

C Programming Language MCQs

C is a general-purpose, procedural computer programming language, **C language** supports structured programming, lexical variable scope, and recursion, with a static type system. **C language** is used to develop software like operating systems, databases, compilers, and so on.

C language MCQs: This section contains *multiple-choice questions and answers on the C programming language*. It will help the students to test their skills and prepare well for their exams.

1) C Basics MCQs

1. C language was developed by ____.

- A. Dennis Rechar
- B. Dennis M. Ritchie
- C. Bjarne Stroustrup
- D. Anders Hejlsberg

Answer: B) Dennis M. Ritchie

Explanation:

C programming language was developed by an American computer scientist Dennis M. Ritchie at Bell Laboratories (formerly AT&T Bell Laboratories).

2. In which year was C language developed?

- A. 1962
- B. 1978
- C. 1979
- D. 1972

Answer: D) 1972

Explanation:

C programming language was developed by Dennis Ritchie at Bell Laboratories in 1972.

3. C language is a successor to which language?

- A. Basic
- B. Cobol
- C. C++
- D. B

Answer: D) B

Explanation:

C programming language is a successor to the programming language B.

4. C is a ____.

- A. Low level language
- B. High level language
- C. Medium level language
- D. None of the above

Answer: C) Medium level language

Explanation:

C is a medium-level language because it contains the features of low-level language as well as high-level language.

5. How many keywords are there in C language?

- A. 32
- B. 33
- C. 64
- D. 18

Answer: A) 32

Explanation:

C language has only 32 keywords.

6. C language is a ____.

- A. Procedural oriented programming language
- B. General purpose programming language
- C. Structured programming
- D. All of the above

Answer: D) All of the above

Explanation:

C programming language is a general-purpose, procedural computer programming language that supports structured programming also.

7. Which is not a valid keyword in C language?

- A. for
- B. while
- C. do-while
- D. switch

Answer: C) do-while

Explanation:

do-while is not a valid keyword in the C programming language. It's a control statement. 'do' and 'while' are the separate keywords. The rest of all 'for', 'while', and 'switch' are the valid keywords in C.

8. What is an identifier in C language?

- A. An identifier is a combination of alphanumeric characters used for conditional and control statements
- B. An identifier is a combination of alphanumeric characters used for any variable, function, label name
- C. Both A and B
- D. None of the above

Answer: B) An identifier is a combination of alphanumeric characters used for any variable, function, label name

Explanation:

An identifier is a combination of alphanumeric characters used for any variable, function, label name. An identifier is a name that is used to identify the variables/ constants, functions, arrays, label name, and user-defined data

9. A C-style comment, simply surround the text with ____.

- A. /* and */
- B. // and //
- C. //
- D. /** and **/

Answer: A) /* and */

Explanation:

A [C-style comment](#), simply surround the text with /* and */.

/* */ is called multi-line comment.

// is called single line comment.

10. Can we place comments between the statement to comments a part of the code?

- A. Yes
- B. No

Answer: A) Yes

Explanation:

Yes, we can place comments between the statement to comments a part of the code.

Example:

```
printf(/*"Hello World"*/ "Hey, how are you?");
```

11. ____ is an informal name for ISO/IEC 9899:1999, a past version of the C programming language standard?

- A. C
- B. C++

- C. C89
- D. C99

Answer: D) C99

Explanation:

C99 is an informal name for ISO/IEC 9899:1999, a past version of the C programming language standard.

11. ___ is an informal name for ISO/IEC 9899:1999, a past version of the C programming language standard?

- A. C
- B. C++
- C. C89
- D. C99

Answer: D) C99

Explanation:

C99 is an informal name for ISO/IEC 9899:1999, a past version of the C programming language standard.

12. In which version of C language, the C++ Style comment (//) are introduced?

- A. C17
- B. C18
- C. C89
- D. C99

Answer: D) C99

Explanation:

C language version C99 introduced C++ style comment (//), they can be used to comment a single line.

13. The C source file is processed by the ___.

- A. Interpreter

- B. Compiler
- C. Both Interpreter and Compiler
- D. Assembler

Answer: B) Compiler

Explanation:

The C source file is processed by the compiler.

14. How many whitespace characters are allowed in C language?

- A. 2
- B. 3
- C. 4
- D. 5

Answer: D) 5

Explanation:

There are 5 whitespace characters are allowed in C language, they are:

- i. Space
- ii. Horizontal tab
- iii. Vertical tab
- iv. Form feed
- v. New-line

15. How many punctuation characters are allowed in C language?

- A. 29
- B. 30
- C. 31
- D. 32

Answer: A) 29

Explanation:

There are 29 punctuation characters are allowed in C language, they are:

_ { } [] # () < > % : ; . ? * + - / ^ & | ~ ! = , \ " '

16. What is the extension of a C language source file?

- A. .c
- B. .cpp
- C. .c99
- D. .h

Answer: A) .c

Explanation:

The extension of a C language source file is ".c".

17. What is the extension of a C language header file?

- A. .c
- B. .cpp
- C. .c99
- D. .h

Answer: D) .h

Explanation:

The extension of a C language source file is ".h".

18. To develop which operating, C language was invented?

- A. Linux
- B. Unix
- C. Android
- D. Mac

Answer: B) Unix

Explanation:

C language was invented to develop Unix operating system.

19. Does C language support object-oriented approach?

- A. Yes
- B. No

Answer: B) No

Explanation:

C language does not support object-oriented approach.

20. Which is/are the disadvantage(s) of C language?

- A. No Garbage Collection
- B. Inefficient Memory Management
- C. Low level of abstraction
- D. Lack of Object Orientation
- E. All of the above

Answer: E) All of the above

Explanation:

The main [disadvantages of C language](#) are:

- i. No Garbage Collection
- ii. Inefficient Memory Management
- iii. Low level of abstraction
- iv. Lack of Object Orientation

2) C Data Types, Operators and Expressions MCQs

21. Which are the fundamental data types in C?

- A. char
- B. int
- C. float
- D. All of the above

Answer: D) All of the above

Explanation:

The [fundamental / basic data types in C language](#):

- char
- int
- float

22. How many byte(s) does a char type take in C?

- A. 1
- B. 2
- C. 3
- D. 4

Answer: A) 1

Explanation:

The `char` data type takes one byte in the memory.

23. For which type, the format specifier "%i" is used?

- A. int
- B. char
- C. float
- D. double

Answer: A) int

Explanation:

In C programming language, both of the format specifier `%d` and `%i` are used for `int` type, where `%d` specifies the type of variable as decimal and `%i` specifies the type as integer.

24. What is the difference between float and double in C?

- A. both are used for the same purpose
- B. double can store just double value as compare to float value
- C. double is an enhanced version of float and was introduced in C99
- D. double is more precise than float and can store 64 bits

Answer: D) double is more precise than float and can store 64 bits

Explanation:

In C programming language, the `double` is more precise than `float` and can store 64 bits.

25. Which is the correct format specifier for double type value in C?

- A. `%d`
- B. `%f`
- C. `%lf`
- D. `%LF`

Answer: C) `%lf`

Explanation:

The `%lf` is used to represent a double type value in C programming language.

26. The short type represents ____.

- A. `int`
- B. `float`
- C. `unsigned int`
- D. `short int`

Answer: C) `unsigned int`

Explanation:

The `short` type represents `short int` in C language.

27. How many byte(s) does a short type take in C?

- A. 1
- B. 2
- C. 3
- D. 4

Answer: B) 2

Explanation:

In C programming language, the `short` or `short int` takes 2 bytes (16 bits) in memory.

28. What is the correct syntax to declare a variable in C?

- A. `data_type variable_name;`
- B. `data_type as variable_name;`
- C. `variable_name data_type;`
- D. `variable_name as data_type;`

Answer: A) `data_type variable_name;`

Explanation:

In C language, the correct syntax to [declare a variable](#) is:

```
data_type variable_name;
```

Where, *data_type* is the type of data (such as *int*, *char*, *float*, etc) and *variable_name* is a [valid identifier](#). For example: *int age;*

29. How many types of qualifiers are there in C language?

- A. 2
- B. 3
- C. 4
- D. 5

Answer: B) 3

Explanation:

There are 3 types of qualifiers in C language, they are:

- Size qualifiers
- Sign qualifiers
- [Type qualifiers](#)

30. Which is/are the size qualifier(s) in C language?

- A. `short`
- B. `long`
- C. `double`

D. Both A. and B

Answer: D) Both A. and B.

Explanation:

The size qualifiers are `short` and `long`.

31. Which is/are the sign qualifier(s) in C language?

- A. signed
- B. unsigned
- C. long
- D. Both A. and B

Answer: D) Both A. and B.

Explanation:

The sign qualifiers are used to specify the signed nature of an integer type. The sign qualifiers are `signed` and `unsigned`.

32. Which is/are the type qualifier(s) in C language?

- A. const
- B. volatile
- C. static
- D. Both A. and B

Answer: D) Both A. and B.

Explanation:

The type qualifiers are `const` and "volatile".

33. Which is correct with respect to the size of the data types in C?

- A. `char > int > float`
- B. `char < int < float`
- C. `int < char < float`
- D. `int < chat > float`

Answer: B) char < int < float

Explanation:

The correct order of the data types as per the size is: char < int < float.

34. Which operator is used to find the remainder of two numbers in C?

- A. /
- B. \
- C. %
- D. //

Answer: C) %

Explanation:

The % operator is known as "Modulus Operator" and it is used to find the remainder of two numbers.

35. Which of the following is not an arithmetic expression?

- A. x = 10
- B. x /= 10
- C. x %= 10
- D. x != 10

Answer: D) x != 10

Explanation:

x != 10 is not a valid arithmetic expression.

36. What will be the output of the following C code?

```
#include <stdio.h>
```

```
int main()
{
    int x = 20;
    x %= 3;
    printf("%d", x);

    return 0;
}
```

- A. 2
- B. 2.5
- C. Error
- D. Warning

Answer: A) 2

Explanation:

In the above code, the value of `x` is 20 and then in the next statement, the expression is `x %= 3`. That will be evaluate as:

```
x %= 3;
x = x % 3;
x = 20 % 3;
x = 2
```

37. What will be the output of the following C code?

```
#include <stdio.h>

int main()
{
    float x = 21.0;
    x %= 3.0;
    printf("%f", x);

    return 0;
}
```

- A. 7
- B. 7.00

- C. 7.000000
- D. Error

Answer: D) Error

Explanation:

In the above code, we are performing modulus operation with `float` values. Modulus operator doesn't work with `float` and `double` operands. Thus, the output will be:

```
error: invalid operands to binary % (have 'float' and 'double')
```

38. What will be the output of the following C code?

```
#include <stdio.h>

int main()
{
    float x = 23.456;
    printf("%.2f", x);
    return 0;
}
```

- A. 23.45600
- B. 23.456
- C. 23.45
- D. 23.46

Answer: D) 23.46

Explanation:

In the above code, the value of `x` is 23.456 and we are printing the value of `x` using the `%.2f` format specifier. `%.2f` rounds the value and prints the 2 digits after the decimal point.

39. What will be the output of the following C code?

```
#include <stdio.h>

void main()
{
    int x = 10;
    int y = x++ + 20;

    printf("%d,%d", x, y);

    return 0;
}
```

- A. 11,30
- B. 11,31
- C. 10,30
- D. 10,31

Answer: A) 11,30

Explanation:

In the above code, we are using a post-increment statement (`x++`), post-increment increases the value after evaluating the current expression. Thus, the value of `y` will be 30 and then `x` will be 11.

40. Increment (++) and decrement (--) are the ___ operators in C?

- A. Unary
- B. Binary
- C. Ternary
- D. None of the above

Answer: A) Unary

Explanation:

Increment (++) and decrement (--) are the unary operators. They need one operand to perform the operation.

41.What will be the output of the following C code?


```
#include <stdio.h>

int main()
{
    int x = -100;

    -100;
    printf("%d", x);

    return 0;
}
```

- A. 100
- B. -100
- C. 0
- D. Error

Answer: B) -100

Explanation:

The statement "-100;" is evaluated and this does not affect the value of "x".

42. Which C keyword is used to extend the visibility of variables?

- A. extend
- B. extends
- C. extern
- D. auto

Answer: C) extern

Explanation:

The "extern: keyword used to define an [extern variable](#), that can be accessed in any source file. i.e., extern is used to extend the visibility of variables in C language.

43. What is the name of "&" operator in C?

- A. Ampersand
- B. And

- C. Address of
- D. None of the above

Answer: C) Address of

Explanation:

The "&" operator is known as '[Address Of operator](#)' which is used to access the address of a variable.

C Conditional Statements MCQs

44. Which of the following are valid decision-making statements in C?

- A. if
- B. switch
- C. nested if
- D. All of these

Answer: D) All of these

Explanation:

All valid [decision-making statements in C](#) program are:

- if statement
- if-else statement
- nested if statement
- switch statement
- nested switch statement

Decision making in the C programming language is ____.

- A. Repeating the same statement multiple times
- B. Executing a set of statements based on some condition
- C. Providing a name of the block of code
- D. All of these

Answer: B) Executing a set of statements based on some condition

Explanation:

Decision-making in C programming is executing a block of code to be executed by the program based on some condition.

45. Which of the following is a true value in C programming?

- A. 1
- B. "includehelp"
- C. ! NULL
- D. All of these

Answer: D) All of these

Explanation:

All non-zero and non-null values in C programming are true.

46.What is the correct syntax of if statement in C program?

- A. if(condition){
 }
- B. if(condition) :
- C. If { [condition] }
- D. None of these

Answer: A)

```
if(condition) {  
}
```

Explanation:

The correct [syntax of if statement](#) in C program is:

```
if(condition) {  
    // code to be executed  
}
```

47. The if statement is a conditional statement?

- A. True
- B. False

Answer: A) True

Explanation:

The if statement is a conditional statement, i.e., the block is executed based on the given condition.

48. When the condition of if statement is false, the flow of code will ____.

- A. go into the if block
- B. Exit the program
- C. Continue the code after skipping the if block
- D. None of these

Answer: C) Continue the code after skipping the if block

Explanation:

When the condition of if statement is false, the code after the if block will be executed and the if block code is skipped.

49. What will be the result of the following condition?

```
(! (25 > 25))
```

- A. True
- B. False
- C. Error
- D. None of these

Answer: A) True

50. Which statement is required to execute a block of code when the condition is false?

- A. for
- B. if
- C. else
- D. All of these

Answer: C) else

Explanation:

In the if-else block, the if block is executed when condition is True and else block is executed when condition is false.

51. Can the else statement exist without the if statement in C?

- A. Yes
- B. No

Answer: B) No

Explanation:

The else statement needs an if statement to execute the block.

52. Which of these if...else block syntax is correct?

- A.

```
if(condition){  
    }  
    else {  
    }  
}
```
- B.

```
if(condition){  
    }  
    else(condition){  
    }  
}
```
- C.

```
if{  
    }  
}
```

```
    else {  
    }  
D. None of these
```

Answer: A)

```
if(condition) {  
}  
else {  
}
```

Explanation:

The [syntax of if...else](#) block of code in C is:

```
if(condition) {  
    // true block  
}  
else {  
    // false block  
}
```

53. The if-elseif-else statement in C programming is used?

- A. Create multiple conditional statements
- B. Return values
- C. Loop in if-else block
- D. All of these

Answer: A) Create multiple conditional statements

Explanation:

The [if-elseif-else statement](#) is a statement which contains multiple statements based on conditions.

54. What will be the output of the following C code?

```
#include <stdio.h>

int main()
{
    int marks = 43;

    if (marks > 90)
        printf("Grade : A ");
    else if (marks > 75)
        printf("Grade : B ");
    else if (marks > 60)
        printf("Grade : C ");
    if (marks > 40)
        printf("Grade : D ");
    else
        printf("Fail ");
    return 0;
}
```

- A. Grade : A
- B. Grade : B
- C. Grade : C
- D. Grade : D

Answer: D) Grade : D

55. How many expressions can be checked using if...elseif...else statement?

- A. 100
- B. 1
- C. Infinite
- D. None of these

Answer: C) Infinite

Explanation:

You can execute any number of expressions using if-elseif-else statements, each elseif statement containing one condition.

56. Is it possible to nest if-else statements in C programming?

- A. Yes
- B. No

Answer: A) Yes

Explanation:

Nesting of if-else statements in C programming is possible.

57. Which of the following syntax is correct for nested if-else statements?

A.

```
if(exp1){  
    if(exp2){  
    }  
}  
else {  
    if(exp3){  
    }  
}
```

B.

```
if(exp1){  
}else {  
}
```

C.

```
if{}
```

D. None of these

Answer: A)

```
if(exp1) {  
    if(exp2) {  
    }  
}  
else {  
    if(exp3) {  
    }  
}
```

Explanation:

The correct syntax for nested if-else statements in C programming is:

```
if (exp1) {  
    if (exp2) {  
    }  
}  
else {  
    if (exp3) {  
    }  
}
```

58. What will be the output of the following C code?

```
#include <stdio.h>  
  
int main() {  
    int n = 65;  
    if (n >= 75) {  
        if (n >= 95) {  
            printf("Excellent");  
        }  
        else  
            printf("Pass");  
    }  
    else  
        printf("Fail");  
}
```

- A. Excellent
- B. Pass
- C. Fail
- D. None of these

Answer: C) Fail

59. Multiple values of the same variable can be tested using ____.

- A. switch
- B. for
- C. Function

D. All of these

Answer: A) switch

Explanation:

The [switch statement in C](#) is used to test for multiple values of a variable.

60. Without a break statement in switch what will happen?

- A. All cases will work properly
- B. Cases will fall through after matching the first check
- C. Switch will throw error
- D. All of these

Answer: B) Cases will fall through after matching the first check

Explanation:

The [break statement](#) is used to terminate the current flow of code. And if it is not present in the switch statement, the cases will execute all cases after the matched case.

61. When all cases are unmatched which case is matched in a switch statement?

- A. Default case
- B. First case
- C. No case
- D. None of these

Answer: A) Default case

Explanation:

The default case of switch statement is executed when no other case is matched.

62. What will be the output of the following C code?

```
#include <stdio.h>

int main() {
    char grade = 'B';

    switch (grade) {
        case 'A':
            printf("Excellent!\n");
        case 'B':
        case 'C':
            printf("Well done\n");
        case 'D':
            printf("You passed\n");
        case 'F':
            printf("Better try again\n");
            break;
        default:
            printf("Invalid grade\n");
    }
}
```

- A. Well done
- B. You passed
- C. Better try again
- D. All of these

Answer: D) All of these

Explanation:

There is no break statement in case B, Case C, case D. the code will fall through executing all the print statements.

4) C Control Statements MCQs

63. Loops in C programming are used to ____.

- A. Execute a statement based on a condition
- B. Execute a block of code repeatedly
- C. Create a variable
- D. None of these

Answer: B) Execute a block of code repeatedly

Explanation:

[Loops in programming](#) are used to execute a block of code repeatedly.

64. Which of these is an exit-controlled loop?

- A. for
- B. if
- C. do...while
- D. while

Answer: C) do...while

Explanation:

The do...while loop check for a condition after executing the loop block once. Hence, it is called an exit-controlled loop.

65. Which statements are used to change the execution sequence?

- A. Loop control statement
- B. Function statement
- C. Conditional statement
- D. All of these

Answer: A) Loop control statement

Explanation:

Loop control statements in C are used to change the execution sequence of the loop.

66. What will happen if the loop condition will never become false?

- A. Program will throw an error
- B. Program will loop infinitely
- C. Loop will not run
- D. None of these

Answer: B) Program will loop infinitely

Explanation:

An infinite loop in a program is a condition when the loop continues when the loop continues to run infinitely because the condition never becomes false.

67. Which of these statements is correct in case of while loop in C?

- A. Executes the block till the condition become false
- B. Is an entry controlled loop
- C. There might be condition when the loop will not execute at all
- D. All of these

Answer: D) All of these

Explanation:

All of the above statements are true in the case of while loop. While loop is an entry-controlled loop, the block executes when the condition becomes false.

68. Which of the following is valid syntax for creating a while loop?

A.

```
while{  
} (condition)
```

B.

```
while(condition){  
}
```

C.

```
while{  
}
```

D. All of these

Answer: B)

```
while (condition) {  
}
```

Explanation:

The correct syntax for creating a while loop is:

```
while(condition){  
}
```

69. What will be the output of the following C code?

```
#include <stdio.h>  
  
int main() {  
    int a = 11;  
  
    while (a < 20) {  
        printf("%d ", a);  
        a += 2;  
    }  
  
    return 0;  
}
```

- A. 11 13 15 17 19
- B. 11 12 13 14 15 16 17 18 19 20
- C. 11 13 15 17 19 21
- D. None of these

Answer: A) 11 13 15 17 19

70. Which loop executes the block a specific number of times?

- A. while loop
- B. for loop
- C. do...while loop
- D. All of these

Answer: B) for loop

Explanation:

The for loop executes the block a specific number of times.

71. Which of the following parts of the for loop can be eliminated in C?

- A. init
- B. condition
- C. increment
- D. All of these

Answer: D) All of these

Explanation:

Syntax of for loop:

```
for(init, condition, increment){  
}
```

Inside the initialization statement (init), any of the three init or condition or increment can be eliminated i.e., all are optional. The loop can work without them also.

72. When all parts of the for loop are eliminated, what will happen?

- A. For loop will not work
- B. Infinite for loop
- C. Error
- D. None of these

Answer: B) Infinite for loop

Explanation:

On eliminating all the parts of a for loop [an infinite for loop](#) will run.

73. When the condition of the do-while loop is false, how many times will it execute the code?

- A. 0
- B. 1
- C. Infinite
- D. All of these

Answer: B) 1

Explanation:

The do-while loop is an exit-controlled loop, hence it will run at least once, even if the condition becomes false.

74. Can a loop be nested in C programming?

- A. Yes
- B. No

Answer: A) Yes

Explanation:

Yes, the [nesting of loop](#) is possible in C programming language.

75. What will be the output of the following C code?

```
#include <stdio.h>

int main() {
    int i, j;

    for (i = 2; i < 10; i++) {
        for (j = 2; j <= (i / j); j++)
            if (!(i % j))
                break;
        if (j > (i / j))
            printf("%d ", i);
    }

    return 0;
}
```


- A. 2 3 4 5 6 7 8 9
- B. 3 5 7 9
- C. 2 3 5 7
- D. 2 3 5 7 11

Answer: C) 2 3 5 7

Prepared By: Md. Abu Zafor

CSE Dept. 8th semester, EWU

