



# EMBEDDED C LANGUAGE

DR SARWAN SINGH

NIELIT CHANDIGARH

SARWAN@NIELIT.GOV.IN

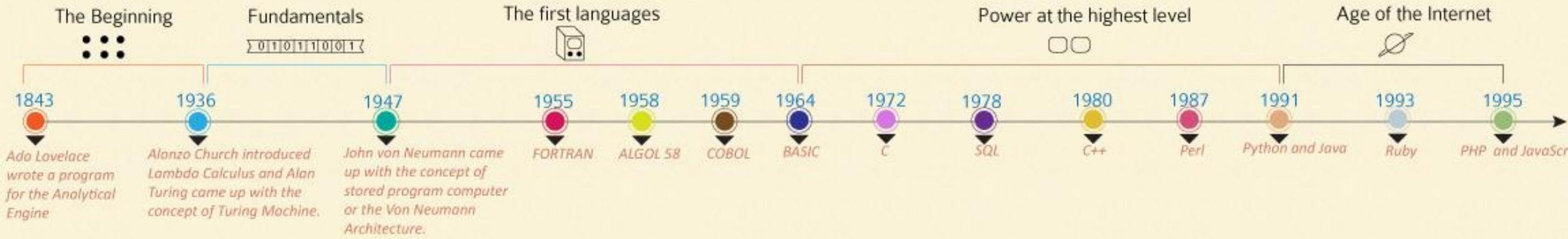
# AGENDA

- Need of computing languages
- C language – history, uses, applications
- C language- data type, operators
- Flow and control statements
- Functions
- Header files

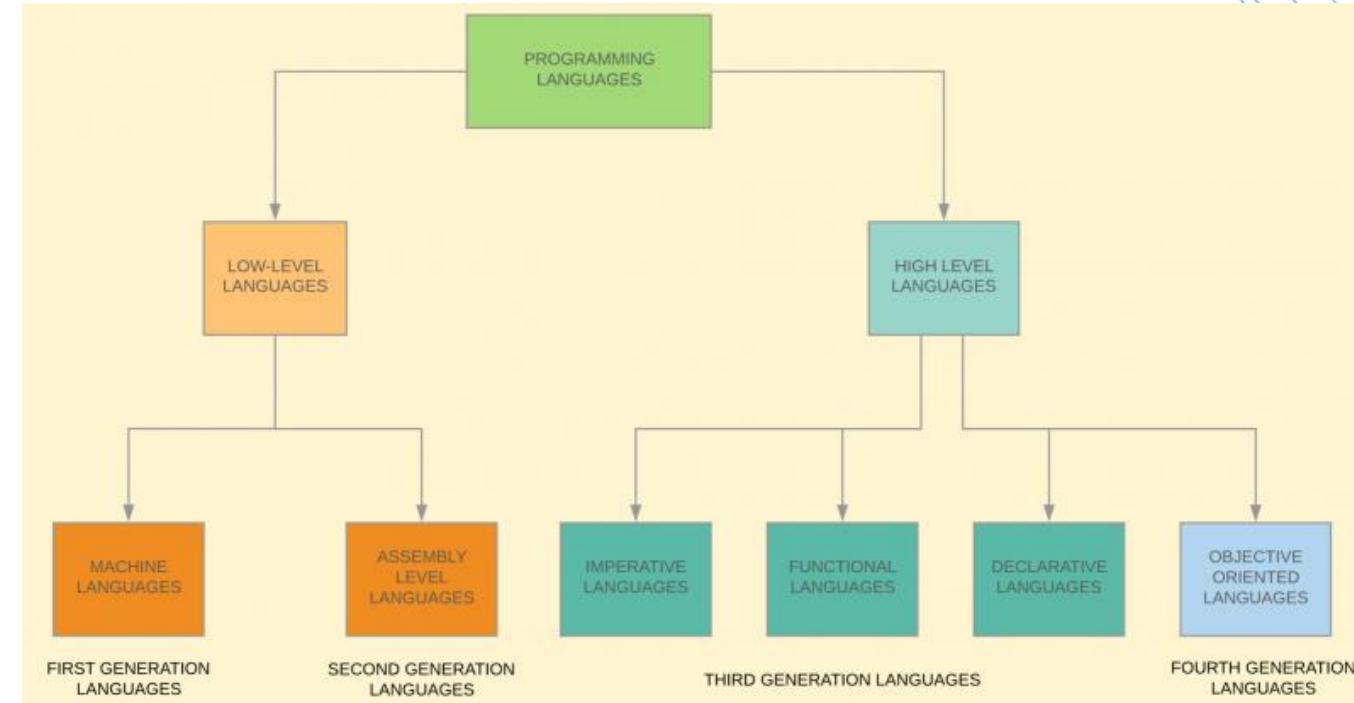


# WORLD OF PROGRAMMING

## Timeline



# CLASSIFICATION OF PROGRAMMING LANGUAGES



## Language Paradigm

### Imperative Paradigm

#### Procedural

C, Pascal

#### Object Oriented

C++, Java

### Declarative Paradigm

#### functional

LISP

#### logical

Prolog

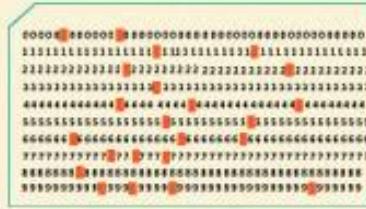
# THE WORLD OF PROGRAMMING

- Pioneers
- Facts & Algos
- Stats

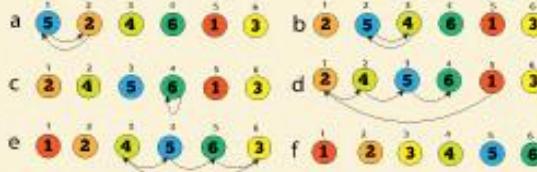


## Charles Babbage

{He first came up with the idea of difference engine & analytical engine and is regarded as father of computer}



// A Punch Card



// Insertion Sort algorithm



## Dennis Ritchie

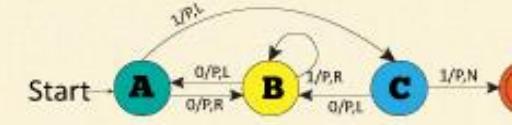
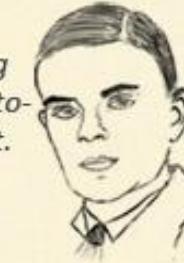
{He is the creator of C programming language and was also amongst the key developers of UNIX operating system. Turing award in 1983}



// Dijkstra's algorithm

## Alan Turing

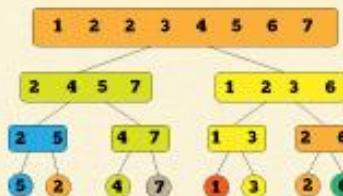
{He is well known for the Halting problem, Turing machines, crypto-analysis of Enigma & Turing test. Turing award is given annually for exceptional work in the field of computing}



// 3 state busy beaver machine

## Ada Lovelace

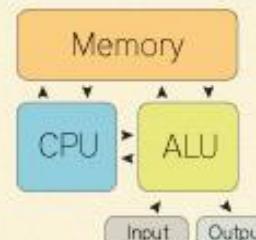
{Wrote a program to calculate sequence of Bernoulli's number using analytical engine. Regarded as World's first programmer}



// Merge sort algorithm

## John von Neumann

{He came up with the concept of stored program computer that uses a CPU and a separate storage to hold both instructions and data. This is also known as von Neumann architecture}



5

## Edsger Dijkstra

{He is known for Dijkstra's algorithm, which is a graph search algorithm that solves the single-source shortest path problem for a graph with nonnegative edge

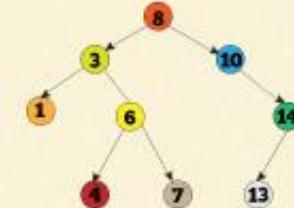


// Von Neumann Architecture



### Donald Knuth

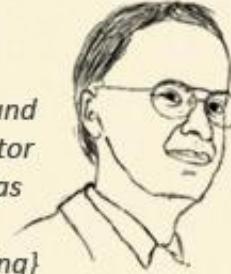
{He is the creator of TEX and MMIX and is well known for the "Art of computer programming" book series. He received the Turing Award in 1974}



// Binary Search Tree

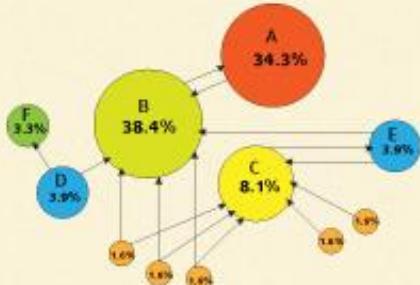
### John Backus

{He is well known for the development of FORTRAN and ALGOL. He is also the inventor of Backus-Naur form and has also helped to popularize functional level programming}

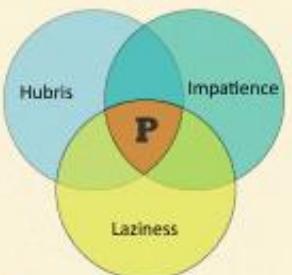


### Ken Thompson

{He is well known as the principal creator of the UNIX operating system and is also the co-creator of the Go programming language}



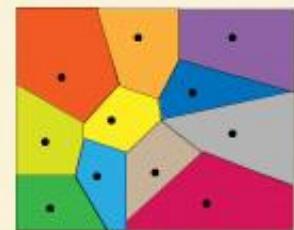
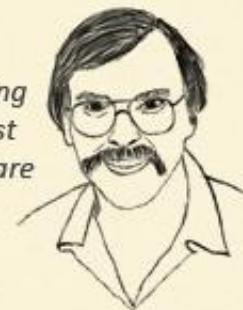
// Google PageRank Algorithm



// 3 virtues of a Programmer - Impatience, Laziness and Hubris  
SARWAN@NIELIT

### Larry Wall

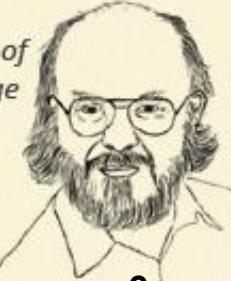
{He is well known for the creation of Perl programming language and is also the first recipient of the Free Software Foundation Award for the Advancement of Free Software}



// Sweep line algorithms can be used for generating a Voronoi diagram

### James Gosling

{He is known as the father of Java programming language and is also the creator of Gosmacs, which was the first Emacs to run under UNIX}

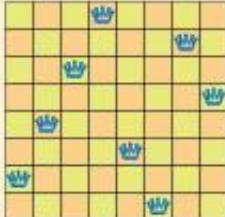




Guido van Rossum

*{He is well known as the author  
of Python programming  
language and is currently  
employed by Google}*

// Infinite Fibonacci sequence in Brainfuck language



## // A solution to the Eight Queen Problem

## Timeline

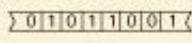


## The Beginning



A circular orange timeline marker with a black arrow pointing downwards, positioned above the text about Ada Lovelace.

Fundamentals



1936  
SARWAN@NIELIT

## The first languages



A horizontal timeline with three colored circles and arrows pointing downwards to labels. The first circle is pink and points to 'FORTRAN'. The second is yellow and points to 'ALGOL 58'. The third is brown and points to 'COBOL'. Above each label is its corresponding year: '1955' above FORTRAN, '1958' above ALGOL 58, and '1959' above COBOL.

Power at the highest level



A horizontal timeline with three vertical arrows pointing downwards from the year labels to the language names. The timeline starts at 1980 and ends at 1991.

- 1980**: Below the arrow is the text *C++*.
- 1987**: Below the arrow is the text *Perl*.
- 1991**: Below the arrow is the text *Python and Java*.

## Age of the Internet

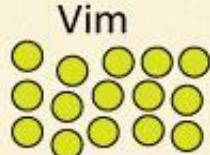


1995  
RHP and IomegaScsi

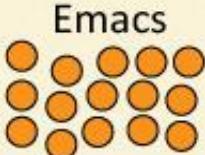


Richard Stallman

*{He is the creator of Emacs editor and the lead architect and organizer of the GNU project. He has been actively involved in the free software movement}*



*// Vim and Emacs are the two most popular editors amongst “Real Programmers”*



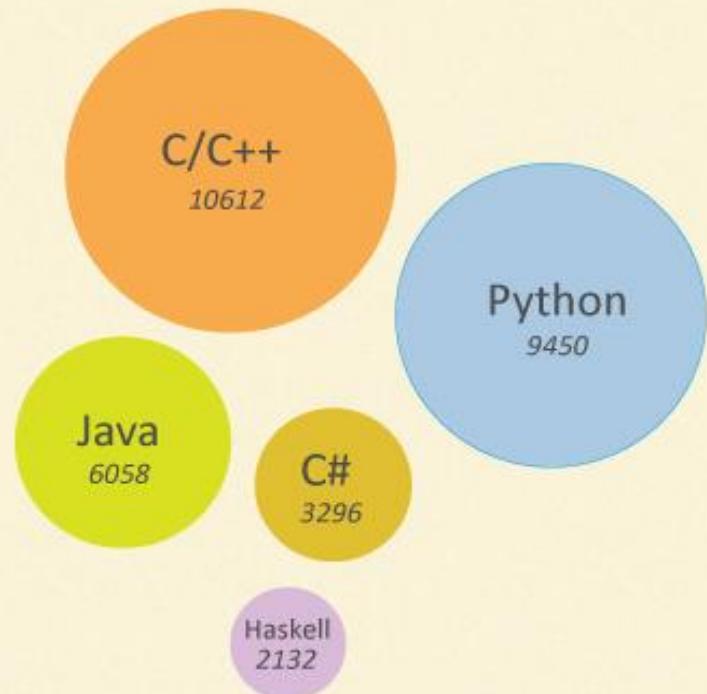
Bill Joy

*{He is the co-founder of Sun Microsystems and is also the creator of vi editor, csh and NFS. He was also a primary figure in the development of Java programming language}*



# Project Euler

Project Euler is a series of challenging mathematical/computer programming problems. Following stat depicts the top five programming languages by most number of users.



# StackOverflow

Stack Overflow is a website featuring questions and answers on a wide range of topics in computer programming. Following stat depicts the top three programming languages by most numbers of questions asked.

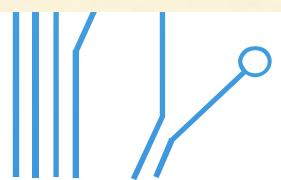


# Tiobe Index

The TIOBE Programming Community index gives an indication of the popularity of programming languages. The index is updated once a month. Following stat depicts the top three programming languages for April 2010.



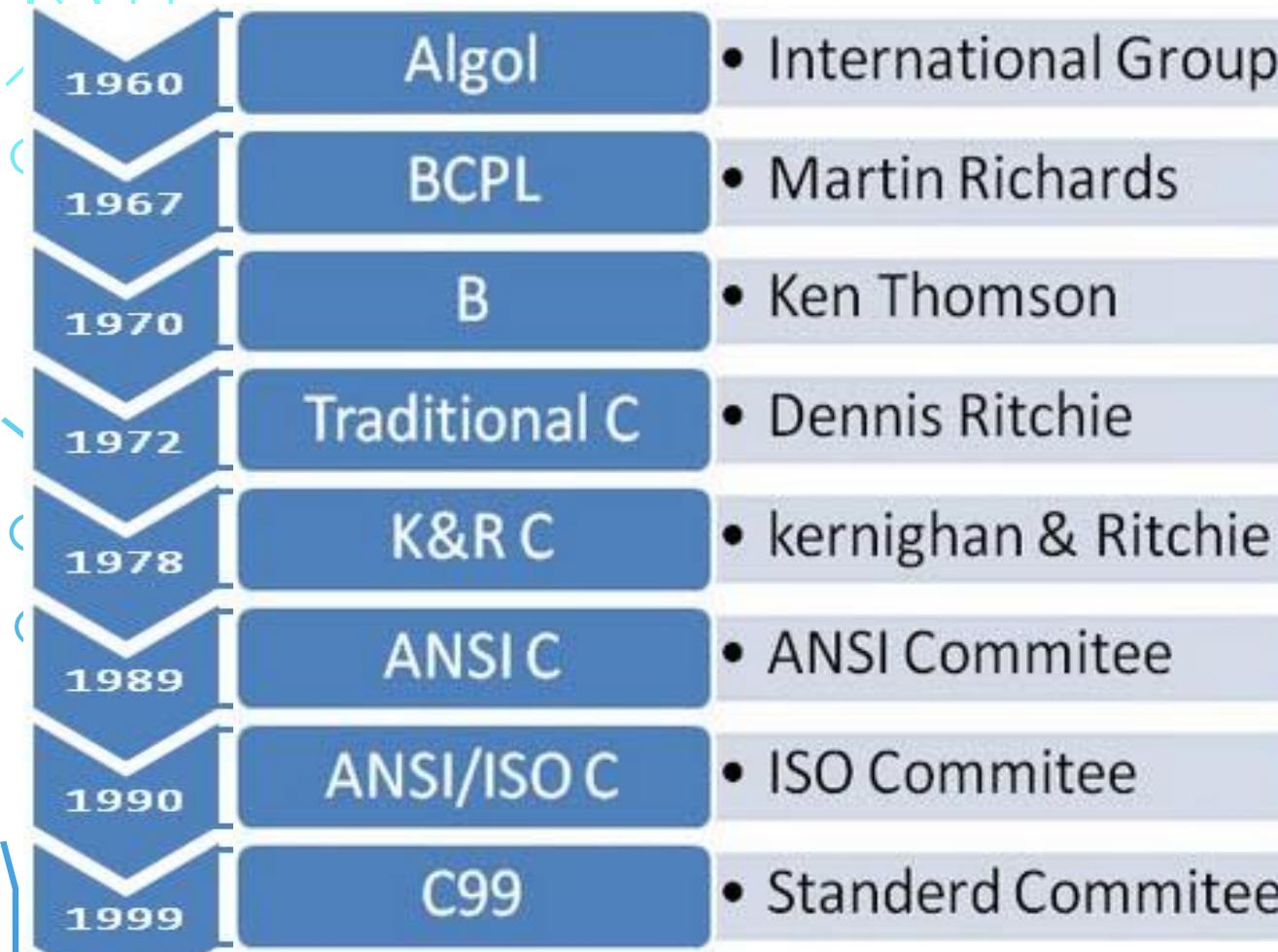
Source: smashingmagazine.com



SARWAN@NIELIT



# C LANGUAGE



SARWAN@NIELIT



- Creator of C Programming lang. and Creator of UNIX operating System
- 1941-2011
- Harvard Graduate



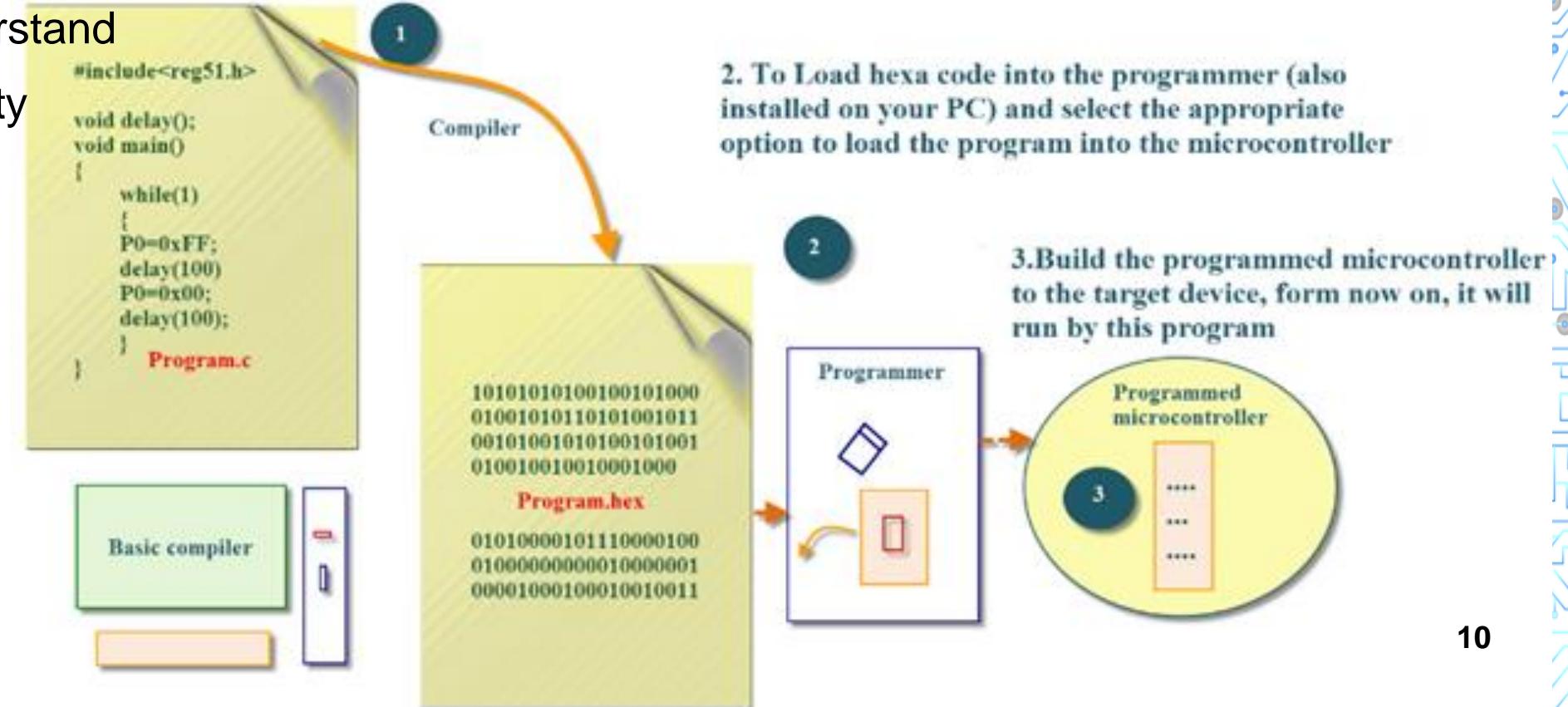
Dennis Ritchie : With Ken Thompson (Creator of BCPL)

# EMBEDDED SYSTEM PROGRAMMING

- Most embedded devices used by us, uses microcontroller that are programmed by embedded C

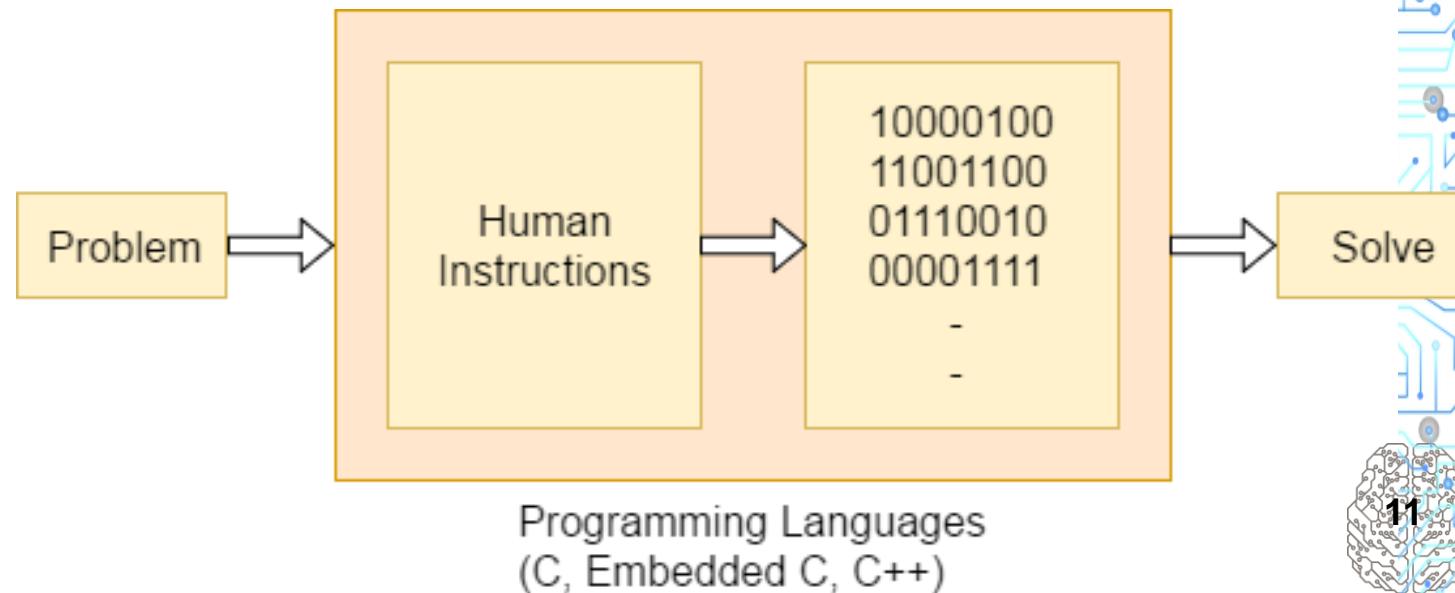
## Advantages

- Easy to understand
- High Reliability
- Portability
- Scalability



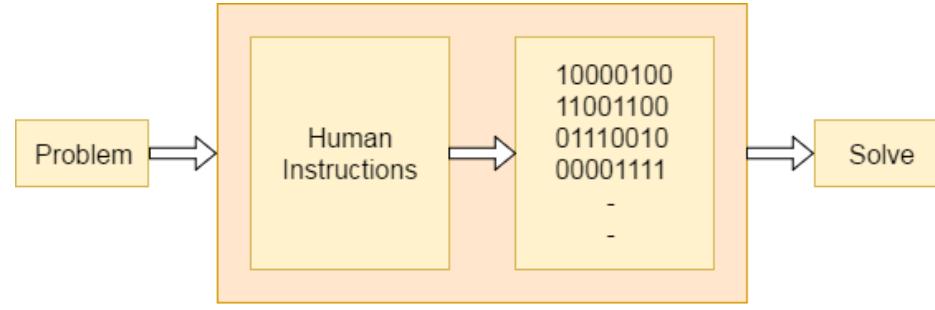
# EMBEDDED SYSTEM PROGRAMMING

- block diagram of Embedded C Programming development
- The C language programming is designed for function with variables, character set, data types, keywords, expression and so on

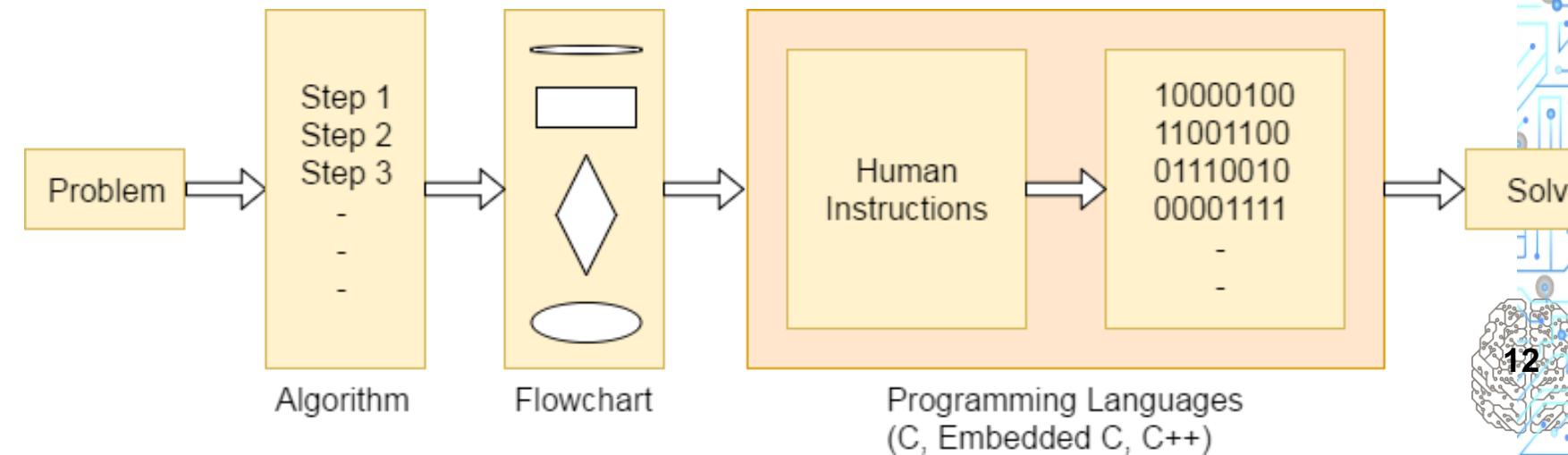


# EMBEDDED SYSTEM PROGRAMMING

- The microcontroller programming is different for each type of operating system. Even though there are many operating system are exist such as Windows, Linux, RTOS, etc



Programming Languages  
(C, Embedded C, C++)



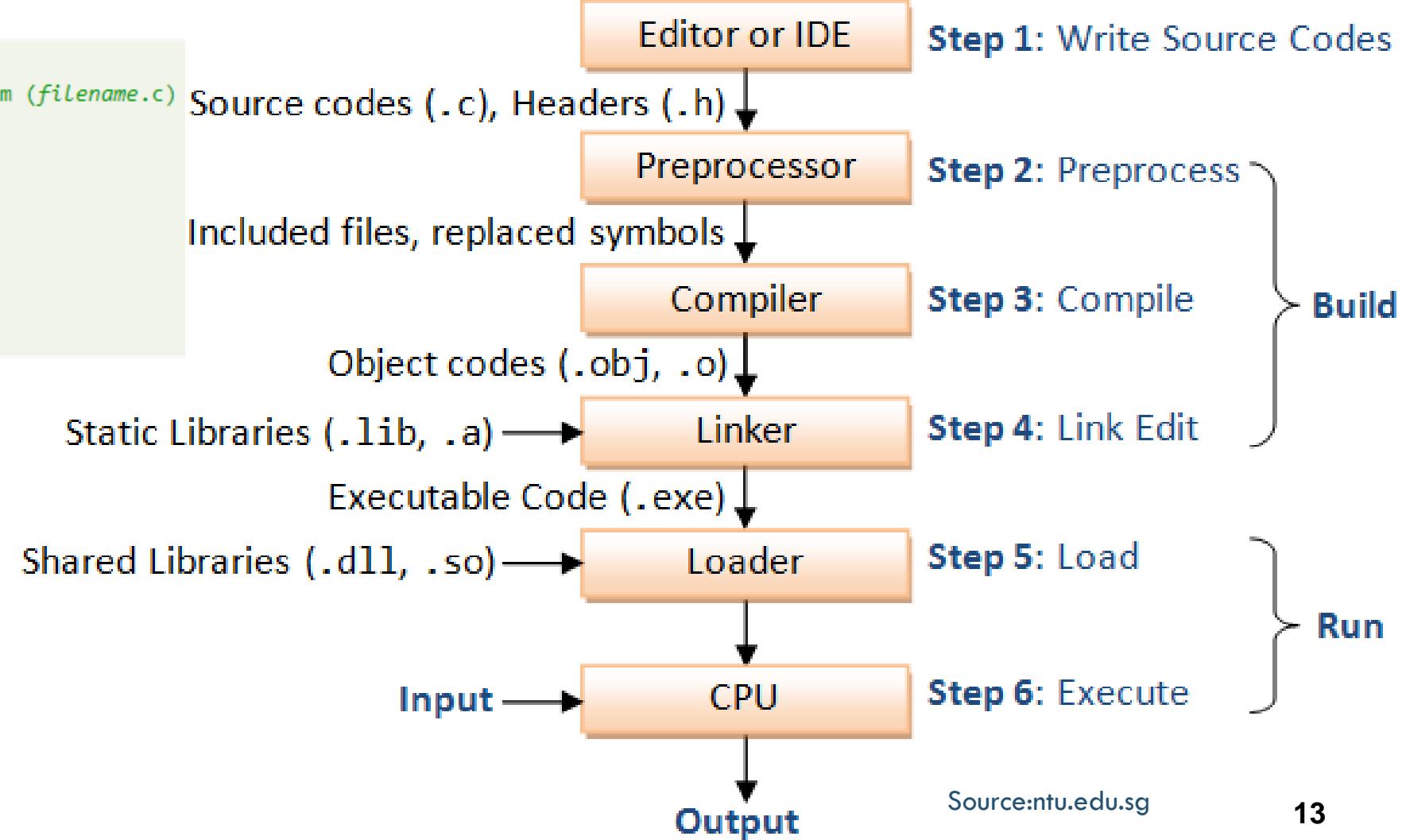
Programming Languages  
(C, Embedded C, C++)



# THE PROCESS OF WRITING A C PROGRAM

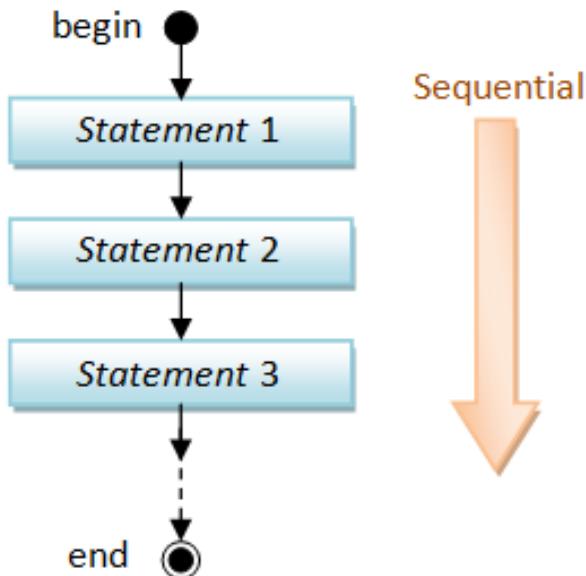
- Program template

```
1  /*
2   * Comment to state the purpose of this program (filename.c)
3   */
4 #include <stdio.h>
5
6 int main() {
7     // Your Programming statements HERE!
8
9     return 0;
0 }
```



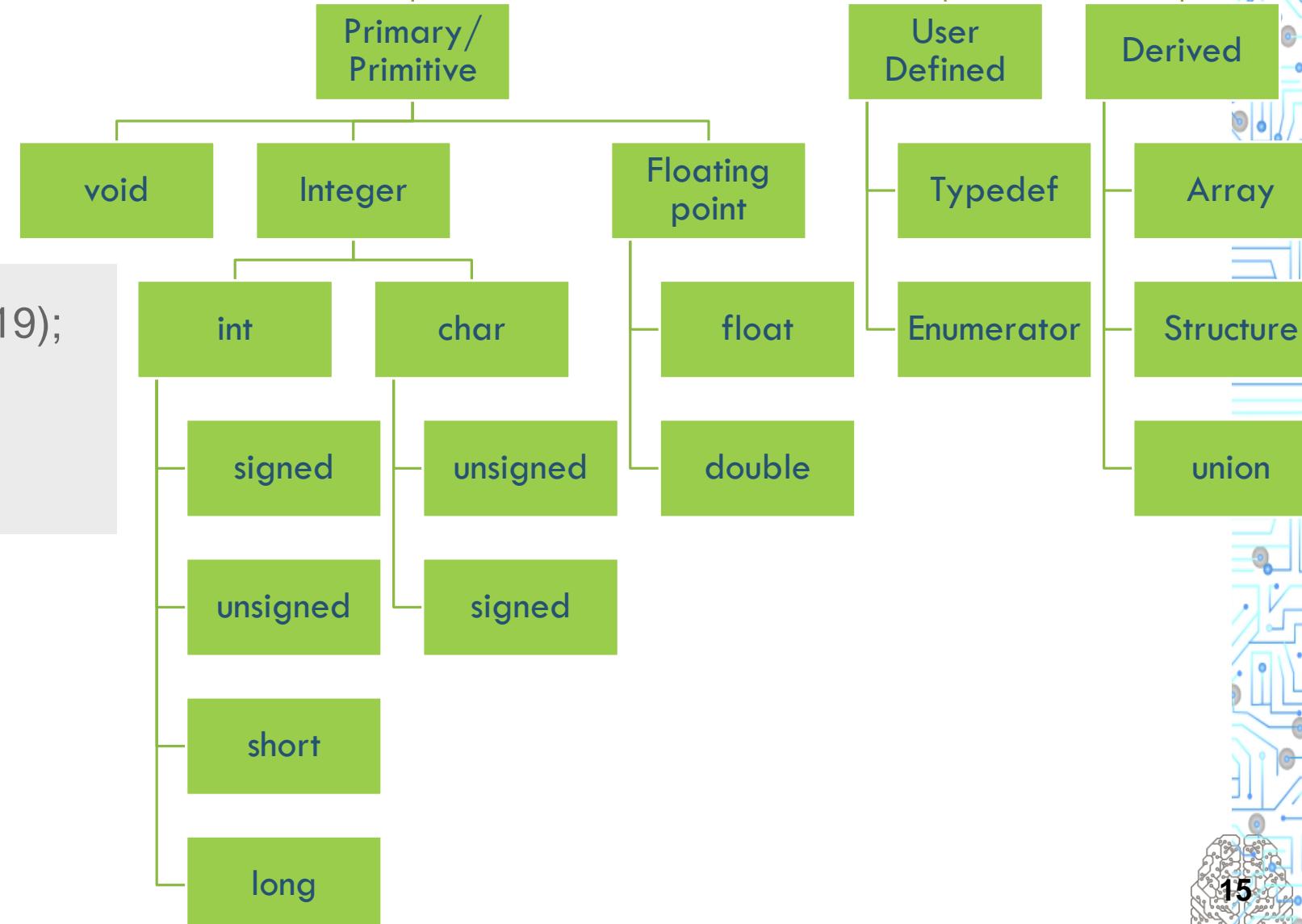
# WHAT IS A PROGRAM

- A *program* is a sequence of *instructions* (called *programming statements*), executing one after another - usually in a *sequential* manner, as illustrated in the previous example and the following flow chart.
- C is case sensitive language
- Each declaration statement is terminated with a semi-colon (;).
- In multiple-variable declaration, the names are separated by commas (,)



# DATATYPE IN C LANGUAGE

Datatype in  
C Lang.



```
typedef int numbers;  
numbers a=1;
```

```
enum prime (2,3,5,7,11,13,17,19);  
enum prime a,b;  
a=5;  
b=19;
```



# DATATYPE IN C LANGUAGE

Data type	Size	Range	Description
<b>char</b>	1 byte	-128 to 127	A character
<b>signed char</b>			
<b>unsigned char</b>	1 byte	0 to 255	A character
<b>short</b>			
<b>signed short</b>	2 bytes	-32,767 to 32,767	Short signed integer of minimum 2 bytes
<b>signed short int</b>			
<b>unsigned short</b>			
<b>unsigned short int</b>	2 bytes	0 to 65,535	Short unsigned integer of minimum 2 bytes
<b>int</b>			
<b>signed int</b>	2 or 4 bytes	-32,768 to 32,767 or -2,147,483,648 to 2,147,483,647	An integer (Both positive as well as negative)
<b>unsigned int</b>	2 or 4 bytes	0 to 65,535 or 0 to 4,294,967,295	An unsigned integer <sub>16</sub> (Positive integer)

# DATATYPE IN C LANGUAGE

Data type	Size	Range	Description
<b>long</b>	4 bytes	-2,147,483,648 to 2,147,483,647	Long signed integer of minimum 4 bytes
<b>signed long</b>			
<b>signed long int</b>			
<b>unsigned long</b>	4 bytes	0 to 4,294,967,295	Long unsigned integer of minimum 4 bytes
<b>unsigned long int</b>			
<b>float</b>	4 bytes	1.2E-38 to 3.4E+38	Single precision floating point number
<b>double</b>	8 bytes	2.3E-308 to 1.7E+308	Double precision floating point number
<b>long double</b>	12 bytes	3.4E-4932 to 1.1E+4932	Double precision floating point number



# QUALIFIER-MODIFIER

- **register**- Local variable are stored in register instead of RAM
- **static** defined local variables do not lose their value between function calls.
- **typedef** used to create new type
- **extern** used to declare global variable
- **volatile** variable values might keep on changing without any explicit assignment by the program

Group	Qualifiers (Modifier)	Default Qualifiers (Modifier)
1	auto, register, static, extern, typedef	auto
2	signed, unsigned	signed
3	Short, long	Not Short, not long
4	Const	Not Const
5	Volatile	Not Volatile



# EMBEDDED C DATATYPES

- **sbit:** This data type is used in case of accessing a single bit of SFR register.
  - sbit a=P2^1;
- **Bit:** This data type is used for accessing the bit addressable memory of RAM (20h-2fh).
  - bit c;
- **SFR:** This data type is used for accessing a SFR register by another name. All the SFR registers must be declared with capital letters.
  - SFR port0=0x80;

SARWAN@NIELIT

Source:elprocus.com

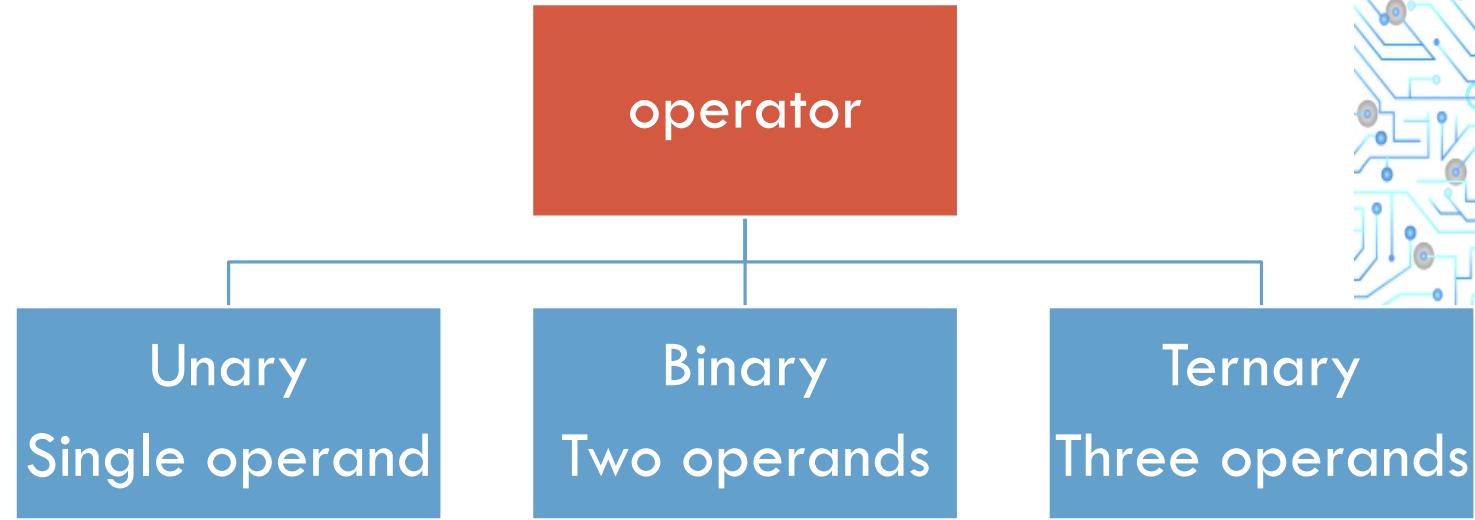
Name	Function
sbit	Accessing of single bit
bit	Accessing of bit addressable memory of RAM
sfr	Accessing of sfr register by another name

**SFR Register:** The SFR stands for ‘Special Function Register’. Microcontroller 8051 has 256 bytes of RAM memory.

- This RAM is divided into two parts:
- the first part of 128 bytes is used for data storage, and
  - the other of 128 bytes is used for SFR registers.

All peripheral devices like I/O ports, timers and counters are stored in the SFR register, and each element has a unique address.

# OPERATOR



**unary operator** → `++`, `--`

**Unary operator**

`+`, `-`, `*`, `/`, `%`

**Arithmetic operator**

`<`, `<=`, `>`, `>=`, `==`, `!=`

**Relational operator**

`&&`, `||`, `!`

**Logical operator**

`&`, `|`, `<<`, `>>`, `~`, `^`

**Bitwise operator**

`=`, `+=`, `-=`, `*=`, `/=`, `%=`

**Assignment operator**

**Ternary operator** → `?:`

**Ternary or conditional operator**

# PRECEDENCE & ASSOCIATIVITY

SARWAN@NIELIT

Operator	Description	Associativity
( ) [] . -> ++ --	Parentheses or function call Brackets or array subscript Dot or Member selection operator Arrow operator Postfix increment/decrement	left to right
++ -- + - ! ~ (type) * & sizeof	Prefix increment/decrement Unary plus and minus not operator and bitwise complement type cast Indirection or dereference operator Address of operator Determine size in bytes	right to left
* / %	Multiplication, division and modulus	left to right
+ -	Addition and subtraction	left to right
<< >>	Bitwise left shift and right shift	left to right
< <=	relational less than/less than equal to	left to right
> >=	relational greater than/greater than or equal to	left to right
== !=	Relational equal to and not equal to	left to right
&	Bitwise AND	left to right
^	Bitwise exclusive OR	left to right
	Bitwise inclusive OR	left to right
&&	Logical AND	left to right
	Logical OR	left to right
? :	Ternary operator	right to left
= += -= *= /= %=&= ^=  = <=>=	Assignment operator Addition/subtraction assignment Multiplication/division assignment Modulus and bitwise assignment Bitwise exclusive/inclusive OR assignment	right to left
,	Comma operator	left to right



रा.इ.सू.प्रौ.सं

NIELIT

$$= 12 + 3 - 4 / 2 < 3 + 1$$

$$= 34 + 12/4 - 45$$

$$12 + 3 - 4 / 2 < 3 + 1$$

1

4

2

4

$$12 + 3 - 2$$

2

15

$$15 + 2$$

3

$$17 < 4$$

5

SARWAN@NIELIT

Ans: 0

$$34 + 12/4 - 45$$

1

2

$$34 + 3$$

$$37$$

3

$$37 - 45$$

Ans : -8

# BITWISE OPERATOR

- Bitwise AND operator &
- Bitwise OR operator |
- Bitwise XOR operator ^
- Bitwise complement operator ~
- Bitwise left shift operator <<
- Bitwise right shift operator >>

# BITWISE OPERATOR

Write a C program using bitwise operator

- to check Least Significant Bit (LSB) / Most Significant Bit (MSB) of a number is set or not.
- to get / set nth bit of a number.
- to clear nth bit of a number.
- to toggle nth bit of a number.
- to get highest / lowest set bit of a number.
- to count trailing / leading zeros in a binary number.
- to flip bits of a binary number using bitwise operator.
- to count total zeros and ones in a binary number.
- to convert decimal to binary number system using bitwise operator.
- to swap two numbers using bitwise operator.
- to check whether a number is even or odd using bitwise operator.

# LOOP – to perform repetitive task

initialization-statement;

while (test) {

    loop-body;

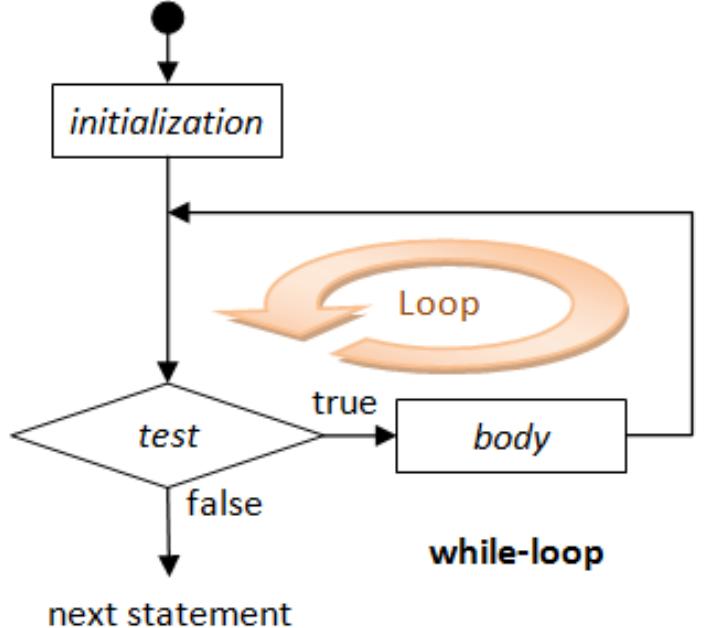
}

next-statement;

Other variants :

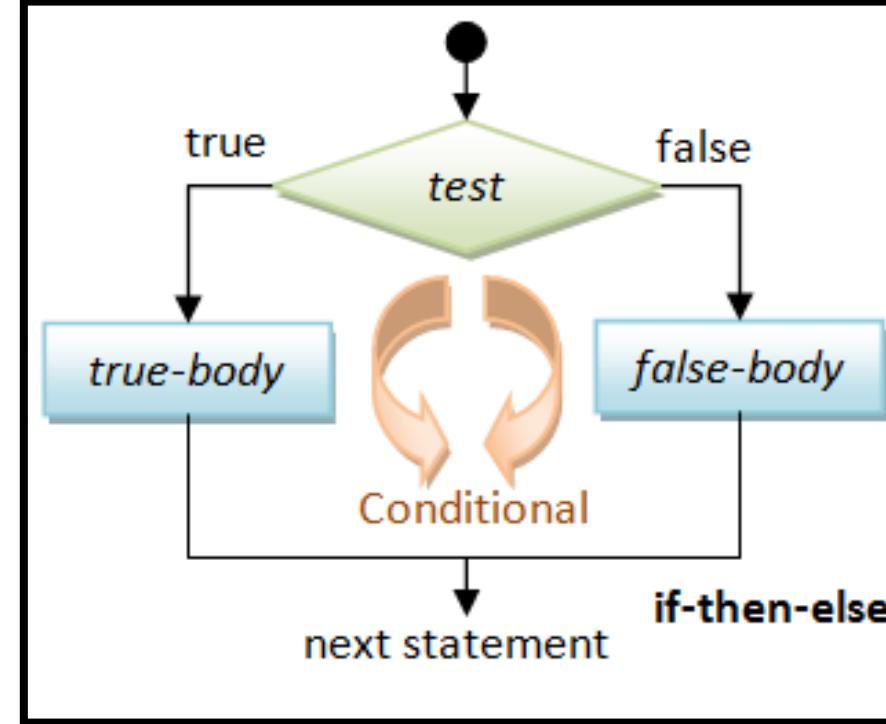
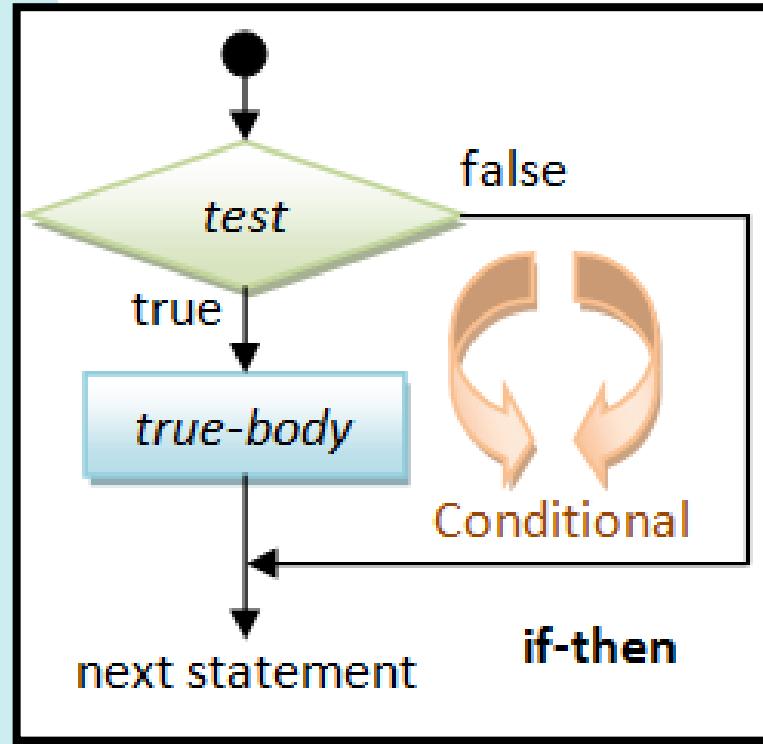
do { .... } while (condition )

for( initialization ; condition ; inc/dec) { .... }



# CONTROL STATEMENT - *conditional (or decision)*

```
// if-then  
if ( test )  
{ true-body; }  
// if-then-else  
if ( test )  
{ true-body; }  
else  
{ false-body; }
```



Other variant

- switch case

# FUNCTION / METHOD

- Reduces code duplication
- Repetitive task can be represented in form of method
- Make code modular
- provide abstraction

```
// A function that takes two integers as  
// parameters and returns an integer  
int max(int, int);
```

```
// A function that takes a int pointer and an  
// int variable as parameters  
// and returns an integer of type int  
int *swap(int*,int);
```

```
// A function that takes a char and an int as  
// parameters and returns an integer  
int fun(char, int);
```

# QUESTION

- Write a one line C function to round floating point numbers

```
int roundNo(float num)
{
    return num < 0 ? num - 0.5 : num + 0.5;
}
```

# HEADER FILES

- Header file is a file that contains function declaration and macro definition for C in-built library functions.
- All C standard library functions are declared in many header files which are saved as `file_name.h`.