

SHORT QUESTIONS

Q.1 What are Identifiers?

Ans. Identifiers are the names used to represent variables, constants, data types, functions, and labels in the program. Identifiers consist of characters (alphabets, numbers, underscore).

Q.2 Define Standard identifiers

Ans. These identifiers have special meaning (defined operations) in C like reserve words. These can be redefined for other purposes but it is not recommended.

Q.3 Define User-defined identifiers:

Ans. These identifiers are defined by the programmer (user). A user uses these identifiers for variables, constants, data types, functions, and labels in the program. These words are used to store or access data from various memory locations.

Q.4 What are Keywords/ Reserve words?

Ans. These are the words which have predefined meaning in C language. There are 32 keywords in C language. These words can not be used or redefined for any other purpose in the C program. All keywords are written in lower case e.g. double, long, void etc.

Q.5 What are Variables?

Ans. A variable is a name of memory location. Variables are used to store values that can be changed during program execution. The value of variables may be numeric or alphabets. Variable name remains same but the value stored in variable may change during the program execution.

Q.6 What is a Variable definition?

Ans. It means to set aside the memory location for the variable.

Q.7 What is Variable Declaration?

Ans. Variable declaration tells the compiler the name and type of value stored in the variable. Before using a variable in a program, it must be declared. When a variable is declared, a certain number of bytes (depending upon the data type) are allocated to the variable in memory. It means declaration not only declares but also defines a variable.

Syntax: Data_type variable names;
 int a, b,c;

Q.8 Define Variable Initialization?

Ans. Assigning a value to a variable at the time of declaration is called initialization of variable.

Q.9 What is a Garbage Value?

Ans. When a variable is declared, the compiler reserves the space for it. If we do not initialize it then it may contain a meaningless data is called garbage value, and with the involvement of such variable may cause unexpected results. To avoid this situation, all variables must be declared and initialized according to program requirement.

Q.10 Define Constants.

Ans. The quantities whose values cannot be changed during program execution. There are two types of constants: Numeric constants and character constants.

Example: `#define pi 3.142857`

pi whose value remains same during program execution.

Q.11 What is a Numeric Constant?

Ans. These constants consist of numbers. There are two types of numeric constants. Integers and float; Integer constants represent values that are counted and without decimal or fractional part. e.g. +56, -678 etc and floating constants represents values that are measured. e.g. 4.786, 0.45 etc.

Q.12 What is a Character Constant?

Ans. It is a single alphabet, a single digit or a single symbol enclosed within apostrophes. The maximum length of a character constant is 1. Examples are: '5', '+', 'B' etc.

Q.13 What is a Data Type?

Ans. Data type defines maximum or minimum set of values and set of operations on values.

Q.14 What is a Standard data type?

Ans. It is a data type which is predefined in C language. In C language there are four basic standard data types (int, float, char and double).

Q.15 What is User-define data type?

Ans. It allows us to define our own data types.

Q.16 What is Character data type?

Ans. A keyword char is used for character data type. It is used to represent a letter, number, or a symbol. A character variable occupies 1 byte in memory. %c is used as format specifier. A character is enclosed in apostrophes. e.g. 'X', '5', '=', '#' etc.

Q.17 What are integers?

Ans. Integers are the numbers without decimal fraction. It may be positive, negative or zero. e.g. 112, -234, 0 etc. An integer variable may be signed or unsigned. If not mentioned then all integers are considered as signed.

Q.18 What are Data Types for Integers?

Ans. int.
short int.
long int.

Q.19 What is int data type?

Ans. It takes 2 bytes in memory. The keyword int is used for integers. There are two further sub-types of int variables.

Signed int
Unsigned int

Q.20 What are Signed int or Short int data type?

Ans. It takes 2 bytes in memory. The keyword signed int and short int are used for integers. Range of int, signed int or short int variables are -2^{15} to $2^{15}-1$ or -32768 to 32767. int, signed int and short int can handle both positive and negative numbers. %d or %i are used as format specifiers.

Q.21 What are unsigned int or unsigned short int?

Ans. It can not handle negative numbers. It takes 2 bytes in memory. The keyword unsigned int is used for integers. It can handle numbers ranging from 0 to $2^{16} - 1$ or 0 to 65535. %u is the format specifier for unsigned int.

Q.22 What is long int data type?

Ans. It is used to represent larger integers. It occupies 4 bytes in memory. It can hold numbers ranging from -2^{31} to $2^{31}-1$ (i.e. -2147483648 to 2147483647) Unsigned long int: can hold numbers ranging from 0 to $2^{32} - 1$ (i.e. 0 to 4294967295).

Q.23 What are float numbers and what are data Types for Floating Point Numbers?

Ans. These are the numbers with fractional part. e.g. 2.13, 0.54 etc. ANSI C specifies three floating point data types and All are different in memory requirement and range.

float
double
long double

Q.24 What is float data type?

Ans. Float may be signed or unsigned numbers. It is represented in decimal or exponential form. It occupies 4 bytes in memory. It can store real values up to 3.4×10^{-38} to $3.4 \times 10^{+38}$. Its accuracy is up to 6 decimal places.

Q.25 What is Scientific or Exponential form of float point number?

Ans. It is a floating point number and the storage area occupied by the number is divided into two sections. Mantissa and Exponent; Mantissa is the value of the number and exponent is the power to which mantissa is raised. In C, scientific forms of a number are +mep or -mep or +mEp, -mEp
m represents the mantissa part, p represents the exponent part, E or e is used instead of base 10, If the number is smaller than 1 then its exponent is negative. (i.e. 0.00524 is equal to 5.24xE-3)

Q.26 What is double data type?

Ans. It is used to store larger floating point numbers. It takes 8 bytes in memory. It can handle real numbers from 1.7×10^{-308} to $1.7 \times 10^{+308}$. Its accuracy is up to 15 decimal places.

Q.27 What is long double data type?

Ans. A long double variable takes 10 bytes in memory. It can handle real numbers from 3.4×10^{-4932} to $3.4 \times 10^{+4932}$. Its accuracy is up to 19 decimal places.

Q.28 What is a Cancellation Error?

Ans. While working with floating point numbers, some unexpected results cause problems. For example manipulation of very large and very small float numbers produced unexpected results e.g. the result of addition of 1970.0 and 0.0000001243 may produce 1970.000000 on some computers. This type of errors is called cancellation errors i.e. in cancellation errors a very small value disappears when it is added to or subtracted from a very large value.

Q.29 What is Arithmetic Underflow?

Ans. When two very small numbers are manipulated, the result may be too small to be represented accurately, so it will be represented as zero. This is called arithmetic underflow i.e. when the value assigned is less than the minimum allowable limit, an underflow occurs.

Q.30 What is Arithmetic Overflow?

Ans. When two very large numbers are manipulated, the result may be too large to be represented. This is called arithmetic overflow i.e. When the value assigned is greater than the maximum allowable limit, an overflow occurs.

Q.31 What is Integer Underflow?

Ans. When the value assigned to a variable is less than the minimum allowable limit of an integer i.e. -32768, an underflow occurs. In case of an integer underflow different compiler takes different actions. But in most cases compiler runs the programs and it produces incorrect results.

Q.32 What is Integer Overflow?

Ans. When the value assigned to a variable is greater than the maximum allowable limit of an integer i.e. +32767, an integer overflow occurs. In case of an integer overflow different compiler takes different actions. But in most cases compiler runs the programs and it produces incorrect results.

Q.33 What are Comments?

Ans. It is used to add informative notes about the statements in a program that helps in debugging and modifying programs. It is a non executable statement. Comments play no role in the execution of the program. The compiler does not translate these statements. It also explains the logic of the program. There are two types of comments in C.

Single line comments

Multi line Comments

Q.34 What are Single line comments?

Ans. Single line comments are inserted by typing two forward slashes before beginning of a line.

Example:

// this program Calculates temperature

Q.35 What is Multi line Comments:

Ans. Multi-line comments are used to provide such informative notes which extend to multi-lines. Multi line comments are inserted by typing slash asterisk (/*) in the beginning and asterisk slash (*/) at the end of paragraph. Then all lines within this block are ignored by the compiler. By omitting ending letters */ will cause the whole program code beneath the opening letters /* for comments to be commented.

Example: /* this program calculates factorial of a number

Number is positive integer. And result is long integer. */

Q.36 What is an Expression?

Ans. An expression is a combination of Operands (constants and variables) and operators. There are two ways to write an expression.

Operand1 operator Operand2

Operator (operand)

There are three types of Expressions

Arithmetic Expressions

Relational Expressions

Logical Expressions

Q.37 What is an operator?

Ans. It is a symbol or a sign used to perform a certain tasks on data. Unary operators required single data item. Binary operators required two data items. Following types of operators are available in C.

Arithmetic Operators, Relational Operators, Logical Operators, Increment and decrement Operators, Assignment Operator, Bitwise operators.

Q.38 What are an Arithmetic Expressions and Arithmetic Operators?

Ans. Arithmetic operators are used with constants and variables to form arithmetic expressions. These operators (+, -, *, /, %) are used to perform arithmetic operations. All arithmetic operators are used for all types of numeric data except remainder operator. Remainder operator or Modulus Operator (%) is used only with integers.

Q.39 What are the modes of arithmetic expressions?

Ans. Integer Arithmetic
Real Arithmetic
Mixed arithmetic

Q.40 What is Integer arithmetic Mode?

Ans. When an arithmetic operation is performed on integer values it always yields an integer result. It does not produce real results.

Q.41 What is Real arithmetic Mode?

Ans. When an arithmetic operation is performed on floating point numbers, it always yields a floating point result.

Q.42 What is mixed arithmetic mode?

Ans. When an arithmetic operation is performed on integers and floating point numbers (anyone is real number), it always yields a floating point result.

Q.43 What are Relational Expression and Relational Operators?

Ans. Relational operators with constants and variables form relational expressions. These are six operators (<, >, <=, >=, ==, !=) used to compare two values. These always evaluates to true or false. A true state is represented by a non zero value (1). A false state is represented by a zero value (0).

Q.44 What are Logical Expressions and Logical Operators?

Ans. These operators are used to combine more than one relational expression i.e. relational expressions with logical operators form logical expressions. And (&&), or (||) and not (!) are three logical operators. Logical operators also produce true or false results. Not (!) is a unary operator but && and || are binary operators.

Q.45 What is And Operator (&&)?

Ans. It is used to combine two or more relational expressions. It produces true if all the relational expressions (conditions) are true.

Q.46 What is Or Operator (||):

Ans. It is also used to combine two or more relational expressions. It produces false if all the conditions (relational expressions) are false. It produces true result if any one of the conditions or all the conditions are true.

Q.47 What is Not or Negation Operator (!)?

Ans. It negates the value of an expression. If an expression evaluates to true then it converts true into false. If an expression produces false then its negation yields true.

Q.48 What is Increment Operator (++):

Ans. ++ is used as increment operator. It is used to increase the value of the variable by one. It can be used before or after the variable name e.g. x++ or ++x. It cannot be used with constants and expressions. Only the variables can be incremented. It is unary operator. There are two types of increment operator

Postfix increment operator

Prefix increment operator.

Q.49 What is a Postfix increment operator?

Ans. When ++ is follows its operand (variable), it is called postfix increment operator. In postfix increment, ++ increases the value of variable after the execution of the statement i.e. it first uses the current value of the variable in the statement and after the completion of the statement it increases the value by one.

Q.50 What is a Prefix increment operator?

Ans. When ++ is precedes its operand (variable), it is called prefix increment operator. In prefix increment, ++ increases the value of the variable before the execution of the statement i.e. it first increases the current value of the variable by one and then new value is used in the statement.

Q.51 What is Decrement Operator (--):

Ans. -- is used as decrement operator. It is used to decrease the value of the variable by one. It can be used before or after the variable name e.g. x-- or --x. It cannot be used with constants and expressions. Only the variables can be decremented. It is unary operator. There are two types of decrement operator

Postfix decrement operator

Prefix decrement operator.

Q.52 What is a Postfix decrement operator?

Ans. When -- is follows its operand (variable), it is called postfix decrement operator. In postfix decrement, -- decreases the value of variable after the execution of the

statement i.e. it first uses the current value of the variable in the statement and after the completion of the statement it decreases the value by one.

Q.53 What is Prefix decrement operator?

Ans. When -- is precedes its operand (variable), it is called prefix decrement operator. In prefix decrement, -- decreases the value of the variable before the execution of the statement i.e. it first decreases the current value of the variable by one and then new value is used in the statement.

Q.54 What are Assignment Statement and Assignment Operator (=):

Ans. Assignment statement is used to assign a value or a computational result to a variable. The symbol equal (=) represents assignment operator e.g. a=2;

There are two types of assignment statement.

Simple Assignment Statement

Compound Assignment Statement

Q.55 What is a Simple Assignment statement?

Ans. The value is written on the right side of the operator and the variable is left side of the operator in the assignment statement. Writing variable to right and value to left causes a syntax error. General form of assignment statement is:

Variable = expression

Expression can be a variable, constant, arithmetic, relational or logical expression

Q.56 What is Compound Assignment statement?

Ans. It is used to assign one value to more than one variable. General form of compound Statement

Var-1 = Var-2 = Var-3 = ----- = Var-n = Value

Q.57 What are Compound Assignment Expression and Compound Assignment Operators?

Ans. It is used to add, subtract, multiply or divide the value to or from a variable. There are four compound assignment operators (+=, -=, *=, /=) that can increment or decrement the value of the variables by other than one. General form of writing compound assignment expressions is

Variable operator = value

Q.58 What is Operator Precedence?

Ans. It determines the order of evaluation in an expression. Each operator has a fixed order. The operator with higher priority evaluated before those with lower priority. When an expression is a combination of arithmetic, relational and logical expression then the expression is evaluated in the following sequence.

Arithmetic

Relational

Logical

taleemcity.com