BASIC PROGRAMMING LANGUAGE

LESSON 1
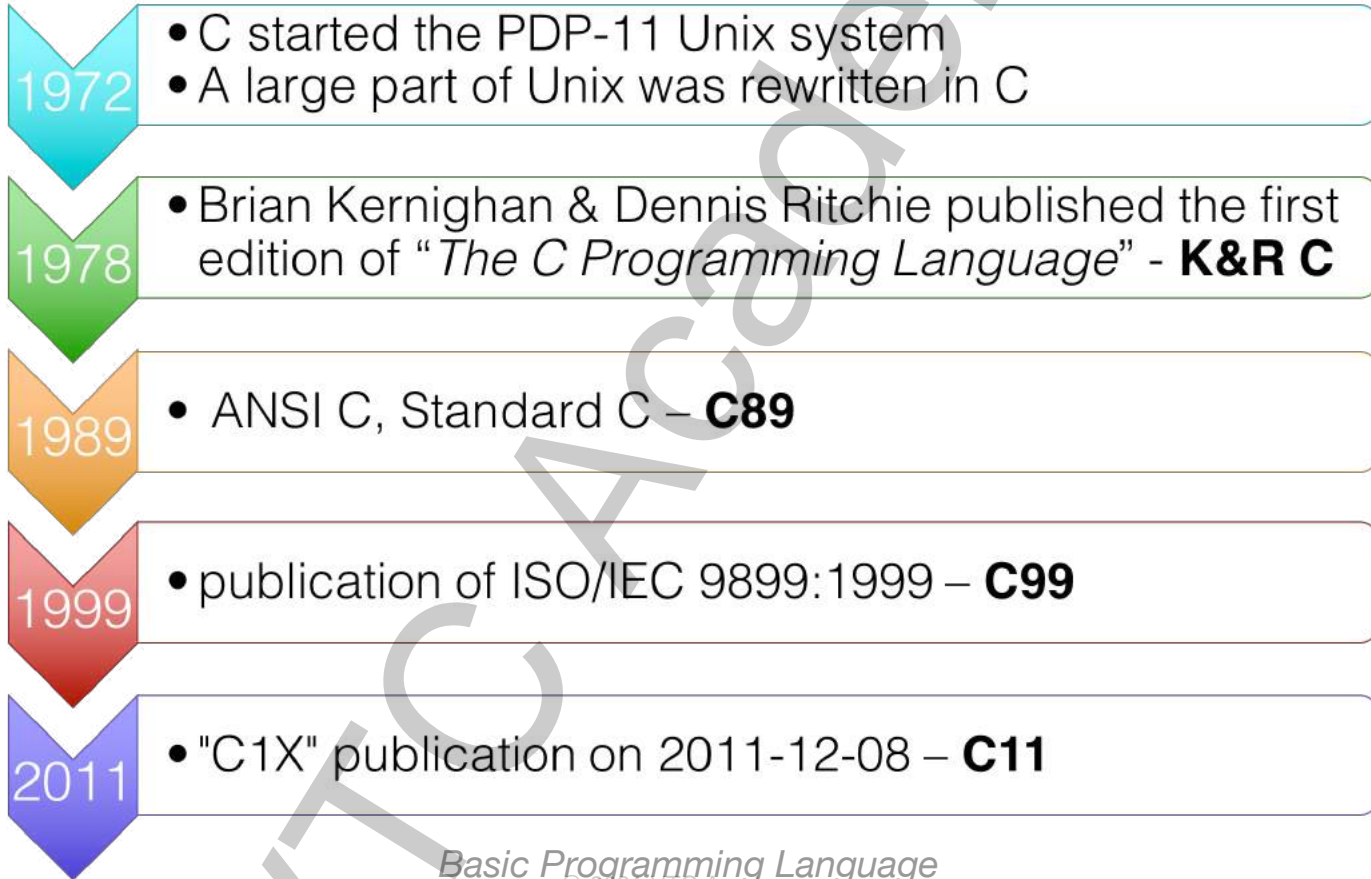
C Programming Language and Problem Solving

CONTENT

C Programming Language Introduction

- C is a general-purpose, procedural computer programming language supporting structured programming, lexical variable scope, and recursion, with a static type system.

- By design, C provides constructs that map efficiently to typical machine instructions.

THE

C

PROGRAMMING
LANGUAGE

C Programming Language History

1972
- C started the PDP-11 Unix system
- A large part of Unix was rewritten in C

1978
- Brian Kernighan & Dennis Ritchie published the first edition of "*The C Programming Language*" - **K&R C**

1989
- ANSI C, Standard C – **C89**

1999
- publication of ISO/IEC 9899:1999 – **C99**

2011
- "C1X" publication on 2011-12-08 – **C11**

Why Use C?

- C was initially used for system development work, particularly the programs that make-up the operating system.

- C was adopted as a system development language because it produces code that runs nearly as fast as the code written in assembly language.

- Applications of C:

  - Operating Systems, Language Compilers, Assemblers

  - Text Editors, Print Spoolers, Network Drivers, Modern Programs

  - Databases, Language Interpreters, Utilities,…

C Development Environment

- Online:

  - http://www.tutorialspoint.com/compile_c_online.php

  - https://replit.com/

- Local Environment Setup:

  - Text Editor: Visual Studio Code, Dev-C++,…

  - C Compiler: GNU C/C++

    - Linux: You will have to install it yourself using the detailed instructions available at http://gcc.gnu.org/install/

    - macOS: The Xcode development environment from Apple's web site and follow the easiest way to obtain GCC is to download the simple installation instructions

    - Windows: To install GCC on Windows, you need to install MinGW

C Program Structure

- A C program basically consists of the following parts:

    - Preprocessor Commands

    - Functions

   - Variables

    - Statements & Expressions

    - Comments

```c
#include <stdio.h>

int main() {
    /* my first program in C */
    printf("Hello, World! \n");

    return 0;
}
```

C Program Structure

- The first line of the program `#include <stdio.h>` is a preprocessor command, which tells a C compiler to include stdio.h file before going to actual compilation.

- The next line `int main()` is the main function where the program execution begins.

- The next line `/*...*/` will be ignored by the compiler and it has been put to add additional comments in the program. So such lines are called comments in the program.
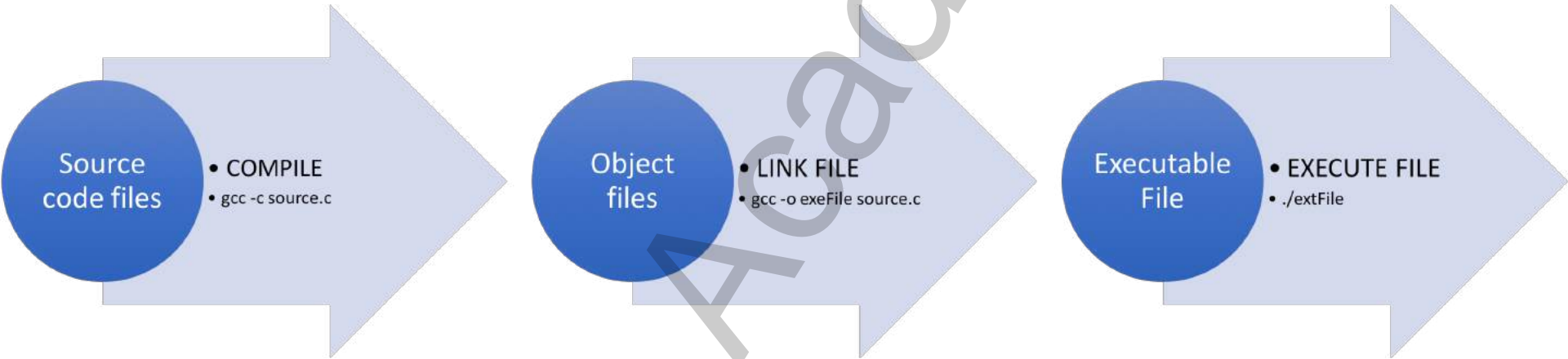
- The next line `printf(...)` is another function available in C which causes the message "Hello, World!" to be displayed on the screen.

- The next line `return 0;` terminates the `main()` function and returns the value 0.
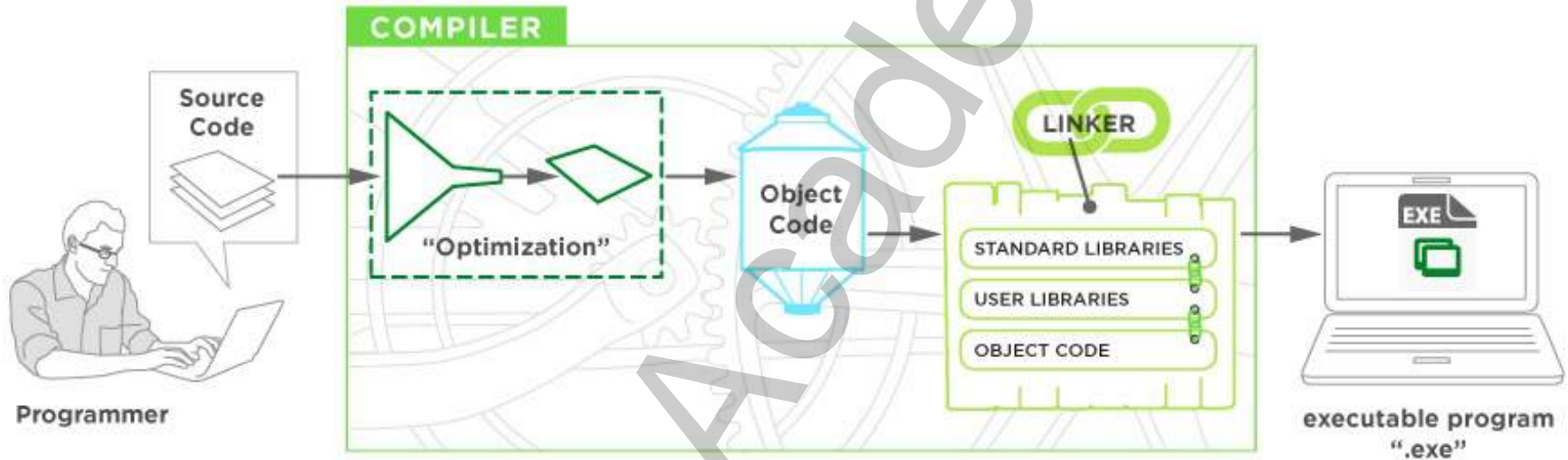
Compiled languages are converted into machine-readable code **prior to execution**, compared with interpreted languages, which are compiled to machine code **at the time of execution**.

Compile & Execute C Program

Source code files
- COMPILE
- gcc -c source.c

Object files
- LINK FILE
- gcc -o exeFile source.c

Executable File
- EXECUTE FILE
- ./extFile

Basic Syntax

- Tokens:
    - C program consists of various tokens and statements
    - A token is either a keyword, an identifier, a constant, a string literal, or a symbol

- Semicolons:
    - The semicolon (;) is a statement terminator
    - Each individual statement must be ended with a semicolon It indicates the end of one logical entity

Basic Syntax

- Comments:
    - Comments are like helping text in your C program & they are ignored by the compiler
    - Examples:

    ```
    // this is single line comment
    /* this is multiple lines comment */
    ```

- C is case-sensitive programming language

- Identifiers:

  - Using identifiers for variables, constant, function, user defined data type

  - Identifier does not allow punctuation characters such as @, $, %, space and other special characters within

  - Identifiers in C is case-sensitive

  - Examples: mohd, zara, abc, move,_name a_123, myname50, _temp, j, a23b9, retVal

Basic Syntax

- Keywords:

 - These reserved words may not be used as constants or variables or any other identifier names

  - Keywords in C:

auto	else	long	switch	break	enum	register	typedef
case	extern	return	union	char	float	short	unsigned
const	for	signed	void	continue	goto	sizeof	volatile
default	if	static	while	do	int	struct	_Packed
double							

Basic Syntax

- Whitespace:

  - A line containing only whitespace, possibly with a comment, is known as a blank line, and a C compiler totally ignores it

  - Whitespace is the term used in C to describe blanks, tabs, newline characters and comments

  - Whitespace separates one part of a statement from another and enables the compiler to identify where one element in a statement, such as int, ends and the next element begins

  - Examples:

    ```
    int age;
    fruit = apples + oranges;    // get the total fruit
    ```

Problem Solving

Solving problems is the core of computer science. Programmers must first understand how a human solves a problem, then understand how to translate this "algorithm" into something a computer can do, and finally how to "write" the specific syntax (required by a computer) to get the job done.
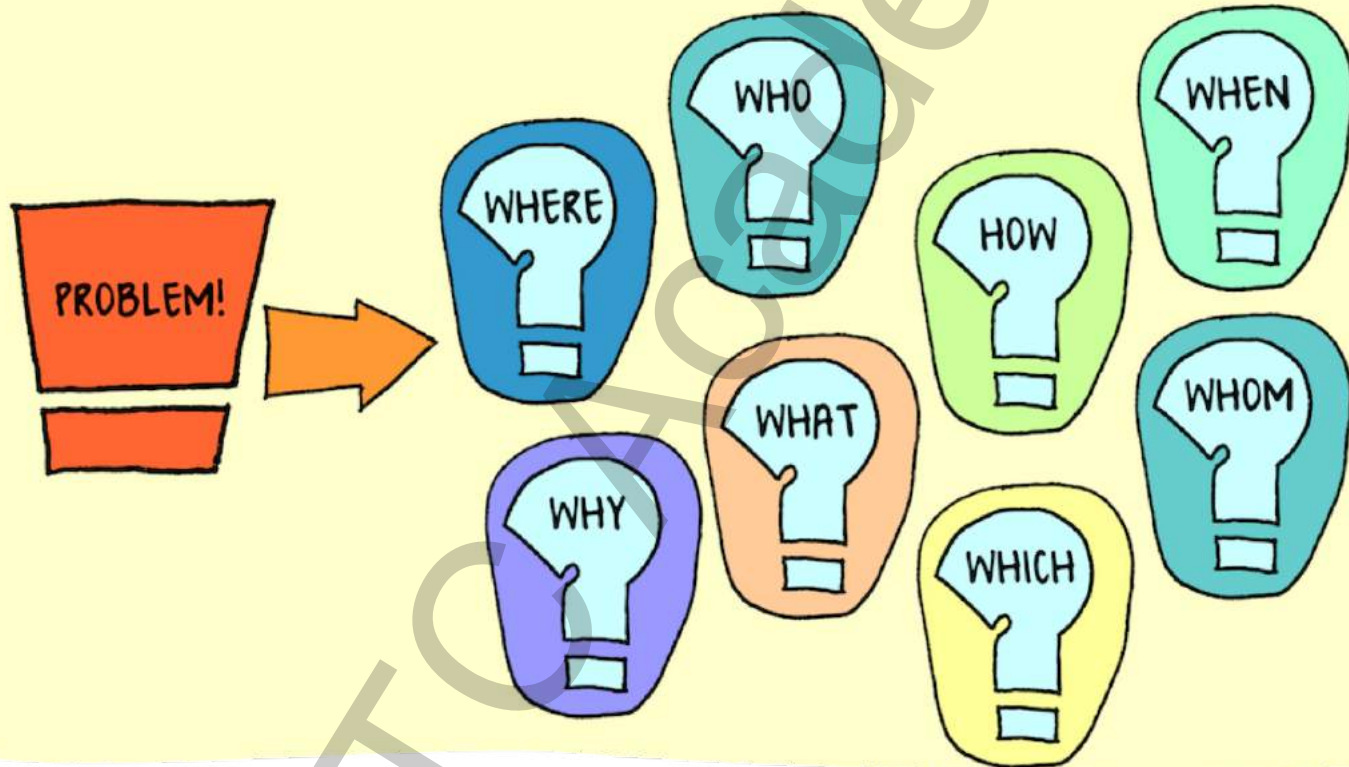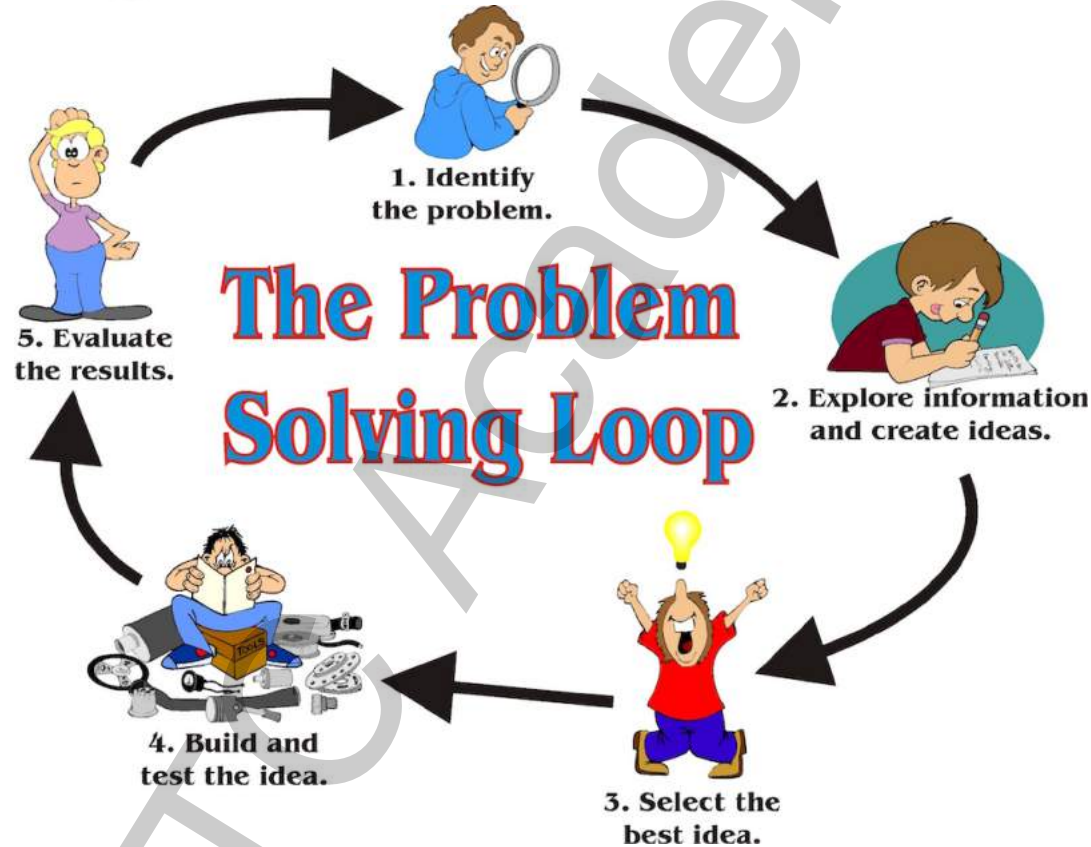
Algorithms

- An algorithm is a finite set of instructions which, if followed, accomplish a particular task and must satisfy the following criteria:

    - input: there are zero or more quantities which are externally supplied

    - output: at least one quantity is produced

    - definiteness: each instruction must be clear and unambiguous

    - finiteness: if we trace out the instructions of an algorithm, then for all cases the algorithm will terminate after a finite number of steps

    - effectiveness: it must be possible to perform each step of the algorithm correctly and in a finite amount of time

How to Analyze Programs?

- There are many criteria upon which we can judge a program:

  - Does it do what we want it to do?

  - Does it work correctly according to the original specifications of the task?

  - Is there documentation which describes how to use it and how it works?

  - Are procedures created in such a way that they perform logical sub-functions?

  - Is the code readable?

- We can use a natural language like English, although, if we select this option, we must make sure that the resulting instructions are definite

    - Text presentation: We will present most of our algorithms in C (or Pascal…), occasionally resorting to a combination of English and C (or Pascal…) for our specifications – Pseudocode

    - Graphic presentations: Flowcharts are another possibility, but they work well only if the algorithm is small and simple

Pseudocode

- It is not actual code. In computer science, pseudocode is a plain language description of the steps in an algorithm or another system.

- Based on pseudocode, programmer can use any programming language to implement an actual program.

- Example:
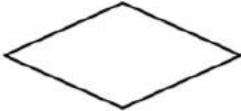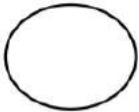
```
begin
    declare num1, num2, sum
    accept num1, num2
    compute sum = num1 + num2
    display sum
end
```
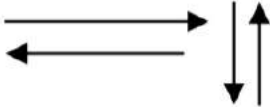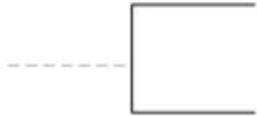
Flowchart

- A flowchart is a type of diagram that represents a workflow or process.

- A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task.

- The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows.

- This diagrammatic representation illustrates a solution model to a given problem.

Flowchart Symbols

| Symbol	Name	Description
	Process	Indicates any type of internal operation inside the Processor or Memory
	Input/Output	Used for any Input / Output (I/O) operation. Indicates that the computer is to obtain data or output results
	Decision	Used to ask a question that can be answered in a binary format (Yes/No, True/False)
	Connector	Allows the flowchart to be drawn without intersecting lines or without a reverse flow.

Flowchart

Symbol	Name	Function
	Predefined Process	Used to invoke a subroutine or an interrupt program
	Terminal	Indicates the starting or ending of the program, process, or interrupt program.
	Flow Lines	Shows direction of flow
	Annotation	It is used to provide additional information about another flowchart symbol in the form of comments or remarks

VTC Academy

- In the condition flow, boolean expression evaluates to true, then the if block will be executed, otherwise, the else block will be executed.
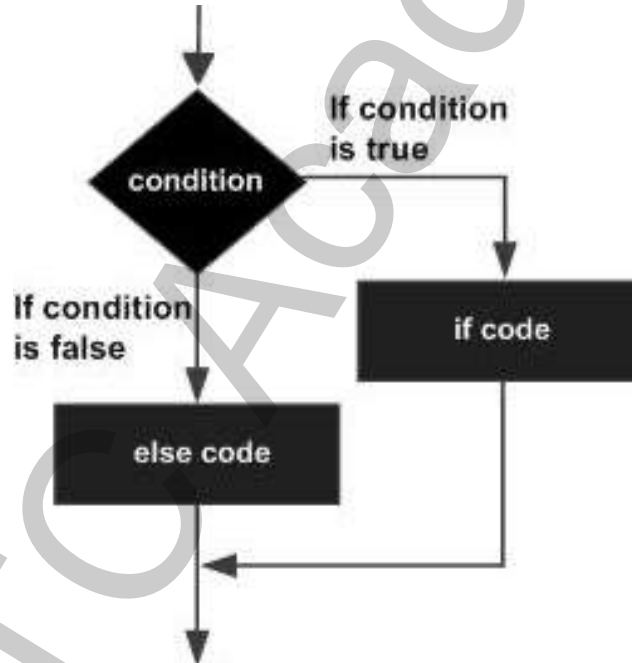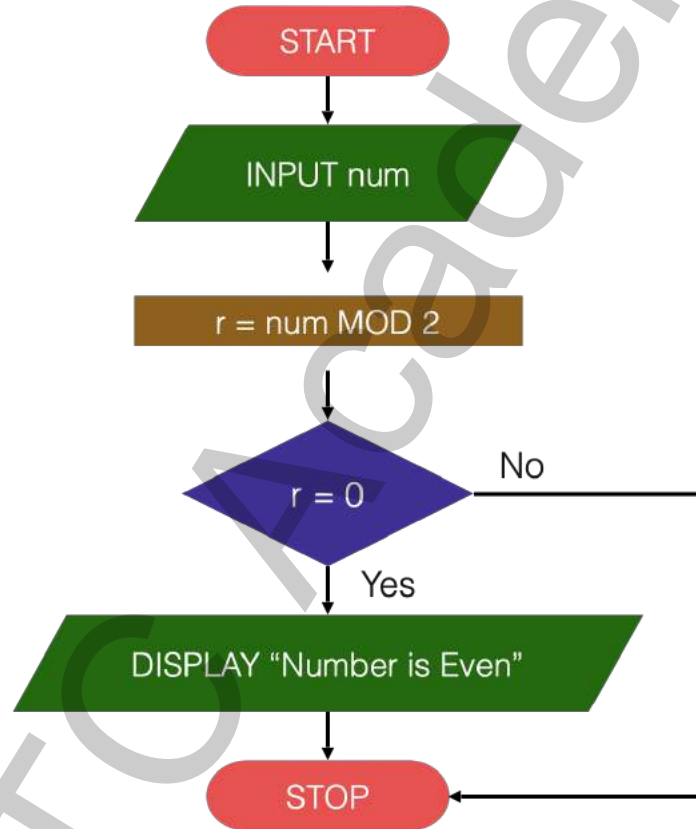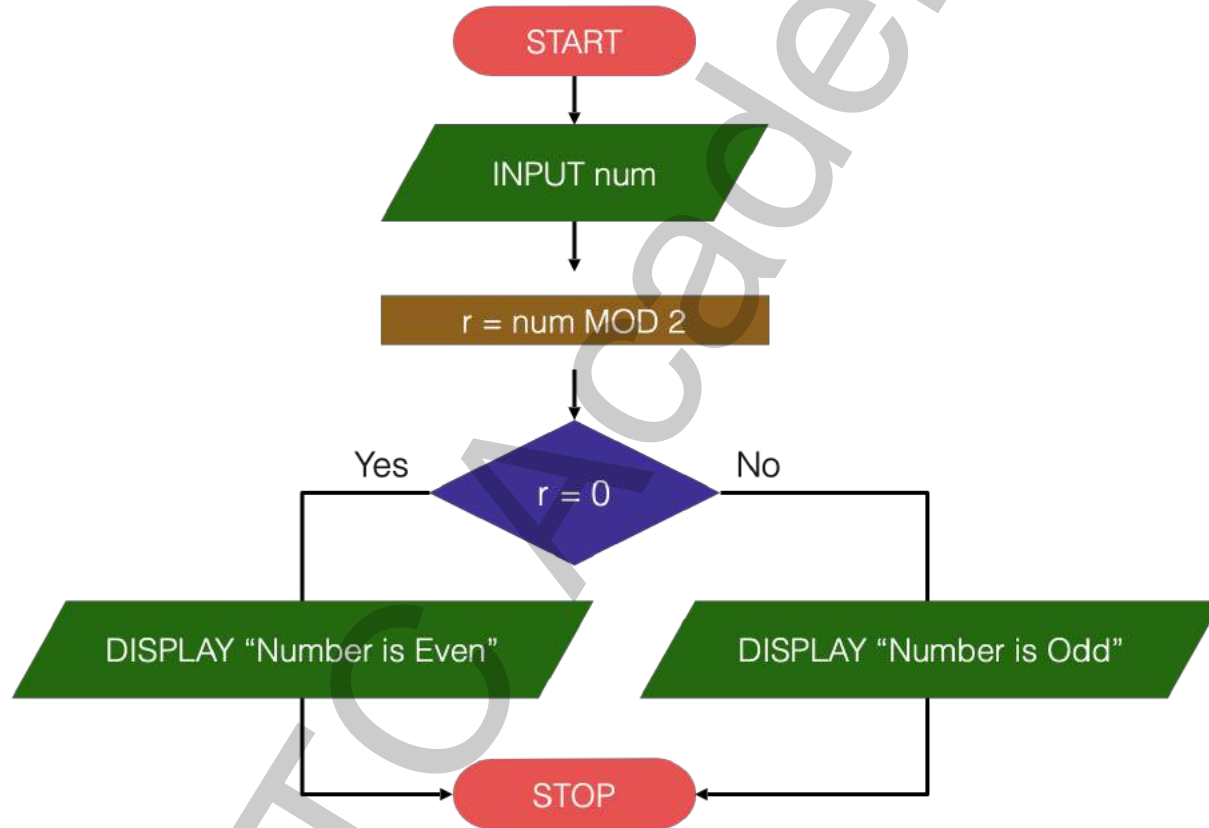
- In the condition flow, if the boolean expression evaluates to true, then the block of code inside the condition block will be executed.
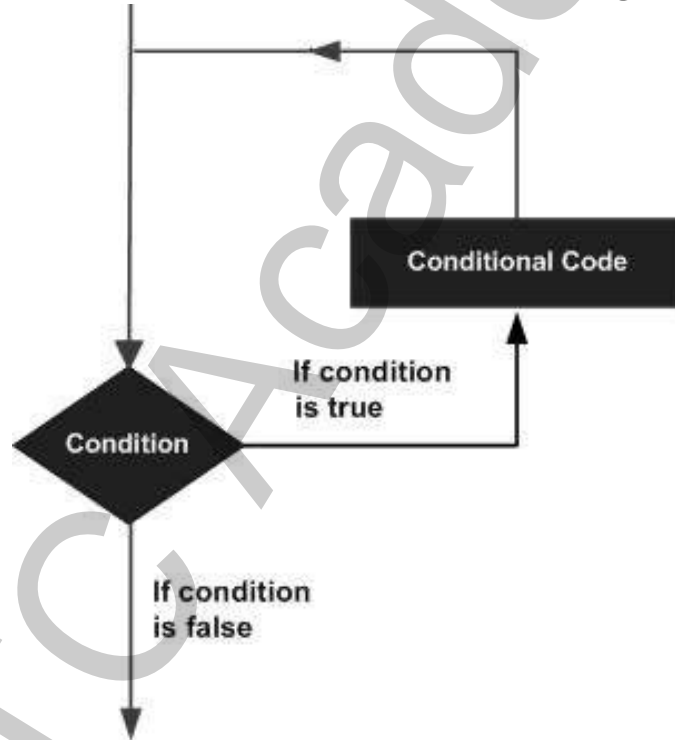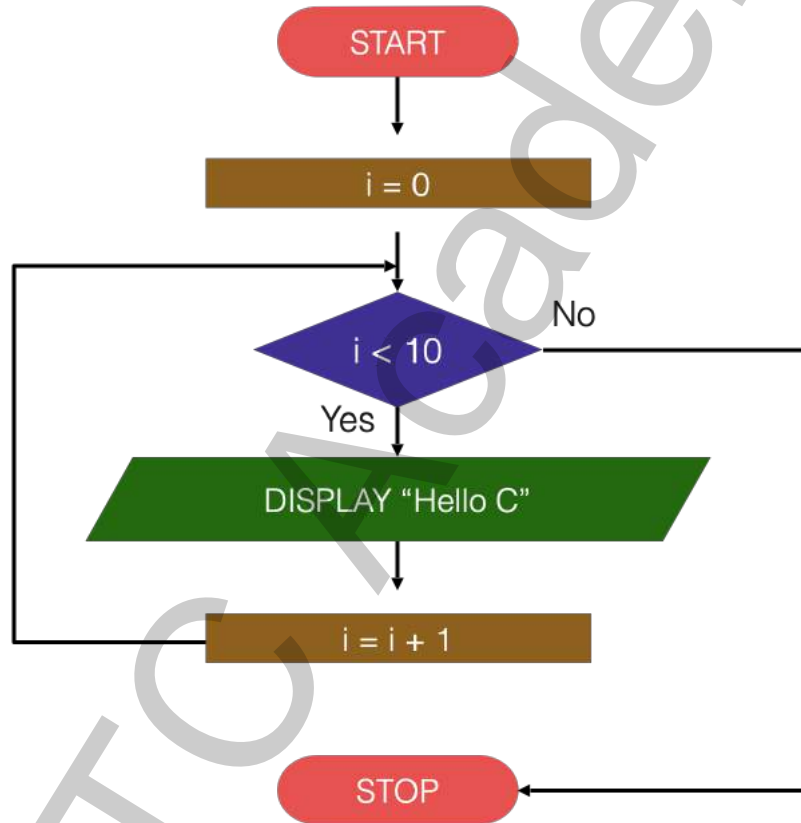
VTC Academy
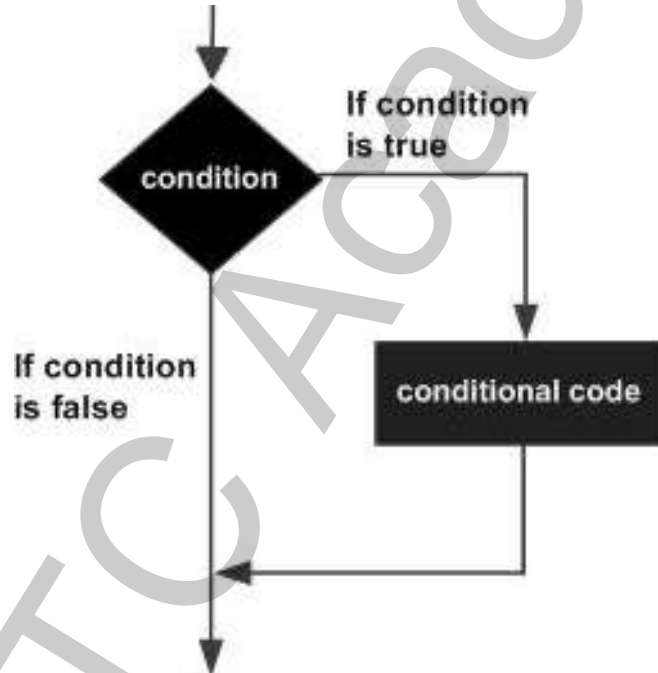
- A loop flow allows us to execute a statement or group of statements multiple times.

VTC Academy

- Introduction to C programming language

- The C program structure

- Basic syntax in C program

- Concept of problem solving and problem solving process

- An algorithm is a finite set of instructions which, if followed, accomplish a particular task

- Programmer can use flowchart to describe a algorithm

References

- Book: Programming Language, 2nd Edition - Brian W. Kernighan, Dennis M. Ritchie

- Online courses:

  - http://www.tutorialspoint.com/cprogramming/

  - http://www.cprogramming.com/tutorial/c/

  - https://ltvaz.com/category/bpl/ (Vietnamese Version)

VTC Academy
Think Ahead, Beyond Boundaries

Thank you!