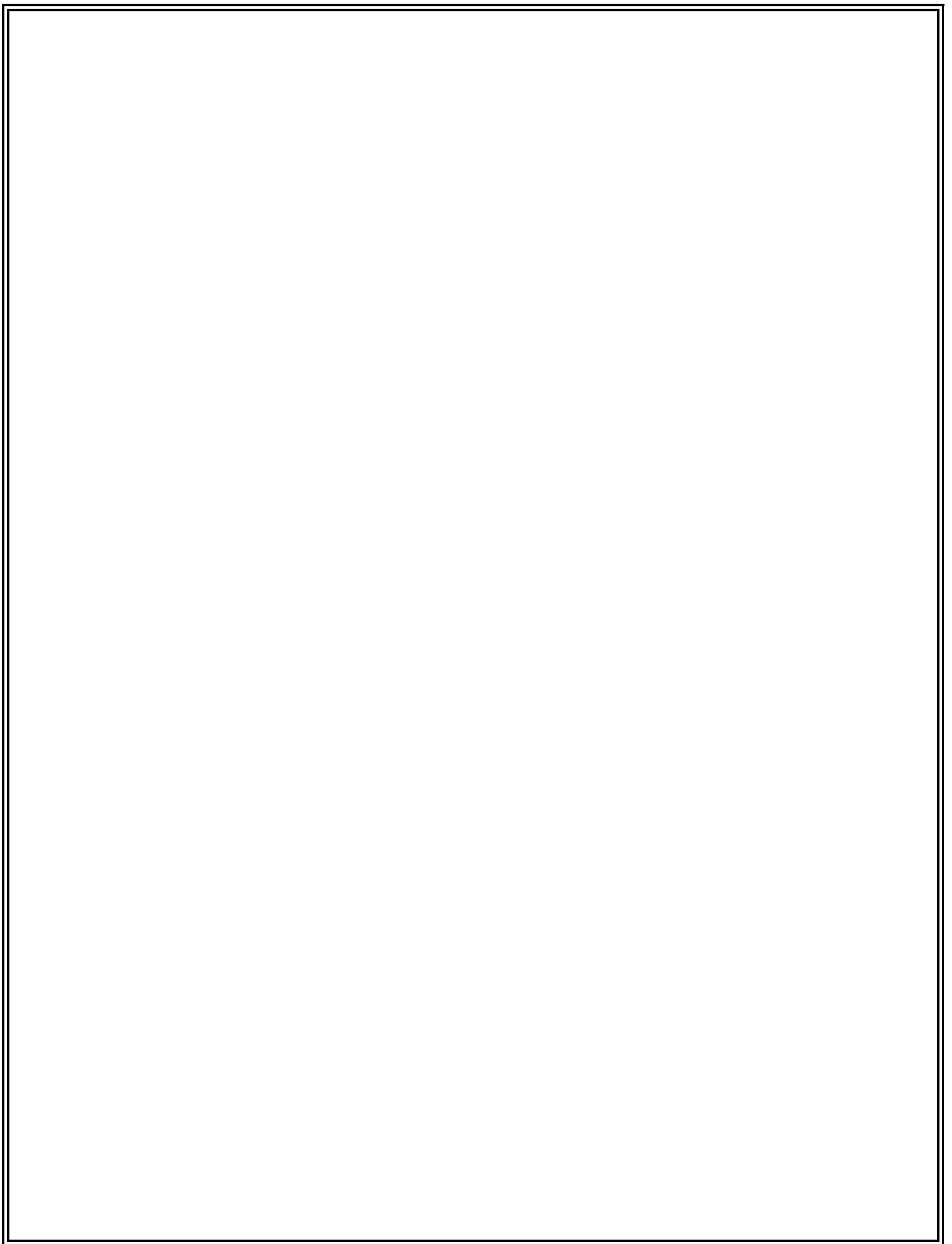
Name of student: \_\_\_\_\_

Roll No: \_\_\_\_\_ PNR No: \_\_\_\_\_

Batch: \_\_\_\_\_

Sr. No.	Practical details	Date of performance	Date of evaluation	Marks/Grade	Sign

Date: \_\_\_\_\_ Lecturer in charge \_\_\_\_\_ Head of Dept. \_\_\_\_\_ Principal \_\_\_\_\_



## Exercise 1: Elements of computer systems

The internal architectural design of computers differs from one system model to another. However, the basic organization remains the same for all computer systems. In order for a computer system to operate, it requires a few key hardware components, the most important of these being the Central Processing Unit (CPU) and the motherboard. The motherboard acts as the central "hub" on which all of the computer's various hardware peripherals, ranging from sound and video cards to hard drives, connect. The primary power source for the computer connects to the motherboard, which then distributes power to the hardware connected to it.

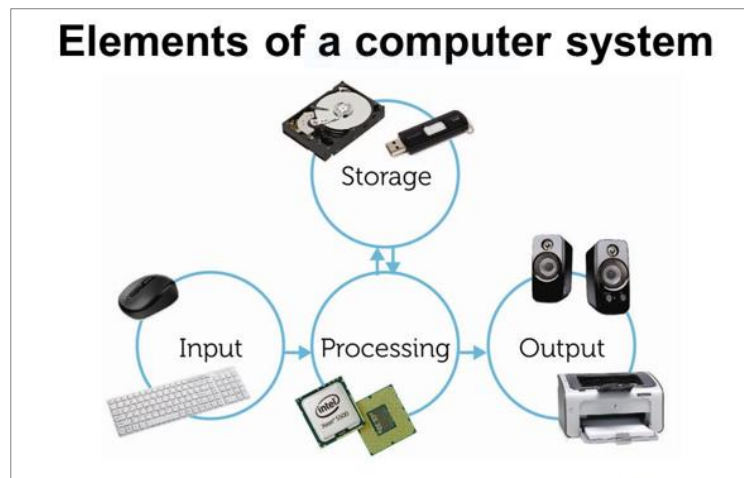


Figure 1.1 Elements of a Computer System

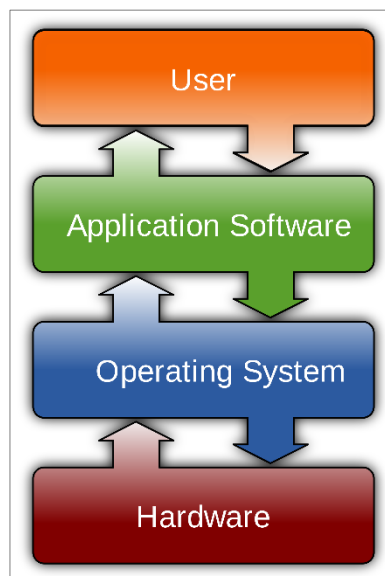


Figure 1.2 Connectivity between elements of a Computer System

## **Practice Exercise 1**

Based on understanding of basic components of a computer system, answer the following questions

The following entities or devices are part of a computer system's hardware (H) or Software (S)?

- i. Operating system \_\_\_\_\_
- ii. Mouse \_\_\_\_\_
- iii. CPU \_\_\_\_\_
- iv. Keyboard \_\_\_\_\_
- v. Compiler \_\_\_\_\_
- vi. Program editor \_\_\_\_\_
- vii. IDE \_\_\_\_\_
- viii. GUI (Graphical User Interface) \_\_\_\_\_
- ix. System Clock \_\_\_\_\_
- x. Modem \_\_\_\_\_
- xi. Assembler \_\_\_\_\_
- xii. Bus \_\_\_\_\_
- xiii. RAM \_\_\_\_\_
- xiv. ROM \_\_\_\_\_
- xv. File \_\_\_\_\_

1. Tally, the computerized accounting software, is which of the following?

- a) Operating System
- b) Application Software
- c) Utility Software
- d) None of the above

Answer:

2. Term 'program' is refers to any piece of

- a) Software
- b) Hardware
- c) Netware
- d) Groupware

Answer:

3. In computer science, algorithm refers to a pictorial representation of a flowchart.

- a) True
- b) False



Answer:

4. The process of drawing a flowchart for an algorithm is called \_\_\_\_\_
- a) Performance
  - b) Evaluation
  - c) Algorithmic Representation
  - d) Flowcharting

Answer:

5. In flow chart, diamond shaped symbol is used to represent
- a) Decision box
  - b) Statement box
  - c) Error box
  - d) if-statement box
6. Part of algorithm which is repeated for fixed number of times is classified as
- a) iteration
  - b) selection
  - c) sequence
  - d) reverse action

**Grade:**

**Signature of teacher:**

## Exercise 2: DOS and Linux commands

### DOS Commands

DOS (Disk Operating System) is an operating system that runs from a hard disk drive. The term can also refer to a particular family of disk operating systems, most commonly MS-DOS (Microsoft Disk Operating System).

DOS commands are small programs, which are made to perform a particular job. Every DOS command performs different task. It is not possible to work on the computer without these commands. There are two types of DOS command.

**Internal Commands:** These commands enter into the computer memory during computer booting. These commands are not in the form of any file; so neither they can be viewed nor can be edited or detected.

For example: MD, CD, TIME, DATE, COPY, TYPE etc.

**External Commands:** These commands are stored in the computer list in the form of files. These Commands can be viewed, copied, changed or deleted.

For example: FORMAT, COPY, PRINT, SYS, EDIT, TREE, SORT, PROMPT etc.

DOS commands are the commands available in MS-DOS that are used to interact with the operating system and other command line based software.

*Unlike* in Windows, DOS commands are the primary way in which you use the operating system. Windows and other modern OSs use a graphics-based system designed for touch or a mouse.

**DOS Commands in Windows:** If you use Windows (like Windows 10, 8, 7, etc.) then you have no need for DOS commands *because you don't have MS-DOS*. The commands in Windows are available from the Command Prompt and are called Command Prompt commands or CMD commands, but they are *not* DOS commands.

### Linux Commands

Linux is a Unix-Like operating system. All the Linux/Unix commands are run in the terminal provided by the Linux system. This terminal is just like command prompt of Windows OS. Linux/Unix commands are *case-sensitive*. The terminal can be used to accomplish all Administrative tasks. This includes package installation, file manipulation, and user management. Linux terminal is user-interactive. The terminal outputs the results of commands which are specified by the user itself. Execution of typed command is done only after you press the Enter key.

**Table 2.1 shows few DOS and corresponding Linux commands**

<b>Command's Purpose</b>	<b>MS-DOS</b>	<b>Linux</b>	<b>Basic Linux Example</b>
<b>Copies files</b>	copy	cp	cp <i>thisfile.txt</i> /home/ <i>thisdirectory</i>
<b>Moves files</b>	move	mv	mv <i>thisfile.txt</i> /home/ <i>thisdirectory</i>
<b>Lists files</b>	dir	ls	ls
<b>Clears screen</b>	cls	clear	clear
<b>Closes shell prompt</b>	exit	exit	exit
<b>Displays or sets date</b>	date	date	date
<b>Deletes files</b>	del	rm	rm <i>thisfile.txt</i>
<b>"Echoes" output to the screen</b>	echo	echo	echo <i>this message</i>
<b>Edits text files</b>	edit	gedit	gedit <i>thisfile.txt</i>
<b>Compares the contents of files</b>	fc	diff	diff <i>file1 file2</i>
<b>Finds a string of text in a file</b>	find	grep	grep <i>word or phrase thisfile.txt</i>
<b>Formats a diskette</b>	format E:	mke2fs	/sbin/mke2fs /dev/fd0 (/dev/fd0 is the Linux equivalent of A:)
<b>Displays command help</b>	<i>command</i> and /?	man or info	man <i>command</i>
<b>Creates a directory</b>	mkdir	mkdir	mkdir <i>directory</i>
<b>Views contents of a file</b>	more	less	less <i>thisfile.txt</i>
<b>Renames a file</b>	ren	mv	mv <i>thisfile.txt thatfile.txt</i>
<b>Displays your location in the file system</b>	chdir	pwd	pwd
<b>Changes directories with a specified path (<i>absolute path</i>)</b>	cd <i>path</i> <i>hname</i>	cd <i>pathname</i>	cd / <i>directory/directory</i>
<b>Changes directories with a <i>relative path</i></b>	cd..	cd ..	cd ..
<b>Displays the time</b>	time	date	date
<b>Shows amount of RAM in use</b>	mem	free	free

## **Practice Exercise 2**

1. Core of Linux operating system is
  - a) Kernel
  - b) Shell
  - c) Terminal

d) Command

Answer:

2. Which directory contains configuration files in Linux?

a) /etc/

b) /bin/

c) /dev/

d) /root/

Answer:

3. Which shell is the default shell used on Linux systems?

a) Csh

b) Rsh

c) Bash

d) Tcsh

Answer:

4. In MS-DOS, which command is used to display system date

a) Date command

b) Ver command

c) Disk command

d) Format command

Answer:

5. If you need to duplicate the entire disk, which command will you use?

a) Copy

b) Diskcopy

c) Chkdsk

d) Format

Answer:

**Grade:**

**Signature of teacher:**

## **Exercise 3: Setting up C++ Development Environment**

C++ is a general purpose programming language. It has object-oriented and generic programming features. C++ runs on variety of platform like Windows, Linux, Unix, Mac etc. To write and execute programs in C++, an environment needs to be set-up on a computer to compile and run C++ programs. Another option is to use an online IDEs for compiling C++ programs.

**Online IDE:** IDE is Integrated Development Environment. It is a software application that can be used for developing software. Many online IDEs are available which can be used to compile and run C++ programs.

**How to setup Local Environment:** To setup a local development environment, install two software:

1. **Text Editor:** Text Editors are the programs used to edit or write programs. For writing C++ programs, we will use text-editors. The usual extension of a text file is (.txt) but for a text file having C++ program needs to be saved with ‘.CPP’ or ‘.C’ extension. These files are source code files. To start writing programs in C++, we should have a text-editor installed to write programs.
2. **C++ Compiler:** After writing a C++ program (‘.CPP’ extension), we require a C++ compiler to compile this file.

A compiler is a computer program which converts high-level language (source-code) into machine understandable low-level language. In other words we can say that it converts the source code written in a programming language into another computer language which computer understands. For compiling a C++ program we will need a C++ compiler which will convert the source code written in C++ into machine codes. Below is the details about setting up compiler on different platforms.

**2.1 For Linux based systems:** We can install the GNU GCC compiler on Linux. To install and work with the GCC compiler on your Linux machine, following steps:

- i. Run the two commands from your Linux terminal window:

- \$ `sudo apt-get update`
- \$ `sudo apt-get install GCC`

This command will install the GCC compiler on your system. Also run the below command to install all essential libraries required to compile and run a C++ program:

- \$ `sudo apt-get install build-essential`

ii. Next step is to confirm installation by checking for GCC version.

- \$ `gcc --version`

It should display GCC version.

iii. Now Linux environment is set up and can be used to compile C++ programs.

iv. Next step is to understand how to compile and run a C++ program.

- Write program using a text editor and save it with any file name having .CPP extension.
- Next step is to open the Linux terminal and change to the directory where .CPP file is saved. Use the following command to compile file:

- \$ `g++ filename.cpp -o any-name`

Here, *filename.cpp* is the name of the source code file and *any-name* is the file name to be assigned to executable file. Executable file is created by compiler post compilation. Run the command as:

- \$ `g++ testfile.cpp -o testfile`

v. The last step creates a new file in the same directory where we have saved the source file and this new file is named as *testfile*.

vi. Next step is to run the program using following command:

- \$ `./testfile`

This command will run the program in the terminal window.

**2.2 For Windows based systems:** A large number of IDE are available for Windows OS to work with C++ programming language. The most popular IDE is **Code::Blocks**. Download Code::Blocks and install.

After successful installation of Code::Blocks, go to *File* menu -> Select *New* and *create an Empty* file.

- i. Write a C++ program in this empty file and save the file with a '.cpp' extension.
- ii. After saving the file with '.cpp' extension, go to *Build* menu and choose the *Build and Run* option.

### **Practice Exercise 3**

Write a C++ program to write and display an essay on you. Write your introduction including family background, strengths, weakness, future plan.

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Grade:**

**Signature of teacher:**

**Exercise 4: Write a C++ program for the given problem description to demonstrate use of function.**

Raising a number  $n$  to a power  $p$  is the same as multiplying  $n$  by itself  $p$  times. Write a function called `power ()` that takes a double value for  $n$  and an int value for  $p$ , and returns the result as double value. Use a default argument of 2 for  $p$ , so that if this argument is omitted, the number will be squared. Write a `main ()` function that gets values from the user to test this function.

**Explanation of problem statement with an example:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Algorithm:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---









**Exercise 5: Write a C++ program for the given problem description.**

A point on the two-dimensional plane can be represented by two numbers: an X coordinate and a Y coordinate. For example, (4,5) represents point 4 units to the right of the origin along the X axis and 5 units up the Y axis. The sum of two points can be defined as a new point whose X coordinate is the sum of the X coordinates of the points and whose Y coordinate is the sum of their Y coordinates. Write a program that uses a structure called point to model a point. Define three points, and have the user input values to two of them. Then set the third point equal to the sum of the other two, and display the value of the new point. Interaction with the program might look like this:

```
Enter coordinates for P1: 3 4
Enter coordinates for P2: 5 7
Coordinates of P1 + P2 are: 8, 11
```

Create the equivalent of a four-function calculator. The program should request the user to enter a number, an operator, and another number. It should then carry out the specified arithmetical operation: adding, subtracting, multiplying, or dividing the two numbers. (It should use a switch statement to select the operation). Finally, it should display the result. When it finishes the calculation, the program should ask if the user wants to do another calculation. The response can be Y or N. Some sample interaction with the program might look like this:

```
Enter first number, operator, second number: 10/ 3
Answer = 3.333333
Do another (Y/ N)? Y
Enter first number, operator, second number 12 + 100
Answer = 112
Do another (Y/ N)? N
```

**Explanation of problem statement with an example:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---







---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Grade:**

**Signature of teacher:**



**Exercise 6: Write a C++ program for the given problem description using friend function.**

A phone number, such as (212) 767-8900, can be thought of as having three parts: the area code (212), the exchange (767) and the number (8900). Write a program that uses a structure to store these three parts of a phone number separately. Call the structure phone. Create two structure variables of type phone. Initialize one, and have the user input a number for the other one. Then display both numbers. The interchange might look like this:

Enter your area code, exchange, and number: 415 555 1212

My number is (212) 767-8900

Your number is (415) 555-1212

Create two classes DM and DB which store the value of distances. DM stores distances in meters and centimetres and DB in feet and inches. Write a program that can read values for the class objects and add one object of DM with another object of DB. Use a friend function to carry out the addition operation. The object that stores the results maybe a DM object or DB object, depending on the units in which the results are required. The display should be in the format of feet and inches or meters and centimetres depending on the object on display.

**Explanation of problem statement with an example:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Algorithm:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Program:**

---

---

---

---

---



---

---

---

---

---

---

---

**Practice Assignment based on above problem description:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Grade:**

**Signature of teacher:**

**Exercise 7: Write a C++ program for the given problem description to demonstrate operator overloading.**

Create a class rational which represents a numerical value by two double values- NUMERATOR and DENOMINATOR. Include the following public member Functions: constructor with no arguments (de-fault), constructor with two arguments, void reduce() that reduces the rational number by eliminating the highest common factor between the numerator and denominator.

    Overload + operator to add two rational numbers

    Overload - operator to enable input through cin

    Overload \* operator to enable output through cout

Write a main ( ) to test all the functions in the class.

**Explanation of problem statement with an example:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Algorithm:**

---

---

---

---



---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Input:**

---

---

---

**Output:**

---

---

---

---





**Exercise 8: Write a C++ program for the given problem description to demonstrate polymorphism.**

Consider the following class definition:

```
class father {
    protected age;
public;
    father (int x) {age = x;}
    virtual void iam()
    {
        cout<<"I AM THE FATHER " ;
        cout << "My age is : " <<age<< endl;}
};
```

Derive the two classes son and daughter from the above class and for each, define iam() to write similar but appropriate messages. You should also define suitable constructors for these classes. Now, write a main() that creates objects of the three classes and then calls iam( ) for them. Declare pointer to father. Successively, assign addresses of objects of the two derived classes to this pointer and in each case, call iam ( ) through the pointer to demonstrate polymorphism in action.

**Explanation of problem statement with an example:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---



---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Input:**

---

---

---

---

**Output:**

---

---

---

---

---

**Practice Assignment based on above problem description:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Grade:**

**Signature of teacher:**

**Exercise 8: Write a C++ program for the given problem description to demonstrate polymorphism.**

Consider the following class definition:

```
class father {  
    protected age;  
public:  
    father (int x) {age = x;}  
    virtual void iam()  
    {  
        cout<<“I AM THE FATHER ” ;  
        cout << “My age is :” <<age<< endl;}  
};
```

Derive the two classes son and daughter from the above class and for each, define iam() to write similar but appropriate messages. You should also define suitable constructors for these classes. Now, write a main() that creates objects of the three classes and then calls iam( ) for them. Declare pointer to father. Successively, assign addresses of objects of the two derived classes to this pointer and in each case, call iam ( ) through the pointer to demonstrate polymorphism in action.

**Explanation of problem statement with an example:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---









## Exercise 9: Write a C++ program for the given problem description to demonstrate use of files.

Write a program that creates a binary file by reading the data for the students from the terminal. The data of each student consist of roll number, name (a string of 30 or lesser number of characters) and marks.

### Explanation of problem statement with an example:

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

### Algorithm:

---

---

---

---

---

---

---



---

---

---

---

---

---

---

**Input:**

---

---

---

---

---

---

---

**Output:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Practice Assignment based on above problem description:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Grade:**

**Signature of teacher:**

**Exercise 10: Write a C++ program for the given problem description to demonstrate use of structure/database.**

A hospital wants to create a database regarding its indoor patients. The information to store include:

- Name of the patient
- Date of admission
- Disease
- Date of discharge

Create a structure to store the date (year, month and date as its members). Create a base class to store the above information. The member function should include functions to enter information and display a list of all the patients in the database. Create a derived class to store the age of the patients. List the information about all the patients to store the age of the patients. List the information about all the paediatric patients (less than twelve years in age).

**Explanation of problem statement with an example:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Algorithm:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Program:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Input:**

---

---

---

---

---

**Output:**

---

---

---

---

---

---

**Practice Assignment based on above problem description:**

---

---

---

---

---

---

---

---

---

---



---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Grade:**

**Signature of teacher:**

**Exercise 11: Write a C++ program for the given problem description.**

Imagine a tollbooth with a class called toll Booth. The two data items are a type unsigned int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both these to 0. A member function called payingCar( ) increments the car total and adds 0.50 to the cash total. Another function called nopayCar( ), increments the car total but adds nothing to the cash total. Finally, a member function called display() displays the two totals i.e. total cars and total cash. Include a program to test this class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car. Pushing the ESC key should cause the program to print out the total cars and total cash and then exit.

**Explanation of problem statement with an example:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Algorithm:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Program:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

**Input:**

---

---

---

---

---

---

---

---

**Output:**

---

---

---

---

---

---

---

---

---

**Practice Assignment based on above problem description:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---



## **Appendix A**

### **Program Outcomes**

#### **Engineering Graduates will be able to:**

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

### **Program Specific Outcomes**

- PSO1. An ability to design, develop and implement computer programs in the areas related to Algorithms, Multimedia, Website Design, System Software, DBMS and Networking.
- PSO2. Develop software systems that would perform tasks related to Research, Education and Training and/or E governance.
- PSO3. Design, develop, test and maintain application software that would perform tasks related to information management and mobiles by utilizing new technologies to an individual or organizations.