Theory of C

Definition of variables and rules for constructing variable Names?

ANS: A **variable** is a data name that may be used to store a data value.

In other words a variable is an entity that may change.

Rules for variable Declaration :

- A variable name is a combination of 1 to 31 alphabets, digits or Underscore. Some compilers allow variable names whose length could be up to 247 characters.
- The first character in the variable name must be an alphabet or Underscore.

No commas or blanks are allowed within a variable name.

No special symbol other than underscore can be used in a variable name.

Definition of keyword?

ANS: A keyword is a word that carries special meaning and This meaning cannot be changed. We cannot use keyword as a variable because

Of Its special meaning that has already been explained to the c compiler.

There are 32 keywords in C language.(page 25 in ANSI C).

Properties of Array:

An Array is a number of same types variables.

1. **The first element in the array is numbered 0, so the last element is 1 less than the size of Array.**
2. **Before using an Array, It's type and dimension must be declared.**
3. **Array reduce the complexity of using a large number of variables.**
4. **In a big Array its element are always stored in contiguous memory location.**

Recursive function:

Ans: **Recursive function** is the function that **calls itself**.

What do You mean by Call by value and Call by reference?

a) Ans: Call by value means that In general no one can alter the actual arguments.

b) Ans: If a function is to be made to return more than one value at a time, then returns these values indirectly by using a call by reference.

Preprocessor Directives:

Ans: The preprocessor offers several features called preprocessor directives. Each of these preprocessor directives begins with a # symbol. The Directives can be placed anywhere in a program but are often placed at the beginning of the program.

➢ Number of Data types in C:

Ans: C supports 3 classes of data types:

I. Primary(or fundamentals) data type.
II. Derived data types.
III. User-defined data types.

All C compilers support five fundamentals data types

	NAME	SIZE (BYTES)
I.	Integer (int).	2
II.	Character(char).	1
III.	Floating point(float).	4
IV.	Long floating point (double).	8

➢ Definition of function:

Ans: A function is a self-contained block of statements that perform a coherent task of some kind. Every C program can be thought of as a collection of these functions. A function consists of

i. Function name;
ii. Function type;
iii. List of parameters;
iv. Local variable declaration;
v. Function statement; &
vi. A return value.

➢ Declaration of a function:

Ans: Like variables all function in c program must be declared before they are invoked. A function declaration consists of four parts:

I. Function type(return type).
II. Function name.
III. Parameter list.
IV. Terminating semicolon;

They are coded in the following format:

Function-type function-name (parameter list);

Function calls:

Ans: A function can be called by simply using the function name followed by a list of actual parameters, if any, enclosed in parameters.

Example:

```
Main
        {
                Int Y;
                Y=mul(10,5);  /* function call*/
                Printf("%d\n",Y);
        }
```

When a compiler encounter a function call, the control is transferred to the function mul(). This function is then executed line by line as described and a value is returned when a return statement is encountered.

The purpose of break statement:

Ans: We often across situations where we want to jump out of a loop instantly, without waiting to get back to the conditional test. The Keyword break allows us to do this. When break is encountered inside any loop. Control automatically passes to the first statement after loop. A break is usually associated with an if.

The purpose of continue statement:

Ans: In some programming situation, we want to take the control to the beginning of the loop, bypassing the statement inside the loop. Which have not been executed. The keyword continue allows us to do this. When continue is encountered inside any loop, Control automatically passes to the beginning of the loop. A continue is usually associated with an if.

Definition of C tokens and it's classification:

Ans: In a passage of text, individual words and punctuation makes are called tokens. Similarly in a C program the smallest individual units are known as C tokens.C has six types of tokens:

1. Keywords (float, while);
2. Identifiers (main, amount);
3. Constants (10,-10.7);

4. Strings (“abc”,”year”);
5. Special symbol ([],{});
6. Operators (+,-,*);

➢ Call by value & Call by reference:

Ans: We are familiar with how to call functions. But, if we observe carefully, whenever we call a function and passed something to it we have always passed the ‘values’ of variables to the called function. Such function calls are called ‘calls by value’. We have also learn that variables also stored somewhere in memory. So instead of passing the value of a variable, can we not pass the location number of the variable to a function? If are able to do so, it would become a ‘call by reference’.

➢ Function prototype:

Ans: Function prototype is the function declaration before main function ended with semicolon. Function prototype indicates that a function which after completing it’s execution either a value will return or not. If value returns , what types of value would return.

➢ The difference between do-while loop, while & for loop:

1. There is a minor difference between while and do- while loop. This difference is the place where the condition is tasted. The while taste the condition before executing any of the statements within the while loop. As against this, the do- while tests the condition after having executed the statements within the loop. This means do- while would execute its statement at least once, even if the *condition* fails for first time.bt while loop doesn’t execute any statement if the first condition fails.
2. Where as the for loop allows us to specify three things about a loop at a line.
    a) Setting a loop counter to an initial value.
    b) Testing the loop counter to determine weather its value is reached the number of representation desired.
    c) Increasing the value of loop counter each time the program segment within the loop has been executed.

➢ The difference between printf & scanf:

Ans:

Theory of C

1. Printf is used to display any statement ,values of variable & the result of an expression, Where as the scanf is used to receive them from the keyboard .
2. The structure of printf function is : printf (“ <format string> ”,<list of variables>); Where as the general structure of scan f is :

scanf(“%data type”, &variables); here & sign differentiate between the printf and scanf.