Complete C Programming

Introduction & Evolution of C

- It is programming language .
- Developed at AT & T's Bell Laboratories of USA in 1972.
- It is design and written by Dennis Ritchie & Ken Thomson.
- C language is the one who replaced previously used Language like ALGOL, BASIC etc .

Features of C language

- What is C?
- How it come in existence ?
- How does it compare with other language ?
- FOUR IMPORTANT ASPECT OF ANY PROGRAMMING LANGUAGE
- The way it store the data .
- The way it operates upon this data.
- How it accomplishes input and output .
- How it lets you control the sequence of execution of instruction in program

Why we use c?

- C language is reliable , simple and easy to use .
- Language that has survived for more than 5 decades is C, so we still use C as programming language.
- It is root of all the programming language .
- Major part of popular operating system like WINDOWS, UNIX , LINUX are still written in C.
- Programs written in C are not only have to run fast but also have to work in limited amount of memory .
- C language provide several language element to interact with hardware without compromising the performance

Keywords and Identifiers

Keywords

Keywords : are the **reserved** words whose meaning has already been defined in the c compiler. it can not be used as variable name.

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

Rules

- Each instruction in a c program is written as a separate statement
- Blank spaces may be inserted between two words to improve readability
- All statement must be in small case.
- C statement must be end with a “;”

Syntax of c program

```
#include<stdio.h>
        Void main()
            {
                statement 1 ;
                statement 2;
                statement 3;
             }
```

Syntax of c program

```
#include<stdio.h>
#include<conio.h>
void main()
{
   clrscr();
    printf("Viva technologies!");
    getch();
}
```

C Constants & Variables

Constant & Variables

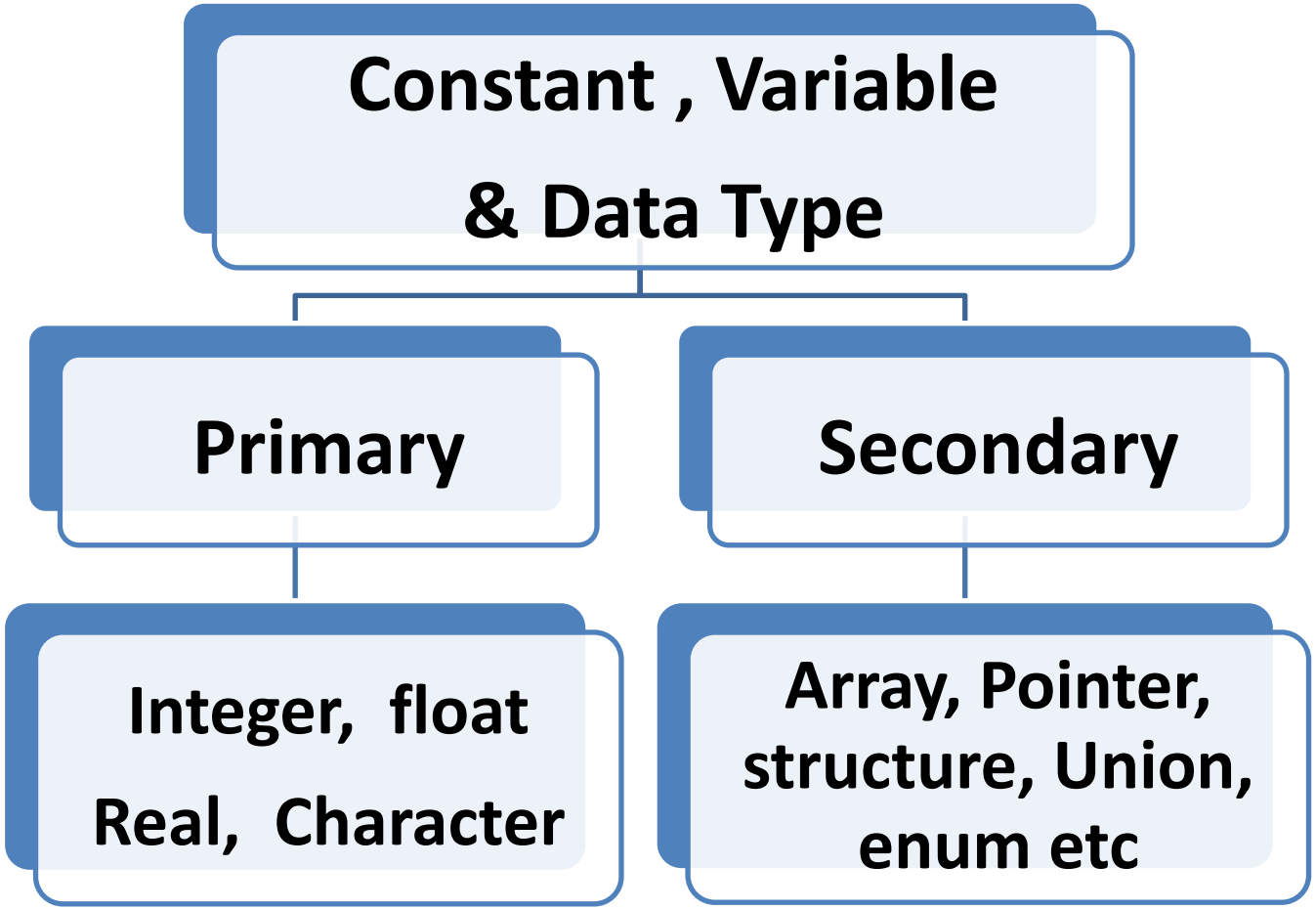
- **CONSTANT:** is an entity that does not change its value.

  Example: a = 5;

- **VARIABLE:** is a entity that may vary during program execution.

  Example : a = b or a ;

Constant & Variables are divided in two categories:



C PROGRAM - Constant

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a=5;   int b=5;  int c;
clrscr();
c = a+b ;
printf ("result:%d",c);
getch();
}
```

C PROGRAM- Addition

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a,b,c;
clrscr();
printf(“Enter First No. for Addition:  \n");
scanf("%d",&a);
printf(“Enter Second No. for Addition:  \n");
scanf("%d",&b);
c=a+b;
printf(“Answer = %d",c);
getch();
}
```

C PROGRAM- Area & Perimeter Calculator

```
#include<stdio.h>
#include<conio.h>
void main()
{
int length, breadth, area, peri;
clrscr();
printf("Enter the length of rectangle :\n");
scanf("%d",&length);
printf("Enter the breadth of rectangle :\n");
scanf("%d",&breadth);
area=length*breadth;
peri=(2*(length+breadth)) ;
printf("\nArea = %d",area);
printf("\nPerimeter = %d", peri);
getch();
}
```

C PROGRAM- Addition

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a;
clrscr();
printf(“Enter a Number: \n");
scanf("%d",&a);
printf(“Square = %d",a*a);
getch();
}
```

C PROGRAM- Addition

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a;
clrscr();
printf(“Enter a Number: \n");
scanf("%d",&a);
printf(“Cube = %d",a*a*a);
getch();
}
```

C Data Types

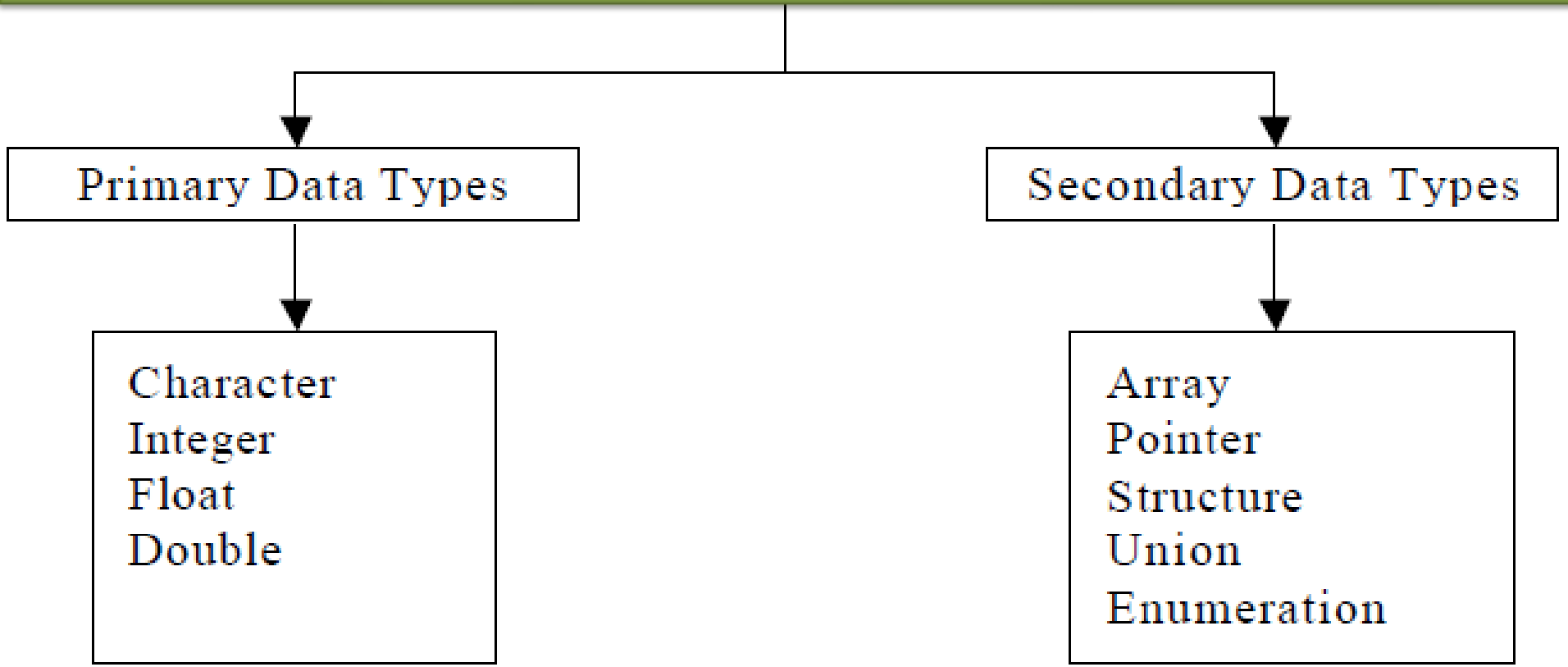
DATA TYPES

Data Type: is used to define a variable before it's use (or limits the contents).

Ex:

char	Char - 1 byte	(A,B,C,D . . . .)
int	Int - 2 bytes	(-2,-1, 0, 1,2,……)
float	Float- 4 bytes	(1.2 , 1.3 …..)
double	Double- 8 bytes	(1.2 , 1.3,1.4 …..)
short		
long		

DATA TYPES


Primary Data Types
Secondary Data Types
Character
Integer
Float
Double
Array
Pointer
Structure
Union
Enumeration

Data Types and sizes

Types	Bits	Byte	Possible values
Char	8	1 Byte	-128 to 127
Unsigned char	8	1 Byte	0 to 255
Int	16	2 Byte	-32768 to 32767
Unsigned int	32	4 Byte	0 to 4,294,967,295
Short	16	2 Byte	-32768 to 32767
Unsigned short	16	2 Byte	0 to 65535
Long	32	4 Byte	- 2147,483648 to 147,483648
Unsigned long	32	4 Byte	0 to 4,294,967,295
Float	32	4 Byte	3.402823466e+38F
Double	64	8 Byte	1.7976931348623158e+308

Escape Characters

Character	Description
\b	Backspace
\n	New line
\a	Beep
\t	Tab
\”	”
\\	\
\’	’
\r	Carriage return

Format Control Strings

Data Type	Conversion Specifier
signed char	%c
unsigned char	%c
short signed int	%d
short unsigned int	%u
long signed int	%ld
long unsigned int	%lu
float	%f
double	%lf
long double	%Lf


thank you