

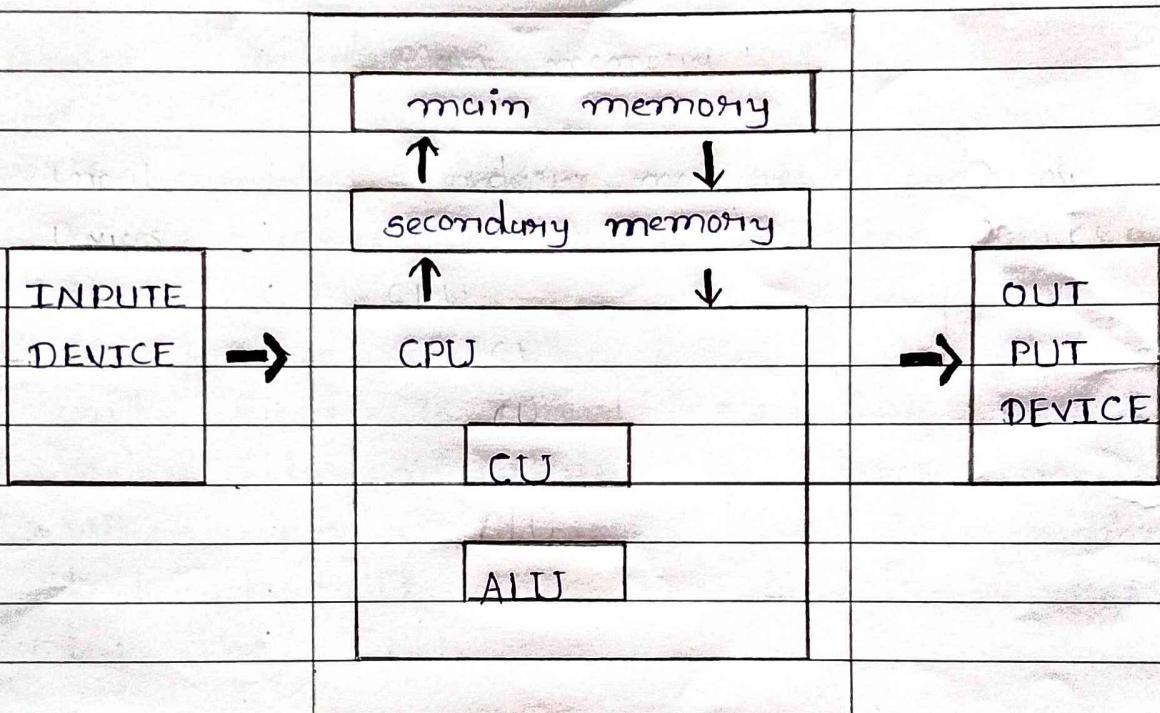
Assignment

Computer Fundamentals

30/09/2023

Q-1 Explain Simple model of Computer in detail.

=> Simple Model of Computer :-



i) Input device :-

- Input device is the device by which we can send something (data) to computer from outside of the computer.
- Ex: Mouse, Keyboard, Scanner, light pen, joystick etc.

ii] Output device :-

- The device send the result done by the computer to the user (outside the computer).
- Ex: Monitor, Printer, Speaker etc.

iii] Main Memory :-

- The main memory of the computer is also known as RAM, standing for Random Access Memory.
- It is constructed from integrated circuits and need to have electrical power in order to maintain its information.
- When power is lost, the information is lost too! It can be directly accessed by the CPU.

iv] Secondary Memory :-

- Secondary Memory is where programs and data are kept on a longterm basis.
- Common secondary storage device are the hard disk and optical disks.
- The hard disk is usually contained inside the case of a computer.

V] CPU:-

The part of a computer in which operations are controlled and executed.

(1) Control Unit:-

- The control unit (CU) is a component of a computer's central processing unit (CPU) that directs operation of the processor.
- It tells the computer's memory, logic/arithmetic unit and input and output devices how to respond to a program's instructions.

(2) Arithmetic Logic Unit:-

- An arithmetic logic unit (A.L.U.) is a digital electronic circuit that performs arithmetic and bitwise logical operations on integer binary numbers.
- It is a fundamental building block of the central processing unit (CPU) found in many computers.

Q-2 Explain RAM and ROM in detail.

=> * Random Access Memory (RAM):-

It is stands

for Random Access Memory.

- It is a volatile memory as the data is lost when the power is turned off.
- RAM is also known as Direct Access Memory.
- The task currently perform by the CPU are stored in RAM.
- RAM is a part of CPU.
- RAM data directly access by CPU.
- It is very fast memory.

The types of RAM:-

(1) SRAM:-

It is stands for Static Random Access Memory.

- Data remains in SRAM as long as there is power supply.
- It does not need to be refresh again and again.
- It is also used as cache memory.
- It is expensive.

(2) DRAM:-

It is stands for Dynamic Random Access Memory.

- Data can be stored in the DRAM only when it is refresh frequently.
- It is inexpensive.
- It is also used as main memory.
- * → It is slower than SRAM.

* [ROM] Read Only Memory :-

ROM is stands

For Read only Memory.

- In ROM we can only read but can't write on it.
- Rom data is permanently during creation of information that's why it is called non-volatile memory.
- It always retains its data.
- It is used in calculators and peripheral devices.
- Used in embedded systems on where the programming needs no change.

Assignment

C - Language

01/10/2023

Q-1

Explain C-Tokens in detail.

=>

In C program smallest individual units are known as C-Tokens. C has 6 types of token. They are mention below.

1. Variable / Identifier / Data Names:
2. Keywords
3. Constants
4. Operators

① Variables / Identifier / Data Names:-

Variable can be considered as a name given to the location in memory, where the constant is stored naturally. The constant of variable can change.

A variable is a Data-Name. That may be used to store a data-value. A variable may take different value at different time during execution. Variable also known as Identifier. It also known as a user define word.

* Rules :-

- The maximum length of a variable name is 32 characters.
- The variable can contain alphabets (a to z) or digits (0 to 9).
- No special symbol other than an underscore _ can be used in a variable name.

- The first character in the variable name is must be an alphabets.
- Keywords can not be used as variable name.
- Comma or blank-space are not allowed within a variable name.
- Uppercase, Lowercase are significant.
ex: The variable - TOTAL is not this sent as variable-total.
- The example of valid name or invalid name are following...

Valid name	Invalid Name
Student	\$name
Std-name	Std name
c1	123
null-no	
Sum	

(2) Keywords:-

The keywords are also called as reserve words. Keywords are the words whose meaning has already been explained to C compiler. All keyword must be written in lowercase.

The keyword can not be used as variable because if, be, do, so we are

8

trying to assign a new meaning to the keyword, which is not allowed why computer.

→ There are 32 keywords in C. The list of 32 keywords are following...

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

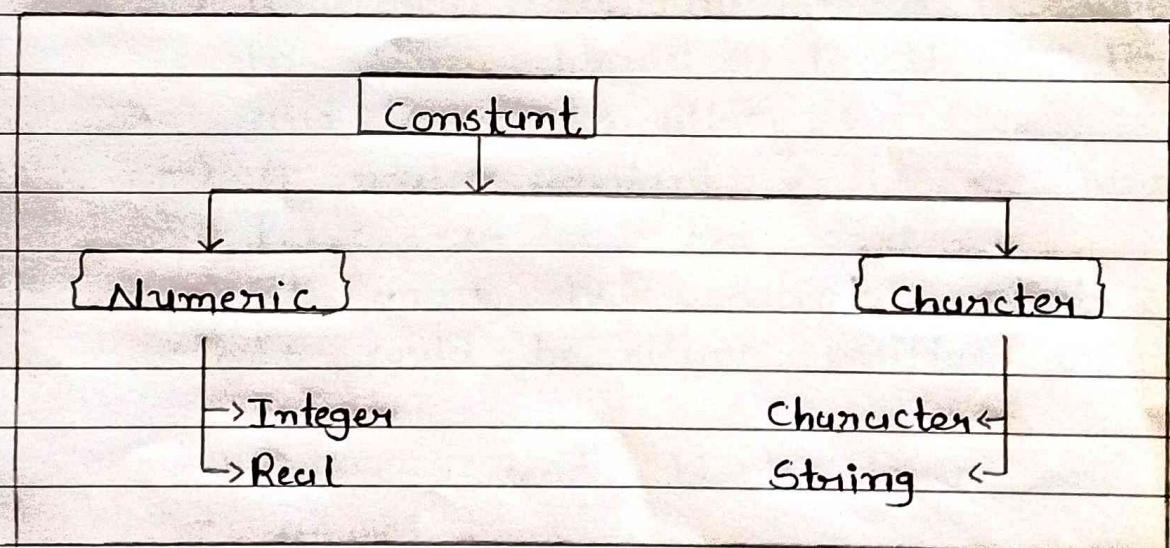
- All keyword have fixed meaning and this meaning can not be change.
- All keywords must be written in lowercase.
- There are 32 keywords in C.
- Some computer may be use additional

keywords that must be identify from the C manual.

(3) Constants:-

A constant is quantity that does not change. This quantity can be stored at a location in the memory to the computer.

→ In a C constant refers to fixed value that do not change during the execution of program.



I) Numeric Constant :-

(a) Integer:-

In a integer constants refers to sequence of digits.

- Does not include decimal value.
- Default sign is positive.

- It could be either positive or negative.
- No comma and blank space are allowed.
- The range for integer constant is:-
ex: -32768 to 32767
5000, -200, +570, etc...

(b) Real :-

Real constant are often called floating constant.

- The real constant could be written in two forms: (1) Fractional Form and (2) Exponential Form.
- A real constant could be must have atleast one digit.
- It must have decimal point.
- It could be either positive or negative.
- Comma and blank space are not allowed.
- No symbols are allowed.
ex: 328.50, -48.84, +101.32 etc.

II > character constant :-

(c) Character :-

A single character constant consist to a single alphabet, single

- digit or single special symbol.
- The maximum length of character constant can be one character.
 - It is enclosed within a pair of single quote marks.
 - ex: 'a', '5', '\$', '@' ...

(b) String:-

- A string constant is a sequence of characters.
- It is enclosed within a pair of double quote marks.
 - The character may be letters (a to z), numbers (0 to 9), special characters and blank space.
 - ex: "We come", "Verical", "Hello...123", etc

④ Operators:-

Operators is a symbol which represent a particular operation. that can be performed on data.

- C supports a rich set of operators.
- An operators is a symbol tells the computer to perform certain mathematical or logical manipulation.

I) Arithmetic operators:-

- It used for performing arithmetic calculation
- There are 5 arithmetic operators in C language

Operators	Meaning / Purpose	Use
+	addition or unary plus	(a+b)
-	Subtraction or unary minus	(a-b)
*	Multiplication	(a*b)
/	Division	(a/b)
%	Module devision	(a%b)

→ The module division produce remainder of an integer division.

II) Relational Operators:-

→ Relational operators are used for compare value of two operators.

→ An expression containing a relational operator is called relational expression.

→ Relational expression are used in decision statements such as if, and, while.

→ There are 6 Relational operators in C language.

Operators	Meaning / purpose	Use
<	Is less than	$a < b$
>	Is greater than	$a > b$
\leq	Is less than or equal to	$a \leq b$
\geq	Is greater than or equal to	$a \geq b$
$=$	Is equal to	$a = b$
\neq	Is not equal to	$a \neq b$

III > Logical Operators:-

- Combined two or more relational expressions are turned as a logical expression or a compound relational expression.
- When decision is to be taken on the basis of evaluation two or more different condition in same expression logical operators are used.
- Logical operators help to combine more than one condition and take a decision.
- C provides 3 logical operators. They are...

Operators	Meaning / Purpose	Use
&&	Logical AND	$(a>b) \&\& (a>c)$
	Logical OR	$(a>b) (a>c)$
!	Not	$!(a!=b)$ $!(a>b)$

Truth - Table

Condition - 1	Condition - 2	Result - &&	Result -
T	T	T	T
T	F	F	T
F	T	F	T
F	F	F	F

The result of the AND operator will be true if both condition are true. The result of OR operator will be true if either one condition true. Inshort the result of OR operator will be false only if both condition are false.

! (Not) :-

The ! (Not) operator make a true expression false and a false expression true. It return true if the evaluation of the expression is false. It's return false if the evaluation of the expression is true. Inshort it returns the reverse value of the given operant.

IV > Assignment Operators:-

- Assignment operators are used to assign the values.
- It assigns the value of an expression to an identifier.
- C language support equal (=) as an assignment operators.
ex: i) $a = 5$
ii) $b = 15.2$
iii) $c = p * R * N / 100$
- C also supports short-hand notation.
The form of short-hand operator is...

Syntax:

Variable operation = Expression ;

- | | |
|-------------|--------------------------|
| Ex: $x + 3$ | $\rightarrow x = x + 3$ |
| $x - 3$ | $\rightarrow x = x - 3$ |
| $x \% 3$ | $\rightarrow x = x \% 3$ |
| $x / 3$ | $\rightarrow x = x / 3$ |

V\S Increment and Decrement Operators:-

$++$ Increment ($+I$)

$--$ Decrement ($-I$)

$\overset{\leftarrow}{++} a \rightarrow$ Pre

$a ++ \rightarrow$ Post

$++a \rightarrow$ Pre Increment

$--a \rightarrow$ Pre Decrement

$a ++ \rightarrow$ Post Increment

$a -- \rightarrow$ Post Decrement

e.g. $a = 3$

$$++a = I+3$$

$$--a = I-3$$

$$a ++ = 3+1$$

$$a -- = 3-1$$

\rightarrow C has two very useful operators not generally found in other languages.

\rightarrow $++$ and $--$ are called increment and decrement operators.

\rightarrow The operator $++$ add 1 to the operand while $--$ subtract 1 to the operand.

VI > Conditional Operators:-

- It is also called as Ternary operators.
- These operators take three operands.
- ? and : are used as conditional operators.
- It is used to check conditions as an alternative for if... else statement.

~~Syntax:~~ Expression-1 ? Expression-2 : Expression-3

- Here expression-1 is evaluated first.
- If expression-1 is evaluated true.. expression-2 is evaluated as if expression-1 is false... expression-3 is evaluated.