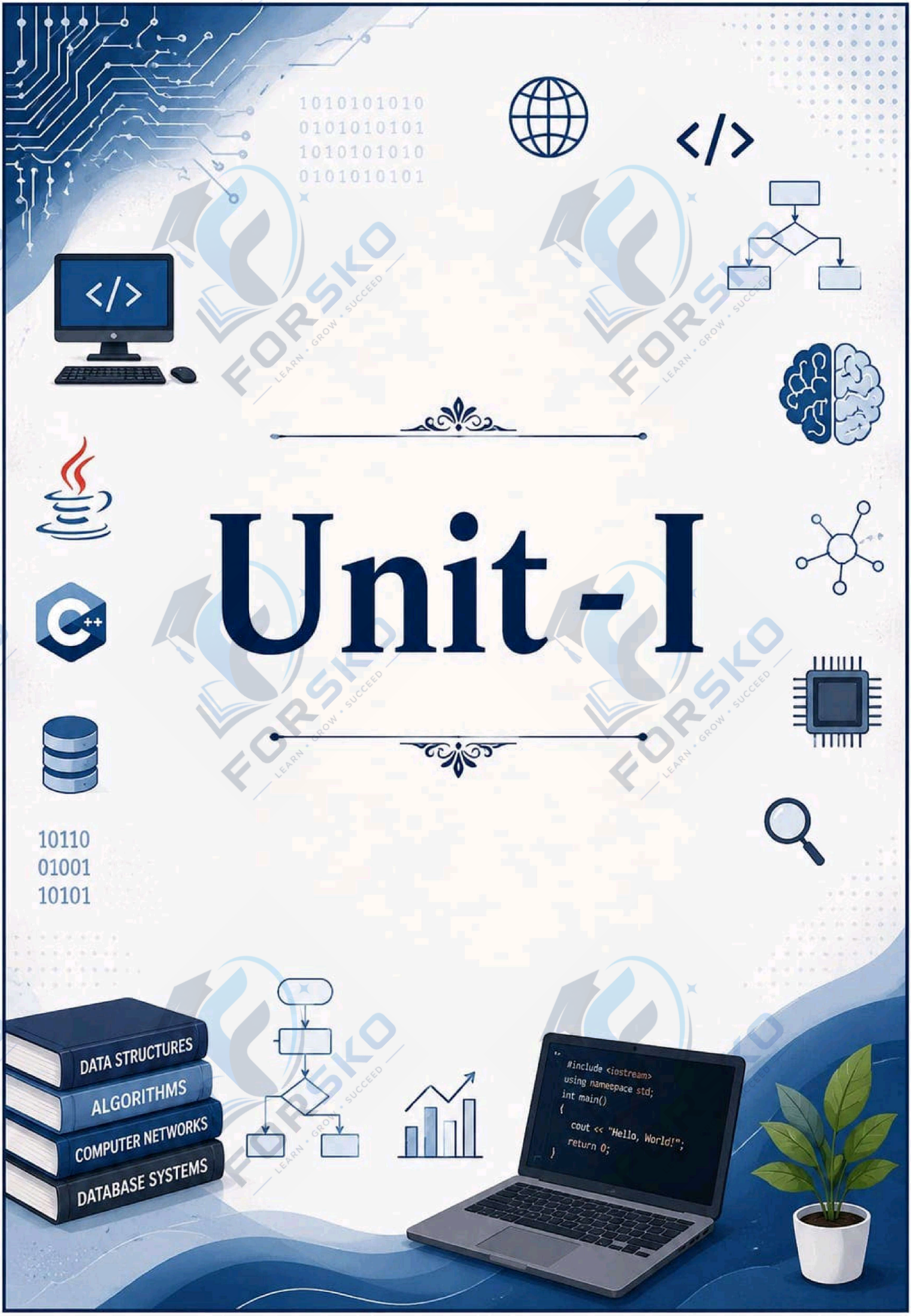
Notes


1010101010
0101010101
1010101010
0101010101
DATA STRUCTURES
ALGORITHMS
COMPUTER NETWORKS
DATABASE SYSTEMS
Good Notes
Better Understanding
Better Results

Unit-I


DATA STRUCTURES
ALGORITHMS
COMPUTER NETWORKS
DATABASE SYSTEMS
#include <iostream>
using nameepace std;
int main()
{
cout << "Hello, World!";
return 0;
}

Chapter No-1

8 M

Short Answer -

1. Define an algorithm.

Ans:- An algorithm is a step-by-step procedure or set of rules to be followed in problem-solving operations.

OR

An algorithm is a sequence of instruction or procedures which is written to solve any complex problem. An algorithm can be implemented in any programming language and used in the computer system.

2. What are the basic symbols used in flowchart

Ansr The basic symbols used in flowcharts are Oval, Rectangle, Arrow, Diamond and parallelogram.

3. What are the different types of programming Languages?

Ansr The three most commonly used types of programming language are

1. Low-level Language
2. Middle-level Language
3. High-Level Language

4. List any three advantages of C programming.

Ansr 1) Simple and Efficient
2) Fast
3) Function-Rich Library.

5. What is a character set in C?

Ans:- The character set in C refers to the set of valid characters that can be used in a C program. These characters include letters, ~~digital~~ digits, special characters, and white spaces.

6. Define identifiers in C.

Ans:- An identifier is a name that identifies a variable, function, array, or other user-defined ~~function~~ programing element. Identifiers can be used to refer to program elements without knowing their specific location in memory.

7. ~~Wel~~ What are constants and variables in C?

Ans:- The constants are those variables or values in the C which cannot be modified once they are defined in the program.

A variable in simple terms is a storage place that has some memory allocated to it. It is used to store some form of data and retrieve it when required.

8. Explain symbolic constants with an example

Ans:- A symbolic constant in C is a name that represents a fixed value in a program. Symbolic constants are useful because they make code more readable and maintain-able. (eg- # define PI 3.14)

9. What is the difference between implicit and explicit type conversion?

Ans:- Implicit done automatically by the compiler
eg- int a = 5; float b = a;

Explicit done manually by the programmer
eg- float a = 5.5; int b = (int)a;

10. What are operators in C? Given an example.

Ans- Operator in C are ~~symbolic~~ symbols that performs operations on variables and values.
1) Arithmetic Operators.
2) Relational Operators.

Long-Answer.

1. Explain the history of the C programming language in detail.

→ 1) C is programing language developed by Dennis Ritchie in the early 1970's.
2) It was one of the most popular and influential programming languages worldwide
3) C was based on the B language, which was developed by Ken Thompson.
C improved B by adding data types and structures.
4) The first C compiler was written in C itself in 1972, which made it portable and flexible across systems.
5) C was deused to rewrite the unix operating system, making it more portable and efficient, which increased Cs popularity.
6) C made programs more portable, meaning they could run on different machines with little or no modification.
7) After ANSI, ISO adopted the standard in 1990, making it globally recognized.
8) C become one of the most widely used programming languages and is the foundation of mordern languages like C++, Java and C#.

2- Continue-

5) Rich Library Support-

C has a wide range of built-in libaries for operations like math, file handling, and string manipulation.

eg- #include<math.h>.

allow using function like sqrt() and pow().

6) Foundation for Other Languages-

C is the base for any modern languages like C++, Java and Python.

eg- Knowledge of C helps in understanding C++ and memory management in Java.

2. Discuss the advantages of the C programming Language with example.

→

1) Simple and Easy to Learn.-

C has a clean and structured syntax which makes it easy to learn and understand. eg- printf("Hello World");

2) High Performance -

C is a compiled language and provides fast execution, making it suitable for system-level programming.
eg- Operating systems and embedded systems are often written in C for speed.

3) Portability -

C program written on Windows can be compiled and run on Linux with little modification. eg--A program written on windows can be compiled and run on Linux with little modification.

4) Low-Level Memory Access-

C allows direct access to memory using pointers, which is useful for system programming.
eg- Managing memory manually using malloc() and free() function.

Explain the structure of C programming with simple program.

The basic structure of C program is divided into 6 parts which makes its easy to read modify, document, and understand in a particular format.

There are 6 basic sections- &

1. Documentation-

This section contain the description of the program, name of program and date, time of the program. Its specified that the start of program in the form of comments.

eg- // Description, name, program name, date, time, etc.

OR

/*Description, name, program name, date, time, etc*/

2. Preprocessor Section. (Link)

All the header files of the program will be declared in the preprocessor section of the program. Header files help us to access other's improved code into our code.

eg-

```
#include <stdio.h>
#include <string.h>
```

3. Definition Section-

Preprocessors are the programs that process our source code before the compilation of actual process. Preprocessor directives start with the '#' symbols.

The #define preprocessor is used to create a constant in program. Whenever this name is encountered the compiler swaps this name with the actual function that was defined earlier.

eg- #define PI 3.14

4. Global Decleration -

The global decleration section contains global variables, functions declaration and static variables. Variables declared int this section can be used in any where in program.

eg-

```
int num=15;
```

5. Main() Function-

The main function of program is written here. Operations like declaration and execution are can performed inside curly braces {} of the main program.

eg-

```
void main()
int main()
```

Sub programs-

User-defined functions are called in this section of the program. The control of the program is shifted to the called function whenever they called from the main or outside ~~the main or outside~~ the main function.

```
int sum (int x, int y)
{ return x+y; }
```

```
int C
int sum(int x, int y)
{ return x+y; }
```

Example of C program Structure.

```
// Declaration.
/* File - Sum.c
 * author - Shid
 * time - 19:20 */
// Link
#include <stdio.h>
// Defination
#define PI 3.14

// Global Declaration
int num=18;

// Main() Function
int main()
{ .... }

// Sub-Program
int sum (int x, int y)
{ return x+y; }
```

4. Describe the character set used in C programming?

→ The character set is the collection of all characters that are used in programs. This character includes Letters, digits special characters, and white spaces.

1) Alphabets - C supports both uppercase and lowercase
   - Uppercase Letters - A, B, ... Z.
   - Lowercase Letters - a, b, c, ... z.

2) Digits - C allows the use of numerical digits.
   - Digits - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

3) Special Character - Special symbols used in C
   ~ . ' ! @ # $ % ^ & * ( ) { } _ - + / \
   ; : [ ] | ' " < > , ? etc.
   This symbols used for arithmetic operation Logical operations, and punctuation.

4) White Spaces - C also includes white spaces, Carriage Space (' '), Tab ('\t'), Newline ('\n') return ('\r').

5) Escape Sequences - Escape sequences are special character represent certain action.

- \n - Newline
- \t - Tab
- \b - Backspace
- \r - Carriage return
- \a - Alert bell
- \\ - Backslash
- \' - single quote
- \" - Double quote.

What are the keywords in C? Provide a list of some important keywords.

Keywords are special words that have a fixed meaning in the C language. They are reserved words which means we cannot use them as variable names or identifiers. C has 32 keywords, and each one performs a specific task in program

List of some important keywords.

Keywords	Use / Meaning
int	Declares an integer variable
float	Declares a variable with decimal
char	Declares a character variable.
if	Used for decision making
else	Used with if for alternative condition
while	Used for Loops
for	Used for loops (counting loops)
return	Ends a function & send value back
void	Shows that a function has a return value.
break	Stops a loop early.

6. Differentiate between variables and constant in C. Provide example.

→

Constant	Variables
1) A constant is a variable or value that cannot be altered once defined.	1) A variable is a name associated with some memory location.
2) Constant is used to hold the fixed values which we can retrieve later but cannot change.	2) Variable is used to hold some values that can be changed according to need.
3) Constants are generally stored in the text segment as they are read-only	3) Variable are stored inside a data segment heap, or stack depending on the environment it decleard in.
4) We can only assign a value to the constant while defining it.	4) We can assign value to the variable anytime.
5) Constant can be defined by using #define or const keyword.	5) Variable can only defined using the standard variable definition syntax.
eg- #define PI 3.14 const int PI = 3.14	eg- int var = 25; var = 10;

Discuss the different types of programming languages with example.

There are three main types of programming

1. Low-Level Language
2. Middle-Level Language
3. High-Level Language

1. Low-Level Language-

Low level programming languages provide minimal abstraction between the language and the processor's instructions. These low-level languages closely similar to the machine code and are called close to hardware. The advantage is that the processors can run low-level program directly without any interpreter or compiler. Because of this low-level programming language are speedy to run.

e.g.- Binary Language, Machine language, and Assembly Language.

2. Middle-Level Language-

A middle level language refers to all languages between high-level and low-level programming languages. A middle level programming language act as a bridge between raw hardware and programs by interacting with abstraction layers, Because of this it is called pseudo or intermediate-level proging languag

eg- C, Java, and C++.

3. High-Level Language-

High-Level language allow for the greatest ~~abstr~~ abstaction between machine code and the language itself, so that they appear closer to human languages than machine languages. For computers to understand and executes commands, we have to use compilers that convert the HLL into computer languages.
The main benifit of high-level programming languages is that they are simple to learn, write, and maintain. or

eg- Phython, Java, Javascript, Pascal, PHP, Swift etc.

10. Write a flowchart for a program that takes two numbers as input and prints their sum.