Qus 1

Problem Analysis

Problem Analysis in C Programming is the Process where we break down problems into its components, so that the problem can easily be understood.
The way to do this to write an Algo

→ The Problem analysis helps in designing and Coding for that particular problem.

1) Input Specifications:- The number of input and what forms are input are available.

2) Output Specifications:- The number of Output and what forms the output should be displayed

⇒ At this stage the problem is decomposed into sub-problems.

Qus: 2 What is the Role of Header file in C Language?

Ans: In C language, header files contain the set of predefined standard library functions
⇒ All the header file have a '.h' an extension.
⇒ We are including these header files in our C program using '#include <filename.h>'.
→ By Including a header file, we can use its contents in our program.

Qus What are the various Input / Output Statements in C?

Ans In 'C' Language, two types of Input / Output Statements are available. Input and Output Statements are used to read and write the data in C Programming There are embedded in stdio.h (standard Input / output header file).

1. Unformatted Input / Output Statements
2. formated Input / Output Statements

1 Unformatted Input / Output Statements :-

In this, Unformatted I/O transfers data in its raw form or binary representation without any Conversions

⇒ These statements are used to Input / Output a single group of characters from/to the Input / Output devices

⇒ It is portable only between machines that uses ASCII Character set

⇒ Example :- putchar(), getchar() etc.

Unformatted Input Statements :- getchar() and gets()

Unformatted Output Statements :- putchar() and puts()

② formatted I/O Statements :-

⇒ formatted I/O is always portable and human readable.

⇒ It enable the user to to specify the type of data and the way in which is should be read and written out.

⇒ for example

formatted Input Statement:- scanf()

formatted Output Statement:- printf()

Q: Difference between While Loop and do-while.

While	Do-while
① While loop is entry Control loop.	① Do-while loop is Exit Control loop.
② A while loop is a Pre-test loop	② Do-while loop is post test loop.
③ It tests the Condition before executing the loop body.	③ It tests the Condition at the end of the loop body.
④ Condition is at top.	④ Condition is at the bottom.
⑤ There is no semicolon at the end of while.	⑤ The semicolon is Compuls~~even if there is~~ a at the

⑥ while (condition)
{
statements
}
	⑥ do
	{
	statements
	} while (condition);

Q: Define Expressions

Ans An Expressions are formed by Combining Variables, Constants and function call using operators

Example
```
a*b-c;
x-y/z;
```

⇒ In this, Two expresion connected by an operator is also an expresion.

⇒ Two Opearctors should not occur in Continuation

⇒ Arithmetic Expressions: The expression which Contains arithmatic operators is called Arithmetic Expression.

example a*b-c

⇒ Evaluation of Expressions: Expresions are evaluated using an assignment statement of the form.

Variable = expression;

Example: x = a+b/c-d;

⇒ Precedence in Arithmetic Operators: An arithmetic

(18) Define Symbolic constants ?

Ans These constants may appear many times in program. for example: Mathematical constant "pi" whose value is 3.142.

Syntax:-

#define symbolic_name value of constant

For example

```
#define PI 3.14159
#define PASS MARK 50
#define A 5
```

Symbolic names are sometimes called constant identifiers.

Rules for defining symbolic constants:-

No blank space between the pound sign '#' and the word define.

'#' must be the first character in the line.

#define statements must not end with a semicolon.

A blank space is required between #define and symbolic name.

A blank space is required between symbolic name and the constant.

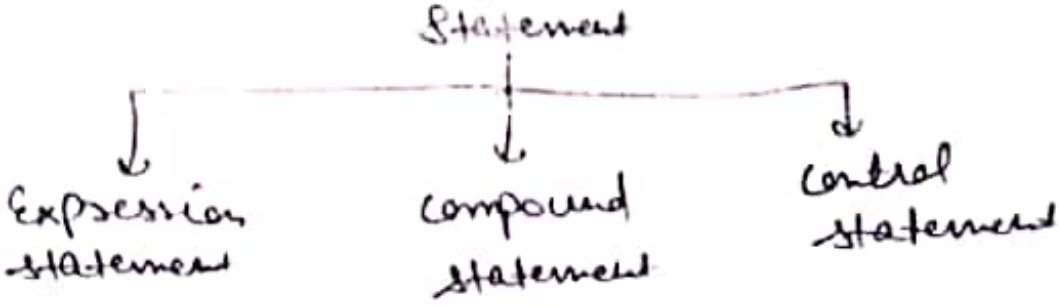
After definition, the symbolic name should not be assigned any other value within the program.

for example #define A 5

A = 5 // is Illegal.

Q- What is statement and type of statement use in C?

Ans The statement causes the computer to carry out some action. There are 3 types of statements and these are:-



① Expression statement:- It consist of an expression. Execution of it causes the expression to be evaluated.

eg:
```
sum = a+b;
a = 10;
printf("%d", a);
```

② Compound statement:- It consist of several individual statements enclosed in a pair of braces ({ and }). It does not ends with a semicolon

eg:-
```
{
  c = a+b;
  c = c+d;
}
```

Example of Control statement

③ Control statement:- They are used to create special program features such as loops, branches and logical tests. These statements are required to take an action according to the

Q.3 Difference between Constants and Variables

Ans

Constants	Variables
A Constant does not change its value.	Variables can change their value.
Constants are usually represented by numbers	Variables are usually represented by alphabets
The face value of Constants is known.	The value of variables is unknown.
For example, in the equation $3x+4=7$, here 4 and 7 are both Constants	For example :- $5x+3y=6$ here x and y are variables.

Q.4 Difference between Call by Value and Call by Reference

Ans

Call by Value	Call by Reference
① In this case passing value of Variable.	① In this case passing address of variables.
② No pointer are used.	② Pointer are used.
③ In this Can't change value of actual argument using formal argument	③ Can change value of actual argument using formal argument
④ It requires more memory	④ It requires less memory
⑤ It is less efficient.	⑤ It is more efficient
⑥ Example:- x(a);	⑥ Example:- x(&a);