VOLUME-II
Chapter-9
INTRODUCTION TO COMPUTERS

Important 2 & 3 Marks

1. What is Hybrid Language?

C++ supports both procedural and Object Oriented Programming paradigms. Thus, C++ is called as a **hybrid language.**

2. Write about C++.

- C++ is one of the most popular programming language developed by Bjarne Stroustrup at AT & T Bell Lab during 1979.
- C++ is a superset (extension) of its predecessor C language. Bjarne Stroustrup named his new language as "C with Classes". The name C++ was coined by Rick Mascitti where ++ is the C language increment operator.

4. What is Character set?

Character set is a set of characters which are allowed to write a C++ program. A character represents any alphabet, number or any other symbol (special characters) mostly available in the keyboard. C++ accepts the following characters.

Alphabets	A …. Z, a ….. z
Numeric	0 …. 9
Special Characters	+ - * / ~ ! @ # $ % ^& [ ] ( ) { } = >< _ \ | ? . , : ‘ “ ;
White space	Blank space, Horizontal tab (→), Carriage return (↲), Newline, Form feed
Other characters	C++ can process any of the 256 ASCII characters as data.

5. What is Lexical Units (Tokens)?

- ✓ Tokens are Individual Units in a Program.
- ✓ Basics types of elements are collectively called as tokens
  - ***It can be classified in to 5 categories.***
    1. Keywords
    2. Identifiers
    3. Constants
    4. Operators
    5. Punctuators

Mobile: +91- 9677635246 E-Mail: nmohan707@gmail.com

6. What is mean by Keywords? List few.

- Keywords are the reserved words which convey specific meaning to the C++ compiler.
- ***Keywords : Class, delete, this, new, try, asm, template, throw, friend, volatile***

7. What is mean by Identifiers?

- Identifiers are the user-defined names given to different parts of the C++ program viz. variables, functions, arrays, classes etc., These are the fundamental building blocks of a program.
- Every language has specific rules for naming the identifiers.

8. What are the Rules for naming an identifiers?

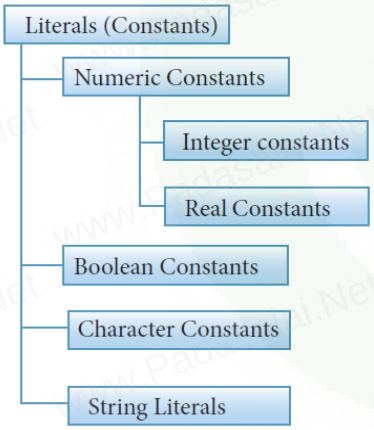
1. The first character of an identifier must be an alphabet or an underscore (_).

2. Only alphabets, digits and underscore are permitted. Other special characters are not allowed as part of an identifier.

3. C++ is case sensitive as it treats upper and lower-case characters differently. Reserved words or keywords cannot be used as an identifier (variable) name.

9. What is Literals (Constants)?

Literals are data items whose values ***do not change*** during the execution of a program. Therefore Literals are called as **Constants.**



10. What is String Literals?

Sequence of characters enclosed within double quotes are called as String literals.

For ex : "A", "Welcome" "1234"

11. What is Operators & Operand?

- ✓ The symbols which are used to do some mathematical or logical operations are called as ***"Operators".***
- ✓ The data items or values that the operators act upon are called as ***"Operands".***

12. How operators can be classified?

In C++, The operators are classified on the basis of the number of operands.

(i) Unary Operators	-	Require only one operand
(ii) Binary Operators	-	Require two operands
(iii) Ternary Operators	-	Require three operands

13. List out C++ Operators.

(1) Arithmetic Operators

(2) Relational Operators

(3) Logical Operators

(4) Bitwise Operators

(5) Assignment Operators

(6) Conditional Operator

(7) Other Operators

14. What is Escape Sequences? Or What is mean by Non-Graphic Characters?

C++ allows certain **non-printable characters** represented as character constants. Non-printable characters are also called as non-graphical characters.

Escape sequence	Non-graphical character
\a	Audible or alert bell
\b	Backspace
\f	Form feed
\n	Newline or linefeed
\r	Carriage return
\t	Horizontal tab
\v	Vertical tab
\\	Backslash
\'	Single quote
\"	Double quote
\?	Question Mark
\On	Octal number
\xHn	Hexadecimal number
\0	Null

15. What are Increment & Decrement Operators?

1. Pre- Increment (++i)
2. Post-Increment (i++)
3. Pre-Decrement (--i)
4. Post-Decrement (i- -)

16. What is Bitwise Operators?

Bitwise operators work on each bit of data and perform bit-by-bit operation. In C++, there are **three** kinds of bitwise operators, which are:

(i) Logical bitwise operators

(ii) Bitwise shift operators

(iii) One's compliment operators

17. What is Separators or Punctuators?

Punctuators are symbols, which are used as delimiters, while constructing a C++ program. They are also called as **"Separators".**

18. What is use of Sizeof Operator?

This is called as compile time operator. It **returns the size** of a variable in bytes.

19. What are the steps to be followed for execution of C++ Program?

For creating and executing a C++ program, one must follow ***four*** important steps.

(1) Creating Source code

(2) Saving source code with extension .cpp

(3) Compilation

(4) execution

20 . What are the structure of C++ Program?

Structure of C++ Program-It consists of 3 parts

1. Include files 2. Declaration of variable & function 3. main( ) function

Important 5 Marks

1. What are the benefits of learning C++?

1. C++ is a highly portable language and is often the language of choice for multi-device, multi-platform app development.
2. C++ is an object-oriented programming language and includes classes, inheritance, polymorphism, data abstraction and encapsulation.
3. C++ has a rich function library.
4. C++ allows exception handling, inheritance and function overloading which are not possible in C.
5. C++ is a powerful, efficient and fast language. It finds a wide range of applications – from GUI applications to 3D graphics for games to real-time mathematical simulations.

2. What are the Types of Errors?

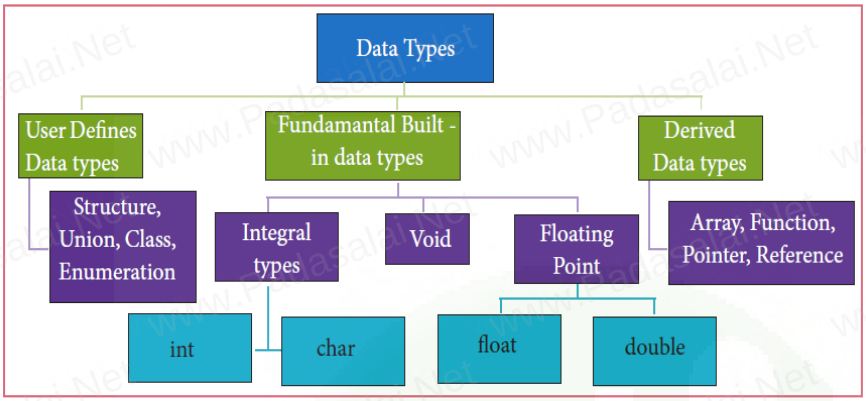
Type of Error	Description
Syntax Error	• Syntax is a set of grammatical rules to construct a program. Every programming language has unique rules for constructing the sourcecode. • Syntax errors occur when grammatical rules of C++ are violated. • Example: if you type as follows, C++ will throw an error. **cout << "Welcome to Programming in C++"** • As per grammatical rules of C++, every executable statement should terminate with a semicolon. But, this statement does not end with a semicolon.
Semantic Error	• A Program has not produced expected result even though the program is grammatically correct.It may be happened by wrong use of variable / operator / order of execution etc. This means, program is grammatically correct, but it contains some logical error. So, Semantic error is also called as **"Logic Error"**.
Run-time error	• A run time error occurs during the execution of a program. It is occurs because of some illegal operation that takes place. • For example, if a program tries to open a file which does not exist, it results in a run-time error

Data Types

1. What are the data types?

In C++, the data types are classified as three main categories

(1) Fundamental data types (2) User-defined data types and (3) Derived data types.



2. What is the use of Void Data Type?

Void type has **two** important purposes:

- ✓ To indicate the function does not return a value
- ✓ To declare a generic pointer.

3. Write the Syntax for Variable Declaration.

Syntax for declaring a variable:

<data type> <variable name>;

Example:

int num1;

4. What is use of *Const* keyword?

Literals are data items whose values ***do not change*** during the execution of a program. Therefore Literals are called as **Constants.**

For ex: const int num=100;

5. What is Reference Variable?

A reference provides an alias for a previously defined variable. Declaration of a reference consists of base type and an & (ampersand) symbol; reference variable name is assigned the value of a previously declared variable.

Syntax:

<type> <& reference_variable> = <original_variable>

6. What is the function of endl , \n, setw, set fill, setprecision?

1. endl – Inserts a new line and flushes the buffer (Flush means – clean)

2. '\n' - Inserts only a new line.

3. setw -setw manipulator sets the width of the field assigned for the output. The field width determines the minimum number of characters to be written in output.

Syntax:

setw(number of characters).

4. **setfill( )** -This manipulator is usually used after setw. If the presented value does not entirely fill the given width, then the specified character in the setfill argument is used for filling the empty fields.

Syntax:

setfill (character);

5. setprecision ( )- This is used to display numbers with fractions in specific number of digits.

Syntax:

setprecision (number of digits);

7. What is mean by Expression?

An expression is a combination of operators, constants and variables arranged as per the rules of C++.

(i) Constant Expression

(ii) Integer Expression

(iii) Floating Expression

(iv) Relational Expression

(v) Logical Expression

(vi) Bitwise Expression

(vii) Pointer Expression

8. What are the types of Conversion?

(1) Implicit type conversion

An Implicit type conversion is a conversion performed by the compiler automatically. So, implicit conversion is also called as ***"Automatic conversion".***

(2) Explicit type conversion.

C++ allows explicit conversion of variables or expressions from one data type to another specific data type by the programmer. It is called as **"type casting".**

Chapter-10
FLOW OF CONTROL

Important 2 & 3 Marks

1. What is statement? What are its types?

A computer program is a set of statements or instructions to perform a specific task.

There are two kinds of statements used in C++.

(i) Null statement (ii) Compound statement

2. What are control statements?

Control statements are statements that alter the sequence of flow of instructions.

Selection Statements:

In a program, statements may be executed sequentially, selectively or iteratively. Every programming languages provides statements to support sequence, selection (branching) and iteration.

Sequential statement:

The sequential statement are the statements, that are executed one after another only once from top to bottom.

Iterative statement:

The ***iteration statement*** is a set of statement are repetitively executed depends upon a conditions.

As soon as the condition becomes false, the repetition stops. This is also known as ***looping statement or iteration statement.***

4. Write the definition & syntax for the following

1. if

The if statement evaluates a condition, if the condition is true then a true-block (a statement or set of statements) is executed, otherwise the true-block is skipped.

The general syntax of the if statement is:

Syntax:

```
if (expression)
        true-block;
statement-x;
```

2. if..else

In the above examples of if, you have seen so for allow you to execute a set of statement is a condition evaluates to true. What if there is another course of action to be followed if the condition evaluates to false. There is another form of if that allows for this kind of either or condition by providing an else clause.

The syntax of the if-else statement is given below:

```
if ( expression)
{
True-block;
}
else
{
False-block;
}
Statement-x
```

3. Nested if:

An if statement contains another if statement is called nested if. The nested can have one of the following three forms.

1. If nested inside if part
2. If nested inside else part
3. If nested inside both if part and else part

Syntax:

```
if (expression-1)
        {
        if (expression)
        {
                True_Part_Statements;
        }
        else
        {
                False_Part_Statements;
        }
        }
  else
          body of else part;
```

4. Switch

The switch statement is a multi-way branch statement. It provides an easy way to dispatch execution to different parts of code based on the value of the expression. The switch statement replaces multiple if-else sequence.

The syntax of the switch statement is;

```
switch(expression)
{
case constant 1:
statement(s);
break;
case constant 2:
statement(s);
break;
.
default:
statement(s);
}
```

5. do… while

Syntax:

The do-while loop is an exit-controlled loop. In do-while loop, the condition is evaluated at the bottom of the loop after executing the body of the loop. This means that the body of the loop is executed at least once, even when the condition evaluates false during the first iteration.

The do-while loop syntax is:

```
do
{
Body of the loop;
} while(condition);
```

6. while

A while loop is a control flow statement that allows the loop statements to be executed as long as the condition is true. The while loop is an entry-controlled loop because the test-expression is evaluated before the entering into a loop.

The while loop syntax is:

```
while ( Test expression )
{
Body of the loop;
}
Statement-x;
```

7. for

The for loop is the easiest looping statement which allows code to be executed repeatedly. It contains three different statements (initialization, condition or test-expression and update expression(s)) separated by semicolons.

Syntax:

```
for (initialization(s); test-expression; update expression(s))
{
Statement 1;
Statement 2;
…….
}
Statement-x;
```

5. Explain ?: operator / What is conditional Operator? / What is Ternary Operator? Explain the the operator, Which is Alternative to if ..else?

The conditional operator (or Ternary operator) is an alternative for 'if else statement'. The conditional operator that consists of two symbols (?:). It takes three arguments. The control flow of conditional operator is shown below

The syntax of the conditional operator is:

expression 1? expression 2 : expression 3

7. What are the parts of the loop?

Every loop has four elements that are used for different purposes. These elements are

1. Initialization expression
2. Test expression
3. Update expression
4. The body of the loop

8. What is nesting of loop?

A loop which contains another loop is called as a nested loop.

The syntax is given below:

```
for (initialization(s); test-expression; update expression(s))
{
for (initialization(s); test-expression; update expression(s)
{
statement(s);
}
statement(s);
}
```

9. What is the use of jump statements? What are its types?

Jump statements are used to interrupt the normal flow of program. Types of Jump Statements are

1. goto statement

The goto statement is a control statement which is used to transfer the control from one place to another place without any condition in a program.

2. break statement

A break statement is a jump statement which terminates the execution of loop and the control is transferred to resume normal execution after the body of the loop.

3. continue statement

The continue statement works quite similar to the break statement. Instead of terminating the loop (break statement), continue statement forces the loop to continue or execute the next iteration.

10. What are the difference between break and continue?

Break	Continue
Break is used to **terminate** the execution of the loop.	Continue is not used to terminate the execution of loop.
It **breaks** the iteration.	It **skips** the iteration.
When this statement is executed, control will come out from the loop and executes the statement immediate after loop.	When this statement is executed, it will not come out of the loop but moves/jumps to the next iteration of loop.
Break is used with loops as well as switch case.	Continue is only used in loops, it is not used in switch case.

11. What is if -else-if ladder

The if-else ladder is a multi-path decision making statement. In this type of statement 'if' is followed by one or more else if statements and finally end with an else statement.

Chapter-11
FUNCTIONS

Important 2 & 3 Marks

1. What is User defined functions & Built in functions?

The functions which are available by default known as "***Built-in***" functions and user can create their own functions known as ***"User-defined"*** functions.

- ✓ User-defined functions – Functions created by users.
- ✓ Built-in functions – Functions which are available in C++ language standard library.

2. What is function & What are the needs for functions?

A large program can typically be split into small sub-programs (blocks) called as functions where each sub-program can perform some specific functionality. Functions reduce the size and complexity of a program, makes it easier to understand, test, and check for errors.

1. Divide and Conquer

2. Reusability

3. What are the types of functions?

Functions can be classified into two types,

1. Pre-defined or Built-in or Library Functions
2. User-defined Function.

4. Write short note on: *stdio*

1. Standard input/output (stdio.h)

This header file defines the standard I/O predefined functions getchar(), putchar(), gets(), puts() and etc.

2. getchar() and putchar() functions

The predefined function getchar() is used to get a single character from keyboard and putchar() function is used to display it.

3. gets() and puts() functions

Function gets() reads a string from standard input and stores it into the string pointed by the variable. Function puts() prints the string read by gets() function in a newline.

5. Write short note on: *Character functions (ctype.h)*

This header file defines various operations on characters. Following are the various character functions available in C++. The header file ctype.h is to be included to use these functions in a program.

Mobile: +91- 9677635246 E-Mail: nmohan707@gmail.com

Character Function	Descriptions
1. isalnum()	This function is used to check whether a character is alphanumeric or not. This function returns non-zero value if c is a digit or a letter, else it returns 0. ***Syntax:*** int isalnum (char c)
2. isalpha()	The isalpha() function is used to check whether the given character is an alphabet or not. ***Syntax:*** int isalpha(char c);
3. isdigit()	This function is used to check whether a given character is a digit or not. This function will return 1 if the given character is a digit, and 0 otherwise. ***Syntax:*** int isdigit(char c)₹
4. islower()	This function is used to check whether a character is in lower case (small letter) or not. This functions will return a non-zero value, if the given character is a lower case alphabet, and 0 otherwise. ***Syntax:*** int islower(char c);
5. isupper()	This function is used to check the given character is uppercase. This function will return 1 if true otherwise 0. For the following examples value 1 will be assigned to n and 0 for m. int n=isupper('A'); **int m=isupper('a');**
7. toupper()	This function is used to convert the given character into its uppercase. This function will return the upper case equivalent of the given character. If the given character itself is in upper case, the output will be the same. **Syntax:** char toupper(char c);
8. tolower()	This function is used to convert the given character into its lowercase. This function will return the lower case equivalent of the given character. If the given character itself is in lower case, the output will be the same. **Syntax:** char tolower(char c);

6. Write short note on: ***String manipulation (string.h)***

String manipulation (string.h) Functions	***Description***
1. strcpy()	The ***strcpy() function takes*** two arguments: target and source. It copies the character string pointed by the source to the memory location pointed by the target. The null terminating character (\0) is also copied.
2. strlen()	The ***strlen()*** takes a null terminated byte string source as its argument and returns its length. The length does not include the null(\0) character.
3. strcmp()	The ***strcmp()*** function takes two arguments: string1 and string2. It compares the contents of string1 and string2 lexicographically.
4. strcat()	The ***strcat()*** function takes two arguments: target and source. This function appends copy of the character string pointed by the source to the end of string pointed by the target.
5. strupr()	The ***strupr()*** function is used to convert the given string into Uppercase letters.
6. strlwr()	The **strlwr()** function is used to convert the given string into Lowercase letters.

7. Write short note on:

Mathematical functions (math.h)

Most of the mathematical functions are defined in math.h header file which includes basic mathematical functions.

Mathematical functions	***Description***
1. cos() function	The cos() function takes a single argument in radians. The cos() function returns the value in the range of [-1, 1]. The returned value is either in double, float, or long double.
2. sqrt() function	The sqrt() function returns the square root of the given value of the argument. The sqrt() function takes a single non-negative argument. If a negative

	value is passed as an argument to sqrt() function, a domain error occurs.
3. sin() function	The sin() function takes a single argument in radians. The sin() function returns the value in the range of [-1, 1]. The returned value is either in double, float, or long double.
4. pow() function	The pow() function returns base raised to the power of exponent. If any argument passed to pow() is long double, the return type is promoted to long double. If not, the return type is double. The pow() function takes two arguments: base - the base value exponent - exponent of the base

8. Write the syntax for function Declaration & Defiinition.

In C++, a function must be defined before it is used anywhere in the program. The general syntax of a function definition is:

```
Return_Data_Type Function_name(parameter list)
{
    Body of the function ;
}
```

9. Write shortly about ***void*** data type.

void type has two important purposes:

1. To indicate the function does not return a value
2. To declare a generic pointer.

10. Define Dafault Argument.

In C++, one can assign default values to the formal parameters of a function prototype. The Default arguments allows to omit some arguments when calling the function.

Example :

```
defaultvalue(x,y);
defaultvalue(200,150);
defaultvalue(150);
defaultvalue(x,150);
```

11. What are parameters? List different type of parameters.

2 Types of parameters:

1. Formal Parameters –Associated with function header

Arguments or parameters are the means to pass values from the calling function to the called function. The variables used in the function definition as parameters are known as formal parameters.

2. Actual Parameters or Arguments- Associated with Calling Statement.

The constants, variables or expressions used in the function call are known as actual parameters.

12. What are the method of calling functions?

In C++, the arguments can be passed to a function in two ways. Based on the method of passing the arguments, the function calling methods can be classified as Call by Value method and Call by Reference or Address method.

14. What is inline functions? Write its Syntax & Adavatages.

An inline function looks like normal function in the source file but inserts the function's code directly into the calling program. To make a function inline, one has to insert the keyword **inline** in the function header**.**

Syntax :

inline returntype functionname(datatype parametername1, … datatype parameternameN)

Advantages of inline functions:

- ✓ Inline functions execute faster but requires more memory space.
- ✓ Reduce the complexity of using STACKS.

15. What is use of return statement?

Returning from the function is done by using the return statement. The return statement stops execution and returns to the calling function. When a return statement is executed, the function is terminated immediately at that point.

Syntax:

return expression/variable;

Example : return(a+b); return(a);

return; // to terminate the function

16. What is Recursive functions?

A function that calls itself is known as recursive function. And, this technique is known as recursion.

17. What are the information provided by the function prototype to the function?

1. Return data type,

2. Name of the function and a list of formal parameters or arguments.

Important 5 Marks

1. How will you call a function?

In C++, the arguments can be passed to a function in two ways. Based on the method of passing the arguments, the function calling methods can be classified as Call by Value method and Call by Reference or Address method.

Call by value Method

This method copies the value of an actual parameter into the formal parameter of the function. In this case, changes made to formal parameter within the function will have no effect on the actual parameter.

-----Any Relevant Example Program

Call by reference or address Method

This method copies the address of the actual argument into the formal parameter. Since the address of the argument is passed ,any change made in the formal parameter will be reflected **back in the actual parameter.**

-----Any Relevant Example Program

2. What is Inline Function?

Normally the call statement to a function makes a compiler to jump to the functions (the definition of the functions are stored in STACKS) and also jump back to the instruction following the call statement. This reduces the speed of program execution. Inline functions can be used to reduce the overheads like **STACKS** for small function definition.

An inline function looks like normal function in the source file but inserts the function's code directly into the calling program. To make a function inline, one has to insert the keyword inline in the function header.

Syntax:

An inline function looks like normal function in the source file but inserts the function's code directly into the calling program. To make a function inline, one has to insert the keyword **inline** in the function header**.**

Syntax :

inline returntype functionname(datatype parametername1, … datatype parameternameN)

Advantages of inline functions:

- ✓ Inline functions execute faster but requires more memory space.
- ✓ Reduce the complexity of using STACKS.

-----Any Relevant Example Program

3. What are the scope rules of variables?

Scope Rules of Variables

Scope refers to the accessibility of a variable. There are four types of scopes in C++.

They are:

1. Local scope,

A local variable is defined within a block.A block of code begins and ends with curly braces { }.

• The scope of a local variable is the block in which it is defined.

• A local variable cannot be accessed from outside the block of its declaration

2. Function scope,

1. The scope of variables declared within a function is extended to the function block, and all sub-blocks therein.

2. The life time of a function scope variable, is the life time of the function block. The scope of formal parameters is function scope.

3. File scope

A variable declared above all blocks and functions (including main ( )) has the scope of a file. The life time of a file scope variable is the life time of a program.

• The file scope variable is also called as global variable.

4. Class scope.

A class is a new way of creating and implementing a user defined data type. Classes provide a method for packing together data of different types.

Data members are the data variables that represent the features or properties of a class.

A class is a new way of creating and implementing a user defined data type. Classes provide a method for packing together data of different types.

Scope Resolution Operators:

1. The scope operator reveals the hidden scope of a variable. The scope resolution operator (::) is used for the following purposes.

2. To access a Global variable when there is a Local variable with same name. An example using Scope Resolution Operator.

-----Any Relevant Example Program

Chapter-12
ARRAYS & STRUCTURES

Important 2 & 3 Marks

1. What is an Array? What are its types?

"An array is a collection of variables of the same type that are referenced by a common name". There are different types of arrays used in C++. They are:

1. One-dimensional arrays
2. Two-dimensional arrays
3. Multi-dimensional arrays

2. What is Traversal?

1. Accessing each element of an array at least once to perform any operation is known as "Traversal".
2. Displaying all the elements in an array is an example of "traversal".

3. What is String? Or Explain One Dimensional Char Array.

A string is defined as a sequence of characters where each character may be a letter, number or a symbol.

Array declaration is:

char array_name[size];

In the above declaration, the size of the array must be an unsigned integer value.

For example,

char country[6];

4. What is Array of String? or Explain Two Dimensional Char Array.

An array of strings is a two-dimensional character array. The size of the first index (rows) denotes the number of strings and the size of the second index (columns) denotes the maximum length of each string.

For example: char Name[6][10];

5. How 2D Array Memory can represented?

There are two types of 2-D array memory representations. They are:

1. Row-Major order
2. Column-Major order

6. What is Structure?

Structure is a user-defined which has the combination of data items with different data types. This allows to group of variables of mixed data types together into a single unit.

7. What is Nested Structure?

The structure declared within another structure is called a nested structure

8. What is Array of Structure?

- ✓ A class may contain many students. So, the definition of structure for one student can also be extended to all the students. If the class has 20 students, then 20 individual structures are required.
- ✓ For this purpose, an array of structures can be used.
- ✓ An array of structures is declared in the same way as declaring an array with built-in data types like int or char.

Mobile: +91- 9677635246 E-Mail: nmohan707@gmail.com

Chapter-13

Introduction to Object Oriented Programming Techniques

Important 2 & 3 Marks

1. What is an Object?

Represents data and its associated function together into a single unit.

2. What is Procedural Programming?

Procedural means a list of instructions were given to the computer to do something.

Procedural programming aims more at procedures. This emphasis on doing things.

3. What is Modular Programming?

- ✓ Modular programming consist of a list of instructions that instructs the computer to do something.
- ✓ But this Paradigm consists of multiple modules, each module has a set of functions of related types. Data is hidden under the modules.

4. What OOP's Concepts Encourages?

The Object-Oriented Programming approach mainly encourages:

Modularisation:

Where the program can be decomposed into modules.

Software re-use:

Where a program can be composed from existing and new modules.

5. What are the features of OOPs?

Main Features of Object Oriented Programming:-

1. Data Abstraction
2. Encapsulation
3. Modularity
4. Inheritance
5. Polymorphism

6. What is Encapsulation?

The mechanism by which the data and functions are bound together into a single unit is known as Encapsulation. It implements abstraction.

7. What is Data Abstraction?

Abstraction refers to showing only the essential features without revealing background details

8. What is Modularity?

Modularity is designing a system that is divided into a set of functional units (named modules) that can be composed into a larger application.

9. What Inheritance?

1. Inheritance is the technique of building new classes (derived class) from an existing Class (base class).

2. The most important advantage of inheritance is code reusability.

10. What is Polymorphism?

Polymorphism is the ability of a message or function to be displayed in more than one form.

11. What are the Advantages of OOPs?

1. Re-usability
2. Redundancy
3. Easy Maintenance:
4. Security:

12. What are the Dis-Advantages of OOPs?

1. Size
2. Effort
3. Speed

**********8

Chapter-14
Classes and objects

Important 2 & 3 Marks

1. What is the need for class?

- Class is a way to bind the data and its associated functions together.
- Classes are needed to represent real world entities that not only have data type properties but also have associated operations.
- It is used to create user defined data type

2. Write the Syntax for class.

```
class class-name
{
private:
        variable declaration;
        function declaration;
protected:
        variable declaration;
        function declaration;
public:
        variable declaration;
        function declaration;
};
```

3. What is Data Hiding?

Data hiding is one of the important features of Object Oriented Programming which allows preventing the functions of a program to access directly the internal representation of a class type.

4. What is Access Specifiers or Visibility Mode?

Access Specifiers are used to define that how member function can be used with in a class.

Which can be classified into 3 categories:

They are:

1. Public:

A public member is accessible from anywhere outside the class but within a program.

2. Private:

A private member cannot be accessed from outside the class. Only the class member functions can access private members. By default all the members of a class would be private.

3. Protected:

A protected member is very similar to a private member but it Provides one additional benefit that they can be accessed

5. What are the class members?

Class comprises of members.

Members are classified as ***Data Members and Member functions.***

Data Members:

- ✓ Data members are the data variables that represent the features or properties of a class.
- ✓ Data members are also called as attributes.

Member functions:

- ✓ Member functions are the functions that perform specific tasks in a class.
- ✓ Member functions are called as methods

6. Define Method of a Class.

The member functions of a class can be defined in two ways.

(1) Inside the class definition

(2) Outside the class definition

7. Write the Syntax for Outside class definition.

Syntax :

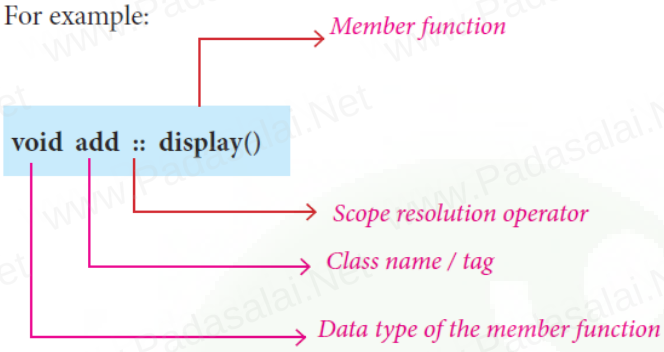
return_type class_name :: function_name (parameter list)

{

function definition

}

For example:



8. How will you create an Object? Explain different way of creating an object.

Objects can be created in two methods,

(1) Global object

(2) Local object

9. How member function can be called?

The members of a class are referenced (accessed) by using the object of the class followed by the dot (membership) operator and the name of the member.

Syntax:

Object_name . function_name(actual parameter);

10. What is Array of Objects?

An array which contains the class type of element is called array of objects. It is declared and defined in the same way as any other type of array.

11. What is Nesting of Member function?

A member function can call another member function of the same class directly without using the dot operator. This is called as nesting of member functions.

12. Define Scope Resolution Operator.

If there are multiple variables with the same name defined in separate blocks then :: (scope resolution) operator will reveal the hidden file scope(global) variable.

13. How object can be passed to Member Function?

Objects can also be passed as arguments to a member function just like any other data type of C++.Objects can also be passed in both ways

(1) Pass By Value

(2) Pass By Reference

14. What is Nested Class?

When one class become the member of another class then it is called Nested class and the relationship is called containership.

Classes can be nested in two ways:-

1. By defining a class within another class
2. By declaring an object of a class as a member to another class

15. What is Container Class?

- ✓ Whenever an object of a class is declared as a member of another class it is known as a container class.
- ✓ In the container-ship the object of one class is declared in another class.

16. What is Constructor? Write its functions.

1. When an instance of a class comes into scope, a special function called the constructor gets executed.
2. The constructor function name has the same name as the class name.
3. The constructors return nothing.
4. They are not associated with any data type.
5. It can be defined either inside class definition or outside the class definition.

Functions of Constructors:

1. To allocate memory space to the object and
2. To initialize the data member of the class object

17. What is Default Constructors or Compiler Generated Constructor or Non-Parameterized Constructor?

A constructor that accepts no parameter is called default constructor.

For example class name is add: add( ) → No Parameter inside the bracket.

18. What is Parameterized Constructor or Used Defined Constructor?

A constructor which can take arguments is called parameterized constructor

For example class name is add: add(int s1, int s2) → Parameter given inside the bracket.

19. Define Invocation of Constructor.

There are two ways to create an object using parameterized constructor

1. Implicit call

In this method ,the parameterized constructor is invoked automatically whenever an object is created.

For example simple s1(10,20);

2. Explicit call

In this method ,the name of the constructor is explicitly given to invoke the parameterized constructor so that the object can be created and initialized .

For example : simple s1=simple(10,20);

20. What is Copy Constructor?

A constructor having a reference to an already existing object of its own class is called copy constructor .

For example : add(add &a)

21. What are the Characteristics of Constructor?

Characteristics of Constructors

1. The name of the constructor must be same as that of the class
2. No return type can be specified for constructor
3. A constructor can have parameter list
4. The constructor function can be overloaded
5. They cannot be inherited but a derived class can call the base class constructor
6. The compiler generates a constructor, in the absence of a user defined constructor.
7. Compiler generated constructor is public member function
8. The constructor is executed automatically when the object is created
9. A constructor can be used explicitly to create new object of its class type

22. What is Destructor? Write its Characteristics?

When a class object goes out of scope, a special function called the destructor gets executed. The destructor has the same name as the class tag but prefixed with a ~(tilde).Destructor function also return nothing and it does not associated with any data type.

Characteristics of Destructors

1. The destructor has the same name as that of the class prefixed by the tilde character '~'.
The destructor cannot have arguments

2. It has no return type

3. Destructors cannot be overloaded i.e., there can be only one destructor in a class

4. In the absence of user defined destructor, it is generated by the compiler

5. The destructor is executed automatically when the control reaches the end of class scope to destroy the object

6. They cannot be inherited

Chapter-15
Polymorphism

Important 2 & 3 Marks

1. What is Function Overloading?

The ability of the function to process the message or data in more than one form is called as function overloading.

2. What is Polymorphism?

The word polymorphism means many forms (poly – many, morph – shapes) Polymorphism is the ability of a message or function to be displayed in more than one form.

3. What is Overload Solution?

The process of selecting the most appropriate overloaded function or operator is called "O**verload resolution"**

4. What are the rules for function overloading?

1. The overloaded function must differ in the number of its arguments or data types
2. The return type of overloaded functions are not considered for overloading same data type
3. The default arguments of overloaded functions are not considered as part of the parameter list in function overloading.

5. What is Constructor Overloading?

Function overloading can be applied for constructors, as constructors are special functions of classes .

A class can have more than one constructor with different signature.Constructor overloading provides flexibility of creating multiple type of objects for a class.

6. What is Operator Overloading?

The term operator overloading, refers to giving additional functionality to the normal C++ operators like +,++,-,—,+=,-=,*.<,>. It is also a type of polymorphism in which an operator is overloaded to give user defined meaning to it .

7. What are the Operators cannot be overloaded?

1. scope operator ::
2. sizeof
3. member selector .
4. member pointer selector *
5. ternary operator ?:

8. What are the Rules/ Restrictions of Operator Overloading?

1. Precedence and Associativity of an operator cannot be changed.
2. No new operators can be created, only existing operators can be overloaded.
3. Cannot redefine the meaning of an operator's procedure. You cannot change how integers are added.Only additional functions can be to an operator
4. Overloaded operators cannot have default arguments.
5. When binary operators are overloaded, the left hand object must be an object of the relevant class

Important 5 Marks

1. What is Function Overloading? Explain with an example program.

The ability of the function to process the message or data in more than one form is called as function overloading.

Rules:

1. The overloaded function must differ in the number of its arguments or data types
2. The return type of overloaded functions are not considered for overloading same data type
3. The default arguments of overloaded functions are not considered as part of the parameter list in function overloading.

Program:

```
#include <iostream>
using namespace std;
void print(int i)
{cout<< " It is integer " << i <<endl;}
void print(double f)
{ cout<< " It is float " << f <<endl;}
void print(string c)
{ cout<< " It is string " << c <<endl;}
int main() {
print(10);
print(10.10);
print("Ten");
return 0;
}
```

Output:

```
It is integer 10
It is float 10.1
It is string Ten
```

2. What is Operator Overloading? Explain with an example Program.

The term operator overloading, refers to giving additional functionality to the normal C++ operators like +,++,-,—,+=,-=,*.<,>. It is also a type of polymorphism in which an operator is overloaded to give user defined meaning to it .

Restrictions:

1. Precedence and Associativity of an operator cannot be changed.
2. No new operators can be created, only existing operators can be overloaded.
3. Cannot redefine the meaning of an operator's procedure. You cannot change how integers are added.Only additional functions can be to an operator
4. Overloaded operators cannot have default arguments.
5. When binary operators are overloaded, the left hand object must be an object of the relevant class

Program:

```
#include<string.h>
#include<iostream>
using namespace std;
class strings
{
public:
char s[20];
void getstring(char str[])
{
strcpy(s,str);
}
void operator+(strings);
};
void strings::operator+(strings ob)
{
strcat(s,ob.s);
cout<<"\nConcatnated String is:"<<s;
}
int main()
{
strings ob1, ob2;
char string1[10], string2[10];
cout<<"\nEnter First String:";
cin>>string1;
ob1.getstring(string1);
cout<<"\nEnter Second String:";
cin>>string2;
ob2.getstring(string2);
//Calling + operator to Join/Concatenate strings
ob1+ob2;
return 0;
}
```

Output:

Enter First String:COMPUTER

Enter Second String:SCIENCE

Concatenated String is:COMPUTERSCIENCE

3. What is Constructor Overloading? Explain with an example Program.

Function overloading can be applied for constructors, as constructors are special functions of classes .

A class can have more than one constructor with different signature.Constructor overloading provides flexibility of creating multiple type of objects for a class.

Program:

```
#include<iostream>
using namespace std;
class add
{
int num1, num2, sum;
public:
add()
{
cout<<"\n Constructor without parameters.. ";
num1= 0;
num2= 0;
sum = 0;
}
add ( int s1, int s2 )
{
cout<<"\n Parameterized constructor... ";
num1= s1;
num2=s2;
sum=0;
}
add (add &a)
{
cout<<"\n Copy Constructor ... ";
num1= a.num1;
num2=a.num2;
sum = 0;
}
void getdata()
{
cout<<"\nEnter data ... ";
cin>>num1>>num2;
}
void addition()
{
sum=num1+num2;
}
void putdata()
```

```
{
cout<<"\n The numbers are..";
cout<<num1<<'\t'<<num2;
cout<<"\n The sum of the numbers are.. "<< sum;
}
};
int main()
{
add a, b (10, 20) , c(b);
a.getdata();
a.addition();
b.addition();
c.addition();
cout<<"\n Object a : ";
a.putdata();
cout<<"\n Object b : ";
b.putdata();
cout<<"\n Object c.. ";
c.putdata();
return 0;
}
```

Output:

```
Constructor without parameters..
Parameterized constructor...
Copy Constructor ...
Enter data ... 20 30
Object a :
The numbers are..20 30
The sum of the numbers are.. 50
Object b :
The numbers are..10 20
The sum of the numbers are.. 30
Object c..
The numbers are..10 20
The sum of the numbers are.. 30
```

Chapter-16
Inheritance

Important 2 & 3 Marks

1. What is Inheritance? What are the types of inheritance?

1. The Process of acquiring base class properties is known an Inheritance

2. It is a process of creating new classes called derived classes, from the existing or base classes.

Types:

1. Single Inheritance,

2. Multiple inheritance,

3. Multilevel inheritance,

4. Hybrid inheritance and

5. Hierarchical inheritance

3. What are the points should be observed while defining the derived class?

While defining a derived class, the derived class should identify the class from which it is derived.The following points should be observed for defining the derived class.

1. The keyword class has to be used

2. The name of the derived class is to be given after the keyword class

3. A single colon

4. The type of derivation (the visibility mode), namely private, public or protected. If no visibility mode is specified, then by default the visibility mode is considered as private.

5. The names of all base classes (parent classes) separated by comma.

Ex:

```
class derived_class_name : visibility_mode base_class_name
{
// members of derived class
};
```

4. What is Visibility Mode?

An important feature of Inheritance is to know which member of the base class will be acquired by the derived class. This is done by using visibility modes.

5. What is Overriding?

In case of inheritance there are situations where the member function of the base class and derived classes have the same name. If the derived class object calls the overloaded member function it leads confusion to the compiler as to which function is to be invoked. The derived class member function have higher priority than the base class member function.This shadows the member functionof the base class which has the same name like the member function of the derived class. The scope resolution operator resolves this problem.

6. Explain 'this' Pointer?

'this' pointer is a constant pointer that holds the memory address of the current object.

7. What is Base class & Derived Class?

Base Class	:	A Class from other classes are derived
Derived Class	:	A Class which is derived from base class.

****

Chapter-17
Computer Ethics And Cyber Security
Important 2 & 3 Marks

1. What is Computer Crime?

Illegal access to the computer or network is called Computer Crime.

For example – illegal money transferviainternet.

2. What are the types of Cyber Crimes?

1. Crime Function
2. Cyber stalking
3. Malware
4. Denial of service attack
5. Fraud
6. Harvesting
7. Identity theft

3. What is Ethics?

Ethics means "What is wrong and What is Right". It is a set of moral principles that rule the behavior of individuals who use computers.

4. What are the Guide Lines of Computer Ethics?

1. **Honesty:** Users should be truthful while using the internet.

2. Confidentiality: Usersshould not share any important information with unauthorized people.

3. Respect: Each user should respect the privacy of other users.

4. Professionalism: Each user should maintain professional conduct.

5. Obey The Law: Users should strictly obey the cyber law in computer usage.

6.Responsibility: Each user should take ownership and responsibility for their actions

5. What is Piracy? List its types.

Making and using duplicate software or hardware components is called "Piracy".

6. What is Hacking?

Hacking is intruding into a computer system to steal personal data without the owner's permission or knowledge (liketo steal a password).

7. What is cracking?

Cracking is where someone edits a program source so that the code can be exploited or modified. A cracker (also called a black hat or dark side hacker) is a malicious or criminal hacker. "Cracking" means trying to get into computer systems in order to steal, corrupt, or illegitimately view data.

8. What is IRC?

It can pretend being your friend and talk to you on Internet Relay Chat(IRC) or by Instant messenger.

9. What are the types of Attacks?

1. Virus
2. Worms
3. Spyware
4. Ransomware

10. What is Phishing?

Phishing is a type of computer crime used to attack, steal user data, including login name, password and credit card numbers.

11. What is Pharming?

- Pharming is a scamming practice in which malicious code is installed on a personal computer or server, misdirecting users to fraudulent web sites without their knowledge or permission.
- Pharming has been called "phishing without a trap".

12. What is MITM?

Man In The Middle (MITM)

Man-in-the-middle attack (MITM; also Janus attack) is an attack where the attacker secretly relays and possibly alters the communication between two parties who believe they are directly communicating with each other.

13. What is Cookies?

Cookies are messages that web servers pass to your web browser when you visit Internet sites

14. What is Firewall?

A firewall is a network security system that monitors and controls incoming and outgoing network traffic based on predetermined security rules.

15. What is Proxy Servers?

A proxy server is a gateway between a local network and a larger-scale network such as the Internet. Proxy servers provide increased performance and security.

16. What is Encryption & Decryption? List its types.

Encryption and decryption are processes that ensure confidentiality that only authorized persons can access the information.

Encryption is the process of translating the plain text data (plaintext) into random and mangled data (called cipher-text).

Decryption is the reverse process of converting the cipher-text back to plaintext.

Types Of Encryption

There are two types of encryption schemes as listed below:

1. Symmetric Key encryption
2. Public Key encryption

17. What is Digital Signature?

Digital signatures are based on asymmetric cryptography and can provide assurances of evidence to origin, identity and status of an electronic document, transaction or message, as well as acknowledging informed by the signer.

☆☆

Chapter-18
Tamil Computing

Important 2 & 3 Marks

1. What is E-Governance?

Getting Government services through internet is known as e-Governance.

2. List some of the familiar tamil keyboards.

NHM Writer, E-Kalappai and Lippikar – are familiar Tamil keyboard interfaces software that is used for Tamil typing which works on Tamil Unicode, using phonetics

3. What is E-Library?

E-Libraries are portal or website of collection of e-books. Tamil e-Library services provide thousands of Tamil Books as ebooks mostly at free of cost. It is the most useful service to Tamil people who live far away from their home land.

4. Expand & Give Short Note on the following: TSCII , ISCII, Unicode

1. TSCII: Tamil Script Code for Information Interchange

TSCII (Tamil Script Code for Information Interchange) is the first coding system to handle our Tamil language in an analysis of an encoding scheme that is easily handled in electronic devices, including non-English computers.

This encoding scheme was registered in IANA (Internet Assigned Numbers Authority) unit of ICANN.

Mobile: +91- 9677635246 E-Mail: nmohan707@gmail.com

2. ISCII: Indian Script Code for Information Interchange

This is one of the encoding schemes specially designed for Indian languages including Tamil. It was unified with Unicode.

3. UNICODE:

Unicode is an encoding system, designed to handle various world languages, including Tamil.

5. What are the tamil keyboards layouts used in Android?

Sellinam and Ponmadal – are familiar Tamil keyboard layouts that works on Android operating system in Smart phone using phonetics.

Points to Remember:

- Tamil topped the list of the most widely used languages in India by the end of 2016, while **42%** are using the Internet.
- ***Google and Bing*** provide searching facilities in Tamil.
- Getting Government services through internet is known as **e-Governance.**
- Tamil e-Library services provide thousands of Tamil Books as ebooks mostly at free of cost.
- Thamizpori (தமிழ்பொறி) is a Tamil tranlation application having more than **30000** Tamil words equalent to English words.
- The first Tamil programming language is ***"Ezhil"*** (எழில்)
- Unicode is an encoding system, designed to handle various world languages, including Tamil.
- Among the various encoding scheme, Unicode is the suitable to handle Tamil.
- Windows Tamil Environment interface should be downloading and install from internet.

All the best !!!

Bring out your best !!! Wish you all get centum in public examination. . . .

N. Mohan, MCA., B.Ed.,
Dept. of Computer Science
SSV Matric Hr Sec School,
Sivagiri, Erode-Dt

Mobile: +91- 9677635246 E-Mail: nmohan707@gmail.com