WEEK 3

INTRODUCTION TO C PROGRAMMING LANGUAGE

Areas…

- Unit 1: Element of C
- Unit 2: Data Type
- Unit 3: Variables, Statements, Expressions

Unit 1

- Unit 1: Element of C
  - ✓Character Set
  - ✓Keywords
  - ✓Identifier

Character Set

- Includes the **Alphabet, Digits, Special Characters, Escape Sequences**, etc.

A,B,C,D.............Z

a,b,c,d,................z

0 1 2 3 4 5 6 7 8 9

! " # & ' ( )

* % + , - / .

; : < = > ? [

^] - _ { } |

~

Character	Escape Sequence	ASCII Value
Back Space	\b	08
Bell	\a	07
Horizontal Tab	\t	09
New Line	\n	10
Vertical Tab	\v	11
Form Feed	\f	12
Carriage Return	\r	13
Quotation Mark	\”	34
Apostrophe	\’	039
Question Mark	\?	063
Backslash	\\	092
NULL		0

Keywords/ Reserved

- These are predefined or reserved words that have special meaning to the C compiler. They are part of the syntax and cannot be used as identifiers in the program
- Below are some of these…

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

Identifier

- C Programming language has two types of words, keywords and identifiers
- Identifiers are **user defined words** and unknown to the C Compiler
- These are used to name any entity like; Functions, Variables, Arrays and Structures.. etc..
- If you are going to use an identifier, there is need to define it!

- **Identifier naming rules:**
  - ✓ Identifier name should contains only **Alphabets, Digits and Underscore** characters.
  - ✓ First character must be alphabet or underscore **not a Digit**.
  - ✓ The Name of an Identifier should not be a keyword.
  - ✓ C language is **Case Sensitive**. So upper case letter and lower case letters are different.

Unit 2

- Unit 2: Data Type
  - ✓ Data Types
  - ✓ Constants

Data Types

- Different types of datatypes store different data values.
- C Programming Language has **Three Fundamental** datatypes
  - ✓**Integer Datatype [** Denoted by **int]**
  - ✓**Character Datatype [** Denoted by **char]**
  - ✓**Float Datatype [**Denoted by **float]**
- There are other datatypes like arrays, Strings and Enums, Which are called derived datatypes.
- These will be looked at later!

Constants

- The things which can’t be changed!
- In C programming Constant is a value that cannot be changed during the execution of the program.

- C Programming support 3 categories of Constants.
  1. Numeric Constants.
     - ✓ Integer Constants **E.g. 431, 85, 65, 31....etc.**
     - ✓ Real or Floating point Constants **E.g. Fractional: 0.7, 4.31, 98.2,...etc. E.g. Exponential: +4.3e-5, 1.2e+5,...etc.**
  2. Character Constants. **E.g. 'M'**
  3. String Constants. **E.g. "Hello“**

Note:

- “M" and ‘M' are **completely different**. “M" is **string constant** and ‘M' is **character constant**.
- “M" contains two characters one is character M and another character is Compiler placed NULL Character while ‘M' contains only one character which is M

Unit 3

- Unit 3: Variables, Statements, Expressions
  - ✓Variables, Variable Declaration, Variable Initialization
  - ✓Expressions
  - ✓Statements
  - ✓Input-Output
  - ✓ASCII code

Variables, Variable Declaration & Initialization

- A variable is a name of Memory location used to store a value
- Used to store user input and save the values in the memory.
- Variable names allow us access those values from the computer memory.
- Variable values can vary/change during the Execution of program apart from constants

❑Variable Declaration

- Variables must be declared before use in the program by specifying the name and Datatype.
- Variable name should not be an operator, Separator, keyword and constant.
- When creating multiple variables on a single line, commas are used to separate them.

Syntax: datatype variable_name;

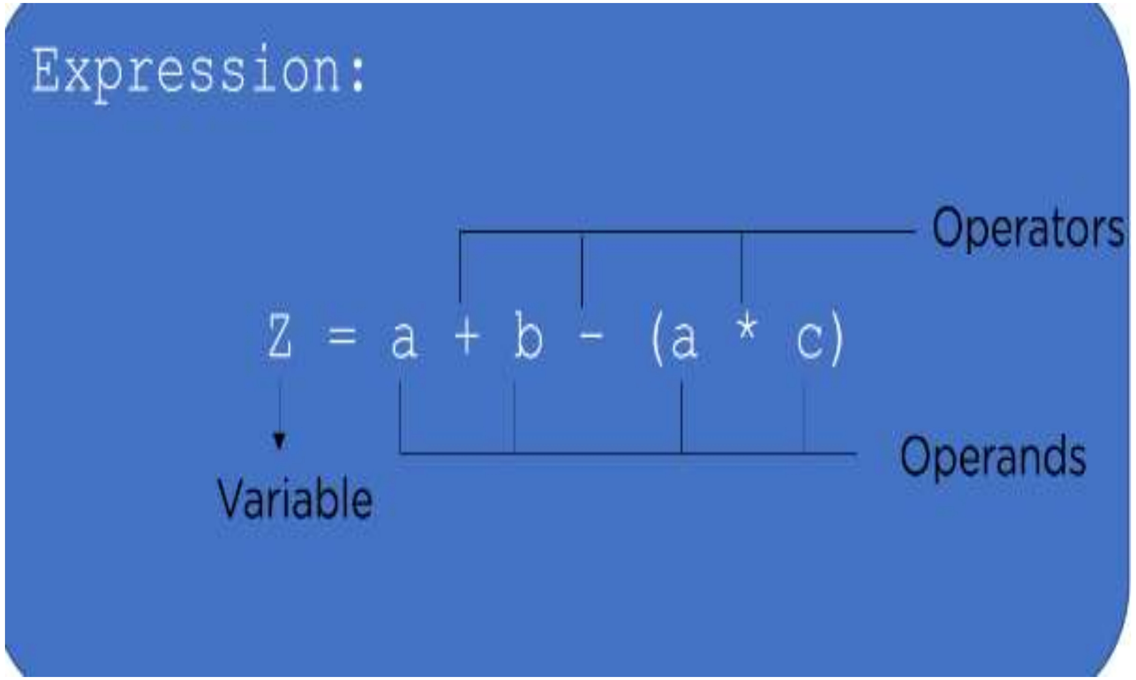
❑Variable Initialization

- We combine two steps; declaration and assignment by giving the value at the time of declaration.

Syntax: datatype variable_name = value;

Expressions

- Are the combination of variables, operands, and operators. The result would be stored in the variable once the expressions are processed based on the operator's precedence.
- Examples;
  - ✓ c=a+b
  - ✓ a-b+c
  - ✓ a+b-(a*c)



- **Types of Expressions in C**
  - ✓ Arithmetic expressions
  - ✓ Relational expressions
  - ✓ Logical expressions
  - ✓ Conditional expressions

Statements

- C programs are a collection of statements. This is an executable part of the program expected to perform some sort of action.
- **Statements Categories**
  - ✓Expression Statements (combination of variables, Constants, Operators, Function Calls and followed by a semicolon)
  - ✓Compound/Block Statements (combination of several expression statements, enclosed within braces { })
  - ✓Selection Statements (Statements are used in decision making situations)
  - ✓Iterative Statements (Also called Loops. Used when there is need to execute part of the program many times)
- Jump Statements (Unconditional statements used to transfer the control from one part of the program to another. E.g. goto, continue, break)

Output

- Output or Print Text!
- It means to display some data on screen, printer, or in any file
- In C, you can use the printf() function

- Example

```
printf("I am learning C.”);
printf(“The Result is %d”, intanswer);
printf(“The Result is %f \n”, floatanswer);
printf(“The Result is %c \n”, charanswer);
```

- **Note:**
- It does not insert a new line at the end of the output!

Input

- Used to take input from the user
- It means to feed some data into a program
- The scanf() function reads formatted input from the standard input such as keyboard.

- Example

```
scanf(“ %d”, &intanswer);
scanf(“ %f ”, &floatanswer);
scanf(“ %c ”, &charanswer);
```

ASCII code

- Do in your tutorial classes….