

Jay Shree Krishna



RUPAREL
EDUCATION PVT. LTD.
CIN : U80300GJ2013PTC075575

**WELCOME TO
RUPAREL EDUCATION PVT. LTD.**

201,Punit Shopping Centre

Ranavav Chowk

Junagadh.

Mobile No.76000 44051.

Introduction to Programming



RUPAREL
EDUCATION PVT. LTD.
CIN : U80300GJ2013PTC075575

- Programming is one of the sequences that define a task which is already predefined in a proper or in a logical area.
- Programming is nothing but provide a simplest way to understand a complex task.
- Now a days there are number of programming languages to develop in various programming task.
- Programming provides a step by step execution or a method to reduce human efforts.
- Programming provides a speed and accuracy in the work because it follows the instruction which is already mentioned in a program compiler or interpreter.
- There are numbers of various programming languages like C – Language, C++, COBOL, Pascal etc...

Various Computer Languages



RUPAREL
EDUCATION PVT. LTD.
CIN : U80300GJ2013PTC075575

- Computer languages like a natural language it's means gujarati, English, hindi etc...
- So, the computer language must be very precise they have some few relatives, rules for composition and strictly control in a word that must be define before use it.
- Computer languages can be divided into two groups.
 - 1) High level language
 - 2) Low level language
 - **High level language:**
High level language designed to be easier to use more abstract, more portable than low level language.



- Most of the modern software are written in high level languages.
Which is compile into object code and than translate into machine instruction.
- **Low level language:**
While low level language has some restriction in computer to high level language.
It is not so different in higher programming concept as well as in a complex application.
It is not portable or abstract in compare to high level language.

History & Overview of C Language



RUPAREL
EDUCATION PVT. LTD.
CIN : U80300GJ2013PTC075575

- The C programming language is a structure oriented programming language, developed at Bell Laboratories in 1972 by Dennis Ritchie
- C programming language features were derived from an earlier language called “B” (Basic Combined Programming Language – BCPL)
- C language was invented for implementing UNIX operating system In 1978, Dennis Ritchie and Brian Kernighan published the first edition “The C Programming Language” and commonly known as K&R C
- In 1983, the American National Standards Institute (ANSI) established a committee to provide a modern, comprehensive definition of C.
- The resulting definition, the ANSI standard, or “ANSI C”, was completed late 1988.

Difference between traditional and modern C – Language.



RUPAREL
EDUCATION PVT. LTD.
CIN : U80300GJ2013PTC075575

- C – Language is popular because it provides various functionality as well as it also provides in built library function and also you can create your own user defined function.
- In a traditional C – Language function declarations are assumed by default to be a type of integer.
- While in a modern C – Language you need to declare a complete function prototype, because compiler does not work automatically. So it requires a complete function declaration process.
- The type signed characters is not available in a traditional C – Language while it is available in modern C – Language.



- In a traditional C unary positive sign are not allowed but in a modern C it allow unary positive as well as unary negative also.
- The data type long is not available in traditional C – Language but in a modern C it allow to work.
- The const[constant] identifier is not available in a traditional C – Language. While modern C – Language allow it.
- The type enumeration is not available in traditional C While in a modern C – Language it allow.

C character Set



RUPAREL
EDUCATION PVT. LTD.
CIN : U80300GJ2013PTC075575

- C – language allow a many characters and symbols that is divided into various categories based on their functionality. That is defined below:

- **Letters**

letters contain alphabets that may be either in upper case A to Z or a lower case a to z.

- **Digits**

Digits contain all digits like 0 to 9.

- **Special Characters**

Special characters use for special purposes the list of special character are

- | | | |
|--------------|-------------------|-------------------------------|
| 1) , (comma) | 2) \ (back slash) | 3) + (plus) |
| 4) . (dot) | 5) ~ (tiled) | 6) < (opening angular braces) |



- | | | |
|-----------------------|--------------------|--------------------------------|
| 7) ; (semi colon) | 8) _ (underscore) | 9) > (closing angular bracket) |
| 10) : (colon) | 10) \$ (dollar) | 11) ((left parenthesis) |
| 12) ? (Question mark) | 13) % (percentage) | 14)) (right parenthesis) |
| 15) ' (single quote) | 16) & (ampersand) | 17) { (left curly bracket) |
| 18) " (double quote) | 19) ^ (caret) | 20) } (right curly bracket) |
| 21) (pipe) | 22) * (asterisk) | 23) # (hash) |
| 24) / (slash) | 25) – (minus) | |

White space.

Blank space, horizontal tab, carriage return, new line, form feed etc are known as white space characters.

C – Tokens



RUPAREL
EDUCATION PVT. LTD.
CIN : U80300GJ2013PTC075575

- C programming is made of various keywords, operators, special symbols and so on. They are called tokens.
- Any C language program cannot be created without help of one or more token.
- We can take token as a basic building block for C language.
- As like block any construction of building is not possible same as a C program is not possible without token. Following is a list of different token is available in C-Language.
 - Keywords ✓
 - Constants ✓
 - Identifiers ✓
 - Strings ✓
 - Special Symbol ✓
 - Operators. ✓



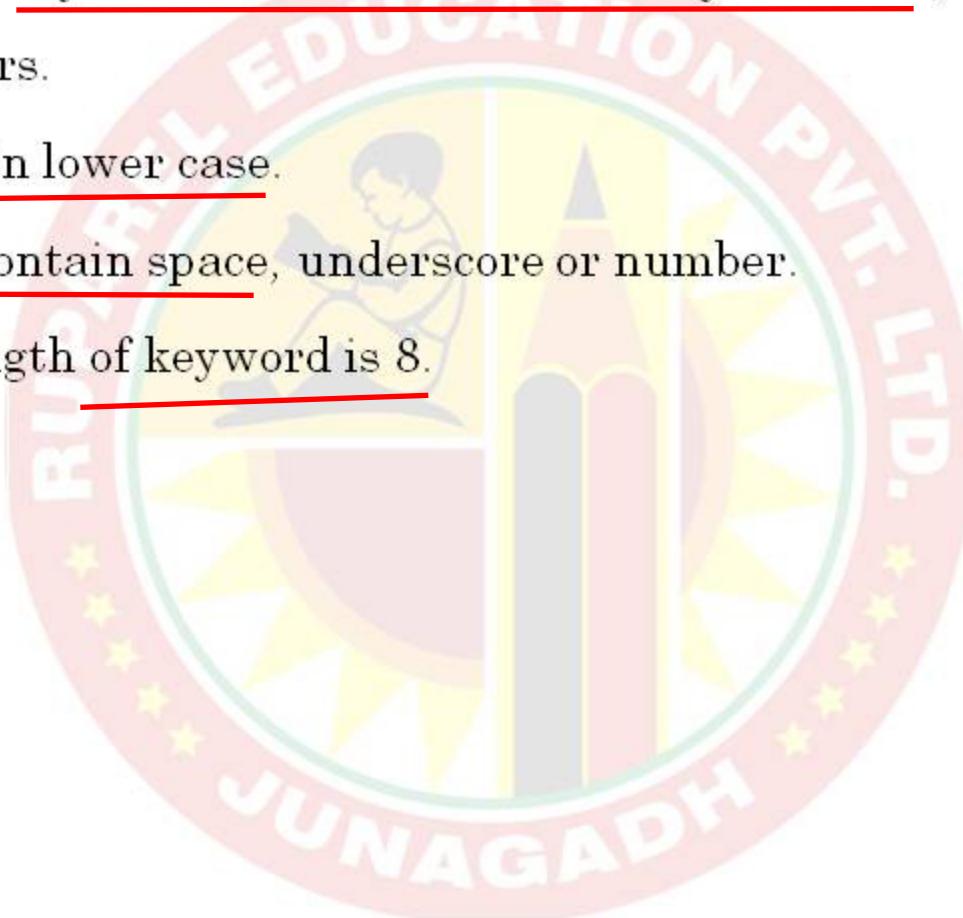
- **Keywords:**

- There are 32 keywords in C – Language
- Keywords are special words which have specific meaning.
- The keywords are the first things without it program cannot be created.
- We can say that keyword is first building block entire C Program depends.
- Following is the list of keyword.

- | | | | |
|-------------|-------------|--------------|------------|
| 1) auto | 2) break | 3) case | 4) char |
| 5) const | 6) continue | 7) default | 8) do |
| 9) double | 10) else | 11) enum | 12) extern |
| 13) float | 14) for | 15) goto | 16) if |
| 17) int | 18) long | 19) register | 20) return |
| 21) short | 22) switch | 23) signed | 24) sizeof |
| 25) static | 26) struct | 27) typedef | 28) union |
| 29) unsined | 30) void | 31) volatile | 32) while |



- The name of the keyword cannot be use for any variable, constants, structure, function, identifiers.
- All keywords are in lower case.
- No any keyword contain space, underscore or number.
- The maximum length of keyword is 8.





- **Identifiers:**

Identifier is name of variable, constants, structure, union, array or function given by the programmer.

- Following is a rules of naming identifiers.

- The name of the identifier can contain letters, digits and underscore.
- It must start with an alphabet or even in some case it can start with underscore but cannot start with digit.
- The name of identifier cannot be the name of any keyword.
- The name of identifier can contain maximum 31 characters without space, comma and any other special symbol are not allowed in the name of identifier.
- C – Language is case sensitive so uppercase and lowercase is a different.

- **Constants:**

Constant is a numeric value used in a program.

Constant is the value that remain same during the execution of the program.

there are mainly two types of constant.

1. Numeric Constant ✓
2. Character Constant ✓

- **Numeric Constant:**

Numeric constant also categorised in a two different constant that is integer and real constant.

Integer is a constant of sequence of digit while real constant is also a sequence of digits with decimal point.



- **Character Constant:**

Character constant can also divide in a two part.

- **Single character constant:**

- Single character constant can have only one character. So it enclosed with one single quote. For example ‘A’

- **String constant:**

- String is a collection of character. It means it contain more than one characters as a string and it enclosed in a double quote. For example “C - Language”.

- **String:**

- C – Language does not supports the concept of string then after it allow user to input a sequence of character to define as a string.
- C – Language provide a number of functions which allow user to interact with string.
- generally, strings are defined in a double inverted comma and behave like one complete word with specific meaning.
- String also contain space, digit and special symbol inside it for example
“C – Language”
“A-123”
“@anc_123”
- According to the above example we can use string in different purposes as per the program requirement.



- Special Symbols:
 - C Language is procedure oriented programming language.
 - So it can contain number of symbols for different types of execution like.
 - () is used for function declaration
 - {} is used for block declaration.
 - , is used for separation of the value or a variable.
 - ; is used for to indicate that the statement is over.
 - etc... are special character are used in a C – Language.



- **Operators:**
- An operator is a symbol that cause a specific mathematical or logical manipulation to be perform.
- Operators are use in a C program to manipulate data and value.
- C – Language is having 8 different type of operators which can be use for different purpose.
 1. Arithmetic operator
 2. Relational operator
 3. Logical operator
 4. Assignment operator
 5. Increment & Decrement operator
 6. Conditional operator
 7. Bitwise operator
 8. Special operator



- **Arithmetic operator:**

- Arithmetic operator contains integers floating point, double etc.
- Number can be added, subtracted, divided or manipulated.
- Following are a list of Arithmetic operators:
 - + (plus) Ex. $c = a + b$
 - - (minus) Ex. $c = a - b$
 - * (multiply) Ex. $c = a * b$
 - / (divide) Ex. $c = a / b$
 - % (modular) Ex. $c = a \% b$



- **Relational Operator:**

- The relational and equality operators are used for test or compare the value between two operands.
- It produced the integer result to express the condition.
- If the condition is true, then return 1 (True) otherwise it returns 0 (False).
- Following are the list of relational operators:
 - < (less than) Ex. $a < b$
 - > (greater than) Ex. $a > b$
 - <= (less than equal to) Ex. $a \leq b$
 - >= (greater than equal to) Ex. $a \geq b$
 - == (equal to) Ex. $a == b$
 - != (not equal to) Ex. $a != b$



- **Logical Operator:**

- Logical operators are used to combine two or more relations.
- Logical operators are also called Boolean operators.
- Which written either true or false.
- Following are the list of operators:
 - && and Ex. $a > b \ \&\& \ a > c$
 - || or Ex. $a > b \ | \ | \ a > c$
 - ! not Ex. $\!a > b$



- **Assignment Operator:**

- The assignment operators / expression in a C – Language is used to assigned a value to the variable.
- Following is assignment operator:
- = (equal to)
- According to above equal to operator is assigned a value to the variable.
- e. g. i=1
- It means value of 1 is assigned to variable i.



- **Increment & Decrement Operator:**
- In a C – Language it is one of the most important operator that are use frequently in a program for increment or decrement in value.
- It is always increment in a value by 1 or decrement value by 1.
- Following is increment and decrement operator:
- `++` (Increment operator)
- `--` (Decrement operator)
- According to the above operators we can represent it in a different two ways:
- pre-increment(`++i`), post-increment(`i++`)
- pre-decrement(`--i`), post-decrement(`i--`)



- Conditional Operator:
 - Conditional operator is special operator which is evaluated in a single line.
 - It is also known as ternary operator.
 - Because it contain two operator and three operands.
 - The conditional operator in C – Language is ? :
 - It is use for checking the condition and give output according to the condition.
 - Following is a syntax of conditional operator are:
 - (Condition) ? True : False;
 - According to above syntax conditional contain three part:
 1. Condition
 2. True part
 3. False part
 - and last it is terminated by semicolon.



- **Bit Wise Operator:**

- C – Language support a different bit operators for bit manipulation.
- It store one or more bytes such as char, int, double etc.
- The operators that are used to perform bit manipulation are called bit operator.
- Following is some list of bit operator:
 1. & Bit And
 2. | Bit Or
 3. ^ Exclusive Or / Bit Caret.
- According to above operator are used for bit manipulation.



- Special Operator:
 - C – Language support a numbers of special operators.
 - They are use for some special purposes.
 - In a C – Language following is a list of special operator:
 - , (comma)
 - sizeof
 - & and * (pointer operator)
 - . and → (member access operator)

Operators and Hierarchy of Operators



- C – Language supports a numbers of operators and it also provides its priority.
- It means each operator have their own priority.
- It is work based on their priority.
- It means any arithmetic calculation also work according to their priority.
- Each operator have priority according to its work. For example:
- () Parenthesis has high priority.
- * and / Has Second Priority
- + and - Has Last Priority
- In operators it is work based on its priority.
- It means first evaluated multiply then after divide next addition and last subtraction.
- If any parenthesis are there then it always executed first.



For Ex. A = 5, B = 10, C = 2, D = 15, E = 3

$$A + \boxed{B * C} - D / E$$

1

$$A + \boxed{B * C} - D / E$$

1



$$A + \boxed{B * C} - D / E$$

1

$$A + \boxed{B * C} - D / E$$

1

4

$$A + B * C - D / E$$

$$5 + 10 * 2 - 15 / 3$$

$$5 + 20 - 15 / 3$$

$$5 + 20 - 5$$

$$25 - 5$$

$$20$$

Data Type in C.



- Data type is symbol that represents the type of the data which we want to require in a program.
- We have to assume a proper data type to get exact value in a program.
- C – Language support a various different types of data type that are listed below:
 - 1) Integer (int) ✓
 - 2) character (char) ✓
 - 3) float ✓
 - 4) double ✓
 - 5) void ✓
 - 6) enumeration (enum)
 - (enum is a one type of user defined data type which we can store anything)



- **Integer (int):**
 - It is used to store a whole number.
 - It means it does not contain any precision value.
 - It requires 2 bytes in a program.
 - The identifier of int type data is “%d”.
 - The range of integer type data is -32767 to 32768.
 - For Example: int a;
 - According to above example a is a variable of type integer.
- **Character (char):**
 - It is used to store a single character.
 - It does not contain more than one character.
 - It requires 1 bytes memory in a program.
 - The identifier of char type data is “%c”.
 - For Example: char a;
 - According to above example a is a variable of type char.



- **Float (float):**

- It is used to store a real number.
- It means it allow precision value in it.
- It requires 4 bytes memory in a program.
- The identifier of float type data is “%f”.
- The range of float type data is -34e-38 to 34e+38.
- For Example: float a;
- According to above example a is a variable of type float.

- **Double:**

- It is used to store a real number.
- It means it allow precision value in it.
- It requires 8 bytes memory in a program.
- The identifier of double type data is “%lf”.
- The range of double type data is -1.7e-108 to 1.7e+108.
- For Example: double a;
- According to above example a is a variable of type double.



- **Void:**

- Void is a special data type in C – Language.
- Because it is used with a function that may be either a primitive type or non-primitive type function.
- The actual meaning of the void is “No”.
- It is used to specify the return type of the function.
- It is also used to specify the arguments of the function.
- If function does not contain any arguments then uses void as an argument same as if the function does not contain any return type then use void.

For Example:

```
void main(void)
{
}
```

- According to above example main is the name of the function which does not require any argument.
 - As well as does not return a value.
- The entire code are enclosed between two curly brackets

Type Casting and Type Conversion.



RUPAREL
EDUCATION PVT. LTD.
CIN : U80300GJ2013PTC075575

- Type casting or Type conversion refer, to change and entity of one data type to another.
- This is done to take advantage of certain features of type hierarchy.
- For example sometimes integer value can be represented as a floating point precision value then we have use the concept of type casting.
- There are two types of conversion:
- **1) Implicit Type Casting:**
- Implicit type casting means it convert value in hierarchy automatically without any external information.
- It is possible when mixed type of data values are available in an expression.



- **2) Explicit Type Casting:**
- Explicit type casting means the conversion of value from one type to another type cannot be achieved automatically.
- It means here we have to give some additional information regarding the type conversion.
- But remember / note that you have to convert any data value in there higher priority / presidency.

Example

- According to above example here we try to display a first ten number using loop and also want to print total and average of it.
- But at the time of calculation of average compiler written their answer in integer.
- We require answer in a float, so here we use explicit type conversion to achieve the require output.

Pre – Processors in C – Language



- Pre – Processors is also known as pre – processor directive.
- It is a unique feature in a C – Language.
- Because it provide in a program to read, modify or portability in a program.
- Pre – processor is a program that process the code before it pass through the compiler.
- Pre – Processor directive follow the special syntax rules and begin with the symbol (#) hash and do not require any semicolon(;).
- The C pre – processor is not the part of compiler but it's a separate step in the compilation process.
- Pre – Processor is also called “Macro”(Recording process).
- C – Language support a numbers of pre – processors directives. That are the listed below:
 - 1) #define
 - 2) #undef
 - 3) #include
 - 4) #if
 - 5) #else
 - 6) #ifdef
 - 7) #endif
- etc...



- Are a pre – processor supported by a C – Language.
- Following is an example of pre – processor directives.
- For example: #include<stdio.h>
- According to above example “#” is a symbol of pre – processor.
- Include is a pre – processor directive.
- <stdio.h> is a file that want to include.
- .h denotes that it is header file.
- Same as we can also use another pre – processor directive like #define PI 3.14.
- According to above again “#” is a pre – processor symbol and define is a directive where we assign a value of PI is 3.14.
- The value of PI is remain same in entire program.

Introduction Of Logical Development Tools.



RUPAREL
EDUCATION PVT. LTD.
CIN : U80300GJ2013PTC075575

- Logic is a technical way of investigation of unknown truth.
- It is a process that implements in a mind and expose in a many different ways.
- There are many reasons that computer programmer required logic to achieve a task.
- There are number of development tools which help to develop logic to implement a complex programming.
- The logical frame work help in a project, design, implementation and evaluation.
- This means that logical development tools use for a develop project cycle.
 - 1) flow chart ✓
 - 2) algorithm ✓
 - 3) dry run ✓



- **Flow chart:**

- Flow chart means visual representation of flow of data or information or processing system.
- Flow chart is a diagram that provide a sequence of information to be perform one after another(one by one).
- Flow chart provides a facility to communicate between a programmer or a business people.
- Once a flow chart is draw it became easy to write a program in any high level language.
- Flow chart is must for better documentation of complex program.
- Following is some standard symbol that represent a flow chart diagram.

1) Start / End



2) Processing Function



3) Input / Output



4) Condition / Decision Making/ Branching



To joint two programs connector



5) Flow Line - Horizontal and Vertical



According to Diagram we can represent any high level or complex programming.



- **Algorithm:**

- Algorithm is descriptive information.
- To make a computer do anything you have to right computer program and for that you have to tell computer information the computer step by step.
- It means computer algorithm is a basic technique to get job done.
- Each algorithm has different cost and different travel time.
- So sometimes algorithm create a complexity on the basis of time and space.
- So, algorithm is one the basic task of accomplish complex program and get required output.



- **Dry Run:**

- Dry run is a testing process where the effect of possible failure are initially maintain.
- Dry run is mentally process to run a computer program.
- Where the computer programmer check the source code one by one steps and determine what it will do when run.
- We can say that run is a mental run of an algorithm.
- The dry run is a process that you test your program without using computer.
- So, the dry run is necessary for error checking, program solution, program maintenance and for logic development.