

1. There are two ways to write comments in C Programming one is Single Line and other one is to span comment on multiple lines.
 - Single line (//) – Single line comments start with two forward slashes (//). Any text between // and the end of the line is ignored by the compiler (will not be executed)
 - Comment that spans Multiple lines (/* - */) – The compiler will assume that everything after the /* symbol is comment until it reaches the */ symbol, even if it spans multiple lines within the c Program.
2. Main() - Without main function file will not be executed.
3. Scanf is used to take inputs from the user. It reads formatted input from the standard input such as keyboard.
4. The c language is a case sensitive. This means that all language keywords, identifiers, function names, and other variables must be entered with consistent letter capitalization. C language can distinguish between upper case and lower case characters and treat the keywords and identifiers accordingly.
5.
 - a) Valid
 - b) Invalid – variable cannot start with a number
 - c) Invalid – we cannot use hyphen between variables
 - d) It's a keyword hence we cannot use it as a variable name
 - e) Invalid – we cannot use special characters
 - f) Valid
 - g) Invalid – there cannot be spaces for variables
 - h) Invalid – there cannot be hyphens during words
 - i) Valid
 - j) Invalid – we cannot represent even numbers also with hyphens

6.

- a) FALSE
- b) FALSE
- c) TRUE
- d) TRUE
- e) TRUE
- f) FALSE
- g) FALSE

7. *

```
**  
***  
****  
*****
```

8.

- a) `scanf(" %d ", &value);`
- b) `printf("The Product of %d and %d is %d \n" , x,y,z);`
- c) `scanf("%d" , &anInteger);`
- d) `printf("Remainder of %d divided by %d is\n", x, y, x % y);`
- e) `print("The sum is %d\n" , x + y);`
- f) `printf("The value you entered is: %d\n " , &value);`

9.

- a) Answer is 2
- b) Nothing will Print
- c) `x=`
- d) `x=2`
- e) `5 = 5`
- f) Nothing will print

- g) Nothing will print
- h) Nothing will print
- i) Nothing will print

10. S

- a) False, they are evaluated according to their precedence and associativity.
- b) True, because valid variable names are a series of underscores, letters and digits that do not start with a digit, and none of these conditions were violated.
- c) False, this statement will just print `a = 5;` to the screen. It does not perform any assignment in the program.
- d) False, it will be evaluated according to their precedence and associativity.
- e) False, `h22` is a valid variable name. All of the other variable names start with digits and are therefore invalid variable names.