Unit 2

Basic Elements of C language

THE C CHARACTER SET

- C uses the uppercase letters **A** to **Z**, the lowercase letters **a** to **z**, the digits **0** to **9**, and certain special characters as building blocks to form basic program elements (e.g., constants, variables, operators, expressions, etc.).
- The special characters are listed below.

=	-	*	/	=	%	&	#
!	?	^	“	‘	~	/	\
<	>	(	)	{	}	[	]
:	;	.	,	-	Blank space		

C TOKENS

- C tokens are the basic buildings blocks in C language which are constructed together to write a C program.
- Each and every smallest individual units in a C program are known as C tokens.
- **C tokens are of six types. They are,**
  - Keywords
  - Identifiers
  - Constants
  - Strings
  - Special symbols
  - Operators

Keywords in C

- A keyword is a **reserved word**. You cannot use it as a variable name, constant name, etc. There are only 32 reserved words (keywords) in the C language.

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

Identifiers

- In C language identifiers are the names given to variables, constants, functions and user-define data. These identifier are defined against a set of rules.

Rules for an Identifier

- An Identifier can only have alphanumeric characters(a-z , A-Z , 0-9) and underscore(_).
- The first character of an identifier can only contain alphabet(a-z , A-Z) or underscore (_).
- Identifiers are also case sensitive in C. For example **name** and **Name** are two different identifiers in C.
- Keywords are not allowed to be used as Identifiers.
- No special characters, such as semicolon, period, whitespaces, slash or comma are permitted to be used in or as Identifier.

Data Types in C

- C data types are defined as the data storage format that a variable can store a data to perform a specific operation.
- Data types are used to define a variable before to use in a program.
- Size of variable, constant and array are determined by data types.

- There are four data types in C language. They are,

Types	Data Types
Basic data types	int, char, float, double
Enumeration data type	enum
Derived data type	pointer, array, structure, union
Void data type	void

BASIC DATA TYPES IN C LANGUAGE:

INTEGER DATA TYPE:

- Integer data type allows a variable to store numeric values.
- “int” keyword is used to refer integer data type.
- The storage size of int data type is 2 or 4
- It varies depend upon the processor in the CPU that we use. If we are using 16 bit processor, 2 byte (16 bit) of memory will be allocated for int data type.
- Like wise, 4 byte (32 bit) of memory for 32 bit processor is allocated for int datatype.
- int (2 byte) can store values from -32,768 to +32,767
- int (4 byte) can store values from -2,147,483,648 to +2,147,483,647.

- If you want to use the integer value that crosses the above limit, you can go for “long int” and “long long int” for which the limits are very high.
- **Note:**
  - We can’t store decimal values using int data type.
  - If we use int data type to store decimal values, decimal values will be truncated and we will get only whole number.
  - In this case, float data type can be used to store decimal values in a variable.

CHARACTER DATA TYPE:

- Character data type allows a variable to store only one character.
- Storage size of character data type is 1 byte. We can store only one character using character data type.
- “char” keyword is used to refer character data type.
- For example, ‘A’ can be stored using char datatype. You can’t store more than one character using char data type.

FLOATING POINT DATA TYPE:

- Floating point data type consists of 2 types. They are,
- float
- double

1. FLOAT:

- Float data type allows a variable to store decimal values.
- Storage size of float data type is 4 bytes. This also varies depend upon the processor in the CPU as “int” data type.
- We can use up-to 6 digits after decimal using float data type.
- For example, 10.456789 can be stored in a variable using float data type.

2. DOUBLE:

- Double data type is also same as float data type which allows up-to 10 digits after decimal.
- The range for double datatype is from 1E–37 to 1E+37.

SIZEOF() FUNCTION IN C LANGUAGE:

- sizeof() function is used to find the memory space allocated for each C data types.

```
#include <stdio.h>
void main()
{
        int a;
        char b;
        float c;
        double d;
        printf("Storage size for int data type:%d \n",sizeof(a));
        printf("Storage size for char data type:%d \n",sizeof(b));
        printf("Storage size for float data type:%d \n",sizeof(c));
        printf("Storage size for double data type:%d\n",sizeof(d));
 }
```

MODIFIERS IN C LANGUAGE:

- The amount of memory space to be allocated for a variable is derived by modifiers.
- Modifiers are prefixed with basic data types to modify (either increase or decrease) the amount of storage space allocated to a variable.
- For example, storage space for int data type is 4 byte for 32 bit processor. We can increase the range by using long int which is 8 byte. We can decrease the range by using short int which is 2 byte.
- There are 5 modifiers available in C language. They are,
  - short
  - long
  - signed
  - unsigned
  - long long

- Below table gives the detail about the storage size of each C basic data type in 16 bit processor. Please keep in mind that storage size and range for int and float datatype will vary depend on the CPU processor (8,16, 32 and 64 bit)

C Data types / storage Size	Range
char / 1	–127 to 127
int / 2	–32,767 to 32,767
float / 4	1E–37 to 1E+37 with six digits of precision
double / 8	1E–37 to 1E+37 with ten digits of precision
long double / 10	1E–37 to 1E+37 with ten digits of precision
long int / 4	–2,147,483,647 to 2,147,483,647
short int / 2	–32,767 to 32,767
unsigned short int / 2	0 to 65,535
signed short int / 2	–32,767 to 32,767
long long int / 8	–(2power(63) –1) to 2(power)63 –1
signed long int / 4	–2,147,483,647 to 2,147,483,647
unsigned long int / 4	0 to 4,294,967,295

Delimiter

- A delimiter is one or more characters that separate text strings.
  Common **delimiters** are commas (,), semicolon (;), quotes (", '), braces ({}), pipes (|), or slashes (/ \ ). When a program stores sequential or tabular data, it delimits each item of data with a predefined character.

C Variable

- C variable is a named location in a memory where a program can manipulate the data. This location is used to hold the value of the variable.
- The value of the C variable may get change in the program.
- C variable might be belonging to any of the data type like int, float, char etc.

RULES FOR NAMING C VARIABLE:

- Variable name must begin with letter or underscore.
- Variables are case sensitive
- They can be constructed with digits, letters.
- No special symbols are allowed other than underscore.
- sum, height, _value are some examples for variable name

DECLARING & INITIALIZING C VARIABLE:

- Variables should be declared in the C program before to use.
- Memory space is not allocated for a variable while declaration. It happens only on variable definition.
- Variable initialization means assigning a value to the variable.

Type	Syntax
Variable declaration	data_type variable_name; Example: int x, y, z; char flat, ch;
Variable initialization	data_type variable_name = value; Example: int x = 50, y = 30; char flag = ‘x’, ch=’l’;

TYPES OF VARIABLES IN C

- Local variable
- Global variable

Constant

- C Constants are also like normal variables. But, only difference is, their values can not be modified by the program once they are defined.
- Constants refer to fixed values. They are also called as literals
- Constants may be belonging to any of the data type.
- Syntax:

  const data_type variable_name;

TYPES OF C CONSTANT:

- Integer constants
- Real or Floating point constants
- Octal & Hexadecimal constants
- Character constants
- String constants
- Backslash character constants

Constant type	data type (Example)
Integer constants	int (53,762, -478 etc) unsigned int (5000u, 1000U etc) long int, long long int (483,647 2,147,483,680)
Real or Floating point constants	float (10.456789) double (600.123456789)
Octal constant	int (Example: 013 /*starts with 0 */)
Hexadecimal constant	int (Example: 0x90 /*starts with 0x*/)
character constants	char (Example: ‘A’, ‘B’, ‘C’)
string constants	char (Example: “ABCD”, “Hi”)

INTEGER CONSTANTS IN C:

- An integer is a numeric constant (associated with number) without any fractional or exponential part
- An integer constant must have at least one digit.
- It must not have a decimal point.
- It can either be positive or negative.
- No commas or blanks are allowed within an integer constant.
- If no sign precedes an integer constant, it is assumed to be positive.
- The allowable range for integer constants is -32768 to 32767.
- There are three types of integer constants in C programming:
  - decimal constant(base 10)
  - octal constant(base 8)
  - hexadecimal constant(base 16)

- For example
  - Decimal constants: 0, -9, 22 etc
  - Octal constants: 021, 077, 033 etc
  - Hexadecimal constants: 0x7f, 0x2a, 0x521 etc
  - Binary constants: 0b1010, 0b11 etc

FLOATING/REAL CONSTANTS IN C:

- A floating point constant is a numeric constant that has either a fractional form or an exponent form
- A real constant must have at least one digit
- It must have a decimal point
- It could be either positive or negative
- If no sign precedes an integer constant, it is assumed to be positive.
- No commas or blanks are allowed within a real constant.
- For example
  - 2.0
  - 0.0000234
  - -0.22E-5

CHARACTER AND STRING CONSTANTS IN C:

- A character constant is created by enclosing a single character inside single quotation marks.
- A character constant is a single alphabet, a single digit or a single special symbol enclosed within single quotes.
- The maximum length of a character constant is 1 character.
- String constants are enclosed within double quotes.
- For example
  - 'a', 'm', 'F', '2', '}' etc;

BACKSLASH CHARACTER CONSTANTS IN C: (ESCAPE SEQUENCE)

- There are some characters which have special meaning in C language.
- They should be preceded by backslash symbol to make use of special function of them.
- Given below is the list of special characters and their purpose.

Backslash character	Meaning
\b	Backspace
\f	Form feed
\n	New line
\r	Carriage return
\t	Horizontal tab
\”	Double quote
\’	Single quote
\\	Backslash
\v	Vertical tab
\a	Alert or bell
\?	Question mark
\N	Octal constant (N is an octal constant)
\XN	Hexadecimal constant (N – hex.dcml cnst)

Symbolic Constant

- We can define symbolic constants in a C program in the following ways.
- By “const” keyword
- By “#define” preprocessor directive

- For e.g. const float PI=3.1416;
- For e.g. #define PI 3.1416

Expression

- In programming, an expression is any legal combination of symbols that represents a value.
- C Programming Provides its own rules of Expression, whether it is legal expression or illegal expression. For example, in the C language x+5 is a legal expression.
- Every expression consists of at least one operand and can have one or more operators.
- Operands are values and Operators are symbols that represent particular actions.

Expressions	Validity
a + b	Expression is valid since it contain + operator which is binary operator
+ + a + b	Invalid Expression

Expressions	Values
-4+6	2
C=2+12	14
9>4	1(true)
4+(a=2+7)	13

- Simple expression
  - 30, PI, ‘a’, area
- Complex expression
  - a+b*c-d/e
- Logical expression
  - a>b
  - x<=y
  - a==b

Format specifier	Description
%c	Character
%d	Integer
%e or %E	Scientific notation of float values
%f	Floating point
%i	Signed Integer (same as %d)
%l or %ld or %li	Signed Integer
%lf	Floating point
%Lf	Floating point
%lu	Unsigned integer
%lli, %lld	Signed Integer
%o	Octal representation of Integer.
%p	Address of pointer to void void *
%s	String
%u	Unsigned Integer
%x or %X	Hexadecimal representation of Unsigned Integer
%n	Prints nothing
%%	Prints % character

Comment in C

- In programming, comments are hints that a programmer can add to make their code easier to read and understand
- **Types of Comments**
- There are two ways to add comments in C:
  1. **//** - ---Single Line Comment
  2. **/*...*/** - ---Multi-line Comment