

# MCQ

1. Which of the following language is the predecessor to C Programming Language?

- a) A
- b) B
- c) BCPL
- d) C++

Ans: c

2. C programming language was developed by

- a) Dennis Ritchie
- b) Ken Thompson
- c) Bill Gates
- d) Peter Norton

Ans: a

3. C was developed in the year \_\_

- a) 1970
- b) 1972
- c) 1976
- d) 1980

Ans: b

4. C is a \_\_ language

- a) High Level
- b) Low Level
- c) Middle Level
- d) Machine Level

Ans: c

5. C language is available for which of the following Operating Systems?

- a) DOS
- b) Windows
- c) Unix
- d) All of these

Ans: d

6. Which of the following symbol is used to denote a pre-processor statement?

- a) !
- b) #
- c) ~

d) ;

Ans: b

7. Which of the following is a Scalar Data type

- a) Float
- b) Union
- c) Array
- d) Pointer

Ans: a

8. Which of the following are tokens in C?

- a) Keywords
- b) Variables
- c) Constants
- d) All of the above

Ans: d

9. What is the valid range of numbers for int type of data?

- a) 0 to 256
- b) -32768 to +32767
- c) -65536 to +65536
- d) No specific range

Ans: b

10. Which symbol is used as a statement terminator in C?

- a) !
- b) #
- c) ~
- d) ;

Ans: d

11. Which escape character can be used to begin a new line in C?

- a) \a
- b) \b
- c) \m
- d) \n

Ans: d

12. Which escape character can be used to beep from speaker in C?

- a) \a

b) \b

c) \m

d) \n

Ans: a

13. Character constants should be enclosed between \_\_

a) Single quotes

b) Double quotes

c) Both a and

b d) None of these

Ans: a

14. String constants should be enclosed between \_\_

a) Single quotes

b) Double quotes

c) Both a and b

d) None of these

Ans: b

15. Which of the following is invalid?

a) "

b) ""

c) 'a'

d) 'abc'

Ans: d

16. The maximum length of a variable in C is \_\_

a) 8

b) 16

c) 32

d) 64

Ans: a

17. What will be the maximum size of a float variable?

a) 1 byte

b) 2 bytes

c) 4 bytes

d) 8 bytes

Ans: c

18. What will be the maximum size of a double variable?

- a) 1 byte
- b) 4 bytes
- c) 8 bytes
- d) 16 bytes

Ans: c

19. A declaration float a,b; occupies \_\_ of memory

- a) 1 byte
- b) 4 bytes
- c) 8 bytes
- d) 16 bytes

Ans: c

20. The size of a String variable is

- a) 1 byte
- b) 8 bytes
- c) 16 bytes
- d) None of these

Ans: d

21. Which of the following is an example of compounded assignment statement?

- a) a=5
- b) a+=5
- c) a=b=c
- d) a=b

Ans: b

22. The operator && is an example for \_\_ operator.

- a) Assignment
- b) Increment
- c) Logical
- d) Rational

Ans: c

23. The operator & is used for

- a) Bitwise AND
- b) Bitwise OR
- c) Logical AND

d) Logical OR

Ans: a

24. The operator / can be applied to

- a) integer values
- b) float values
- c) double values
- d) All of these

Ans: b

25. The equality operator is represented by

- a) :=
- b) .EQ.
- c) =
- d) ==

Ans: d

26. Operators have hierarchy. It is useful to know which operator

- a) is most important
- b) is used first
- c) is faster
- d) operates on large numbers

Ans: b

27. The bitwise AND operator is used for

- a) Masking
- b) Comparison
- c) Division
- d) Shifting bits

Ans: a

28. The bitwise OR operator is used to

- a) set the desired bits to 1
- b) set the desired bits to 0
- c) divide numbers
- d) multiply numbers

Ans: a

29. Which of the following operator has the highest precedence?

- a) \*

b) ==

c) =>

d) +

Ans: d

30. The associativity of ! operator is

a) Right to Left

b) Left to Right

c) (a) for Arithmetic and (b) for Relational

d) (a) for Relational and (b) for Arithmetic

Ans: a

31. Which operator has the lowest priority?

a) ++

b) %

c) +

d) ||

Ans: d

32. Which operator has the highest priority?

a) ++

b) %

c) +

d) ||

Ans: a

33. Operators have precedence. Precedence determines which operator is

a) faster

b) takes less memory

c) evaluated first

d) takes no arguments

Ans: c

34. Integer Division results in

a) Rounding the fractional part

b) Truncating the fractional part

c) Floating value

d) An Error is generated

Ans: b

35. Which of the following is a ternary operator?

- a) ?:
- b) \*
- c) sizeof
- d) ^

Ans: a

36. What will be the output of the expression  $11 \wedge 5$ ?

- a) 5
- b) 6
- c) 11
- d) None of these

Ans: d

37. The type cast operator is

- a) (type)
- b) cast()
- c) (;;)
- d) // "

Ans: a

38. Explicit type conversion is known as

- a) Casting
- b) Conversion
- c) Disjunction
- d) Separation

Ans: a

39. The operator + in  $a+=4$  means

- a)  $a=a+4$
- b)  $a+4=a$
- c)  $a=4$
- d)  $a=4+4$

Ans: a

40.  $p++$  executes faster than  $p+1$  because

- a) p uses registers
- b)  $p++$  is a single instruction
- c)  $++$  is faster than +

d) None of these

Ans: b

41. Which of the following statements is true?

- a) C Library functions provide I/O facilities
- b) C inherent I/O facilities
- c) C doesn't have I/O facilities
- d) Both (a) and (c)

Ans: a

42. Header files in C contain

- a) Compiler commands
- b) Library functions
- c) Header information of C programs
- d) Operators for files

Ans: b

43. Which pair of functions below are used for single character I/O.

- a) Getchar() and putchar()
- b) Scanf() and printf()
- c) Input() and output()
- d) None of these

Ans: a

44. The printf() function retunes which value when an error occurs?

- a) Positive value
- b) Zero
- c) Negative value
- d) None of these

Ans: c

45. Identify the wrong statement

- a) putchar(65)
- b) putchar('x')
- c) putchar("x")
- d) putchar('\n')

Ans: c

46. Which of the following is character oriented console I/O function?

- a) getchar() and putchar()

- b) gets() and puts()
- c) scanf() and printf()
- d) fgets() and fputs()

Ans: a

47. The output of printf("%u", -1) is

- a) -1
- b) minimum int value
- c) maximum int value
- d) Error message

Ans: c

48. An Ampersand before the name of a variable denotes

- a) Actual Value
- b) Variable Name
- c) Address
- d) Data Type

Ans: c

49. Symbolic constants can be defined using

- a) # define
- b) const
- c) symbols
- d) None of these

Ans: b

50. Null character is represented by

- a) \n
- b) \0
- c) \o
- d) \e

Ans: b

51. Which header file is essential for using strcmp() function?

- a) string.h
- b) strings.h
- c) text.h
- d) strcmp.h

Ans: a

52. malloc() function used in dynamic allocation is available in which header file?

- a) stdio.h
- b) stdlib.h
- c) conio.h
- d) mem.h

Ans: b

53. File manipulation functions in C are available in which header file?

- a) streams.h
- b) stdio.h
- c) stdlib.h
- d) files.h

Ans: d

54. C supports how many basic looping constructs

- a) 2
- b) 3
- c) 4
- d) 6

Ans: b

55. A statement differs from expression by terminating with a

- a) ;
- b) :
- c) NULL
- d) .

Ans: a

56. What should be the expression return value for a do-while to terminate

- a) 1
- b) 0
- c) -1
- d) NULL

Ans: b

57. Which among the following is an unconditional control structure

- a) do-while
- b) if-else
- c) goto

d) for

Ans: c

58. Continue statement is used

- a) to go to the next iteration in a loop
- b) come out of a loop
- c) exit and return to the main function
- d) restarts iterations from the beginning of the loop

Ans: a

59. Which operator in C is called a ternary operator

- a) if..then
- b) ++
- c) ?:
- d) ()

Ans: c

60. Which of the following header file is required for strcpy() function?

- a) string.h
- b) strings.h
- c) files.h
- d) strcsspy()

Ans: a

61. The meaning of conversion character for data input is

- a) Data item is a long integer
- b) Data item is an unsigned decimal integer
- c) Data item is a short integer
- d) None of the above

Ans: c

62. The conversion characters for data input means that the data item is

- a) An unsigned decimal integer
- b) A short integer
- c) A hexadecimal integer
- d) A string followed by white space

Ans: b

63. An expression contains relational, assignment and arithmetic operators. If Parenthesis are not present, the order will be

- a) Assignment, arithmetic, relational
- b) Relational, arithmetic, assignment
- c) Assignment, relational, arithmetic
- d) Arithmetic, relational, assignment

Ans: d

64. Which of the following is a keyword used for a storage class

- a) printf
- b) external
- c) auto
- d) scanf

Ans: c

65. In the C language 'a' represents

- a) a digit
- b) an integer
- c) a character
- d) a word

Ans: c

66. The number of relational operators in the C language is

- a) Four
- b) Six
- c) Three
- d) One

Ans: b

67. A compound statement is a group of statements included between a pair of

- a) double quote
- b) curly braces
- c) parenthesis
- d) a pair of /'s

Ans: a

68. A Link is

- a) a compiler
- b) an active debugger
- c) a C interpreter
- d) an analyzing tool in C

Ans: d

69. The continue command cannot be used with

- a) for
- b) switch
- c) do
- d) while

Ans: a

70. In C, a Union is

- a) memory location
- b) memory store
- c) memory screen
- d) None of these

Ans: b

71. When the main function is called, it is called with the arguments

- a) argc
- b) argv
- c) None of these
- d) both a & b

Ans: d

72. A multidimensional array can be expressed in terms of

- a) array of pointers rather than as pointers to a group of a contiguous array
- b) array without the group of a contiguous array
- c) data type arrays
- d) None of these

Ans: a

73. C allows arrays of greater than two dimensions, who will determine this

- a) programmer
- b) compiler
- c) parameter
- d) None of these

Ans: b

74. A pointer to a pointer is a form of

- a) multiple indirections
- b) a chain of pointers
- c) both a and b

d) None of these

Ans: c

75. Pointers are of

- a) integer data type
- b) character data type
- c) unsigned integer data types
- d) None of these

Ans: d

76. Maximum number of elements in the array declaration int a[5][8] is

- a) 28
- b) 32
- c) 35
- d) 40

Ans: d

77. If the size of the array is less than the number of initializers then,

- a) Extra values are being ignored
- b) Generates an error message
- c) Size of Array is increased
- d) Size is neglected when values are given

Ans: a

78. Array subscripts in C always start at

- a) -1
- b) 1
- c) 0
- d) Value provided by the user

Ans: c

79. A Structure

- a) can be read as a single entity
- b) cannot be read as a single entity
- c) can be displayed as a single entity
- d) has member variables that cannot be read individually

Ans: b

80. Which is the correct way to declare a pointer?

- a) int\_ptr;

- b) int \*ptr;
- c) \*int ptr;
- d) None of these.

Ans: b

81. If you want to exchange two rows in a two-dimensional array, the fastest way is to:

- a) Exchange the elements of the 2rows
- b) Exchange the address of each element in the two-row
- c) Silence the address of the rows in an array of the pointer and exchange the pointer
- d) None of these.

Ans: c

82. A typecast is used to

- a) Define a new data type
- b) Force a value to be a particular variable type
- c) Rename an old type
- d) None of these.

Ans: b

83. Operator precedence determines which operator

- a) Operators on the largest number
- b) Is used first
- c) Is most important
- d) None of these.

Ans:b

84. If you don't initialize a static array, what will be the element set to?

- a) Zero
- b) A floating-point
- c) An undetermined value
- d) None of these.

Ans: a

85. Which is more appropriate for reading in a multi-word string?

- a) gets( )
- b) Printf( )
- c) scanf( )

d) puts ( ).

Ans: a

86. The process of translating a source program into machine language is a

function

of:

- a) Compiler
- b) Translator
- c) Assembler
- d) None of these.

Ans: a

87. Function argument can be

- a) A structure member
- b) A pointer variable
- c) A complete structure
- d) All of the above.

Ans: d

88. A "switch" statement is used to:

- a) Switch between user-defined functions in a program
- b) Switch from one variable to another variable
- c) Jump from one place to another in a program.
- d) None of these.

Ans: d

89. Consider the foll statement: "using C language programmers can write their own library functions".

- a) True
- b) False
- c) Maybe
- d) None of these.

Ans: a

90. C is a \_\_\_\_\_ level programming language?

- a) Low
- b) High
- c) Middle
- d) None of these.

Ans: c

91. A function is a subroutine that may include one or more \_\_\_\_\_ designed to perform a specific task.

- a) Functions
- b) Statements
- c) Libraries
- d) Datatypes.

Ans: b

92. What is used as a terminator in C?

- a) ?
- b) ;
- c) :
- d) \_

Ans: b

93. Which function is necessary to exist in each & every program?

- a) void
- b) sum
- c) main
- d) None of these.

Ans: c

94. What is the answer of: 7%3

- a) 2.5
- b) 1
- c) 2
- d) 3

Ans: b

95. The \_\_\_\_\_ chars have values from -128 to 127.

- a) signed
- b) unsigned
- c) long
- d) none

Ans: a

96. What is the control character for "a single character".

- a) %c
- b) %d
- c) %i

d) %p

Ans: a

97. What is the control character for "a decimal integer".

a) %c

b) %d

c) %i

d) %p

Ans: b

98. What is the control character for "a floating point number".

a) %c

b) %d

c) %i

d) %f

Ans: d

99. C supports the \_\_\_\_\_ statement to branch unconditionally from one point to another in the program.

a) continue

b) goto

c) break

d) for

Ans: b

100. The \_\_\_\_\_ is used to break out of the case statements.

a) continue

b) break

c) default

d) case

Ans: b

\_\_\_\_\_ in programming languages are used to make programs more readable by naming actions to be performed.

- a. Special words
- b. Statements
- c. Operators
- d. Labels

Direct descendants of C alongside with itself do not allow subprogram nesting.

- a. False, except C.
- b. True for all.
- c. True, except C.
- d. False for all.

The scopes created by blocks, which could be nested in larger blocks, are treated exactly like those created by subprograms.

- a. It is a true sentence.
- b. It is a false sentence.
- c. Cannot be compared the two thing.
- d. It depends on the language.

\_\_\_\_\_ can have \_\_\_\_\_ entry points, which are controlled by the \_\_\_\_\_ themselves.

- a. Coroutines , multiple , subroutines
- b. Coroutines , multiple , coroutines
- c. Subroutines, single, subroutines
- d. Subroutines, multiple, subroutines

Consider the following example:

```
void func( int i, int j)
```

- a. It is a subprogram header.
- b. It is a subprogram protocol.
- c. It is a subprogram profile.
- d. It is a subprogram definition.

The \_\_\_\_\_ of a statement in a \_\_\_\_\_ language is the locally declared variables, plus the variables of all other subprograms that are currently active.

- a. referencing environment, static scoped
- b. referencing environment, dynamically scoped
- c. referencing block, dynamically scoped
- d. referencing block, static scoped

Consider the following C function

```
void swap (int a, int b)
{
    int temp;
    temp = a;
    a = b;
    b = temp;
}
```

In order to exchange the values of two variables x and y:

- a. swap(x,y) cannot be used as the parameters are passed by value
- b. swap(x,y) cannot be used as it does not return any value
- c. Call swap (x, y)
- d. Call swap (&x, &y)

Pure interpretation is often requires more space than compilation.

- a. True, because there is the intermediate code generator.
- b. True, because it translates programs into an intermediate form.
- c. True, because e.g. the symbol table need to be available.
- d. False, there is no need for more space,

Determine output:

```
void main()
{
    int i=10;
    i = !i>14;
    printf("i=%d", i);
}
```

The name for a memory location that may hold data is \_\_\_\_\_.

- a. address
- b. variable
- c. pointer
- d. storage
- b. 0
- c. 14
- d. None of these.
- e. 10

[Clear my choice](#)

### Output?

```
int var = 20;
int main()
{
{
    int var = var;
}
{
    printf("%d", var);
}
return 0;
}
```

The pseudocode below demonstrates a loop.  
The code segment is an example of which type of loop?

```
num=10
while(num<20){
    print num;
    num += 1
}
```

- a. Compiler Error
- b. 20
- c. Garbage Value
- d. 10
- a. Counter-controlled loop
- b. Logically controlled loop
- c. Statement-controlled loop
- d. Infinite loop

In the below statement, ptr1 and ptr2 are uninitialized pointers to int i.e. they are pointing to some random address that may or may not be valid address.

```
int* ptr1, ptr2;
```

- a. True, because they are in the same statement.
- b. False, because they are different type variables.
- c. True, because they are not initialized.
- d. False, because they are pointing to the null address.

In C, parameters are always

- a. Passed by value result
- b. Passed by value
- c. Non-pointer variables are passed by value and pointers are passed by reference
- d. Passed by reference

Are there any difference between variable declaration and variable definition?

- a. A declaration occurs once, but a definition may occur many times.
- b. Both can occur multiple times, but a declaration must occur first.
- c. Both can occur multiple times, but a definition must occur first.
- d. There is no difference between them.
- e. A definition occurs once, but a declaration may occur many times.

3. In the following program fragment, a <= b will be printed if?

```
if(a > b)
    printf("a > b");
else
    printf("else part");
printf("a <= b");
```

- a. a < b
- b. a > b
- c. does not depend upon if condition
- d. a == b

4. In the below statement, ptr1 and ptr2 are uninitialized pointers to int i.e. they are pointing to some random address that may or may not be valid address.

```
int* ptr1, ptr2;
```

- a. False, because they are pointing to the null address.
- b. True, because they are in the same statement.
- c. True, because they are not initialized.
- d. False, because they are different type variables.

5. What will be output of the following program?

```
void main()
{
    int a=7,b=2,c=10;
    c=a=b;
    printf("%d",c);
}
```

- a) 2
- b) 10

- c) 7
- d) 0

# Programming Languages 1 - 2022

December 20, 2022.

Name: \_\_\_\_\_ Neptun ID: \_\_\_\_\_

COURSE CODE: INBPA0211E or INBMA0211E or INGK301 or INJK301 or INHK301

## Declaration

*By understanding this declaration, I admit that I have noticed that any kinds of devices or materials are*

*NOT allowed and NOT acceptable on this exam, and I agree to that. I accept all the ethical rules of the University.*

*I admit that the voice and/or video might be recorded during my exam.*

*Without my signature this exam can NOT be graded and counted in to my final grade.*

\_\_\_\_\_  
*signature*

## Part I. – Test

1. In the following program fragment, s2 will be executed if?

```
if(a > b)
    if(b > c)
        s1;
    else
        s2;
```

- a. a > b and b <= c
- b. a <= b
- c. b <= c and a <= b
- d. b > c

2. Direct descendants of C alongside with itself do not allow subprogram nesting.
  - a. False, except C.
  - b. True, except C.
  - c. True for all.
  - d. False for all.

Programming Languages 1 - 2022 Spring

Name: TANISHQ CHAUDHARY Neptun ID: C77067I Date: May 18, 2022

MAY 18, 2022

### **Declaration**

*By understanding this declaration, I admit that I have noticed that any kinds of devices or materials are NOT allowed and NOT acceptable on this exam, and I agree to that. I accept all the ethical rules of the University.*

*I admit that the voice and/or video might be recorded during my exam.*

*Without my signature this exam can NOT be graded and counted in to my final grade.*

*[Signature]*

### **Part I. - Test**

1. What will be the output of following c code?

```
void main()
{
    int i=2, j=2;
    while(i+1?--i:j++)
        printf("%d", i);
}
```

- a) 1      c)  
b) 2      d)

2. What will be the output of the following program?

```
void main()
{
    int i=1;
    i=2+2*i++;
    printf("%d", i);
```



3. What will be output of the following program?

```
void main()
{
    int a=0,b=10;
    if(a==0)
    {
        printf("true");
    }
    else
    {
        printf("false");
    }
}
```

- a) true
  - b) false

4. Choose the correct statement - based on the C language.

Select one:

- a. IF is a valid identifier.
  - b. An identifier may end with an underscore.
  - c. An identifier may start with an underscore.
  - d. All of the answers.

5. A **single-value variable** is a variable that is bound to a value only once.

- a. static variable
  - b. named constant
  - c. literal constant
  - d. implicit heap dynamic variable

6. What will be output of the following program?

```
void main()
{
    int a=2,b=7,c=10;
    c=a==b;
    printf("%d",c);
}
```

- a) 2  
b) 7  
c) 10  
d) 0

What will be the result of given code?

```
main()
{
    int i = 1;
    for(;;)
    {
        printf("%d",i++);
        if(i > 5)
            break;
    }
}
```

Select one:

- a. error because condition in for loop is must
- b. error because of two semicolon inside for loop.
- c. error because of break inside for loop.
- d. 1 2 3 4 5

Which of the following is not a valid identifier?

- a. \$examVar
- b. \_examVar
- c. 1examVar
- d. examVar

In C, parameters are always

- a. Passed by value result
- b. Passed by value
- c. Non-pointer variables are passed by value and pointers are passed by reference
- d. Passed by reference

A pointer variable can be \_\_\_\_\_.

Select one:

- a. Passed to a function as an argument.
- b. Returned by a function.
- c. Changed within function.
- d. All of the answers.

In the below statement, ptr1 and ptr2 are uninitialized pointers to int i.e. they are pointing to some random address that may or may not be valid address.

```
int* ptr1, ptr2;
```

- a. False
- b. True

Is the NULL pointer same as an uninitialised pointer?

Select one:

- True
- False

A variable which is visible only in the function in which it is defined, is called \_\_\_\_\_.

Select one:

- a. external variable
- b. local variable
- c. static variable
- d. auto variable

Output?

```
int main()
{
    {
        int var = 10;
    }
    {
        printf("%d", var);
    }
    return 0;
}
```

- a. 10
- b. Garbage Value
- c. Compiler Error

We use pointers in a C program because

- a. Pointers allow different functions to share and modify their local variables.
- b. To pass large structures so that complete copy of the structure can be avoided.
- c. Pointers enable complex "linked" data structures like linked lists and binary trees.
- d. All of the above

What will be the output of the following C code?

(Take care of scopeing!)

```
#include <stdio.h>
int main()
{
    int i;
    for ( i=0; i<5; i++ )
    {
        int i = 10;
        printf( "%d ", i );
        i++;
    }
    return 0;
}
```

- a. 0 1 2 3 4
- b. Compilation error
- c. 10 10 10 10 10
- d. 10 11 12 13 14

---

A C variable cannot start with

- a. a digit
- b. a special symbol other than underscore
- c. a dollar symbol
- d. all of the above choices.

C programs are converted into machine language with the help of

- a. a compiler.
- b. an Editor.
- c. an operating system.
- d. None of these.

\_\_\_\_\_ variable is initialized only once and remains into existence till the end of program?

Select one:

- a. static
- b. automatic
- c. register
- d. external

Determine output:

```
void main()
{
    int i=10;
    i = !i>14;
    printf("i=%d", i);
}
```

- a. 10
- b. 14
- c. 0
- d. None of these.
- e. 1

[Clear my choice](#)

# Programming Languages 1 - 2022

December 20, 2022.

Name: \_\_\_\_\_ Neptun ID: \_\_\_\_\_

COURSE CODE: INBPA0211E or INBMA0211E or INGK301 or INJK301 or INHK301

## Declaration

*By understanding this declaration, I admit that I have noticed that any kinds of devices or materials are*

*NOT allowed and NOT acceptable on this exam, and I agree to that. I accept all the ethical rules of the University.*

*I admit that the voice and/or video might be recorded during my exam.*

*Without my signature this exam can NOT be graded and counted in to my final grade.*

\_\_\_\_\_

## Part I. – Test

1. In the following program fragment, s2 will be executed if?

```
if(a > b)
    if(b > c)
        s1;
    else
        s2;
```

- a. a > b and b <= c
- b. a <= b
- c. b <= c and a <= b
- d. b > c

2. Direct descendants of C alongside with itself do not allow subprogram nesting.

- a. False, except C.
- b. True, except C.
- c. True for all.
- d. False for all.

3. In the following program fragment, a <= b will be printed if?

```
if(a > b)
    printf("a > b");
else
    printf("else part");
printf("a <= b");
```

- a. a < b
- b. a > b
- c. does not depend upon if condition
- d. a == b

4. In the below statement, ptr1 and ptr2 are uninitialized pointers to int i.e. they are pointing to some random address that may or may not be valid address.

```
int* ptr1, ptr2;
```

- a. False, because they are pointing to the null address.
- b. True, because they are in the same statement.
- c. True, because they are not initialized.
- d. False, because they are different type variables.

5. What will be output of the following program?

```
void main()
{
    int a=7,b=2,c=10;
    c=a=b;
    printf("%d",c);
}
```

- a) 2
- b) 10

- c) 7
- d) 0

Programming Languages 1 - 2022 Spring

Name: TANISHQ CHOURHARI Neptun ID: CT70GJ May 18, 2022

May 18, 2022

## Declaration

*By understanding this declaration, I admit that I have noticed that any kinds of devices or materials are NOT allowed and NOT acceptable on this exam, and I agree to that. I accept all the ethical rules of the University.*

*I admit that the voice and/or video might be recorded during my exam.*

*Without my signature this exam can NOT be graded and counted in to my final grade.*

*[Signature]*

## **Part I. - Test**

1. What will be the output of following c code?

```
void main() {
    int i=2, j=2;
    while(i>1?--i:j++)
        printf("%d", i);
}
```



- ? What will be the output of the following program?

```
void main()
{
    int i=1;
    i=2+2*i++;
    printf("%d", i);
```



3. What will be output of the following program?

```
void main()
{
    int a=0,b=10;
    if(a==0)
    {
        printf("true");
    }
    else
    {
        printf("false");
    }
}
```

- a) true
  - b) false

4. Choose the correct statement - based on the C language.

Select one:

- a. IF is a valid identifier.
  - b. An identifier may end with an underscore.
  - c. An identifier may start with an underscore.
  - d. All of the answers.

5. A **reference** is a variable that is bound to a value only once.

- a. static variable
  - b. named constant
  - c. literal constant
  - d. implicit heap dynamic variable

6. What will be output of the following program?

```
void main()
{
    int a=2,b=7,c=10;
    c=a==b;
    printf("%d",c);
}
```

Type in which a variable has different type in different time during execution.

```
dynode thru ex v C
    C Programming language : It is called -.
    Union data()
        int intValue;
        float floatValue;
        Char stringValue;
    } Int main()
```

3. The following function computes the maximum value contained in an integer array p[] of size n ( $n \geq 1$ )

```
int max(int *p, int n)
{
    int a=0, b=n-1;
    while (_____)
    {
        if (p[a] <= p[b])
        {
            a = a+1;
        }
        else
        {
            b = b-1;
        }
    }
    return p[a];
}
```

- a.  $a \leq n$
- b.  $b \leq 0$**
- c.  $b > (a + 1)$
- d.  $a \neq b$

4. If we don't initialize a static array, what will be the elements set to:

- a. Character constant
- b. A floating point number
- c. 0**
- d. An undetermined value

5. Consider the following C declaration

```
struct (
short s[5];
union {
float y;
long z;
}u;
)t;
```

Assume that the objects of the type short, float and long occupy 2 bytes, 4 bytes and 8 bytes, respectively. The memory requirement for variable t, ignoring alignment consideration, is

- a) 22 bytes**
- b) 18 bytes
- c) 14 bytes
- d) 10 bytes

## Part I. - Test

1. What will be the output of the program?

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

int main()
{
    char str[] = "ZERO\0\\SIX\0";
    printf("%s\n", str+6);
    return 0;
}
```

- a. SIX
- b. 6
- c. 12
- d. ZERO
- e. ZERO\0\SIX
- f. 0

2. Direct descendants of C alongside with itself do not allow subprogram nesting.

a. False, except C.

b. True, except C.

c. True for all.

d. False for all.

incorrect

```
Void main () {  
    Int i=2 , j = 2;  
    while(i + 1? --i : j++)  
        printf("%d", i);  
}  
a) 1      b) 2      c) 4      d) 6
```

```
void main () {  
    int i = 1;  
    i = 2+2*i++;  
    printf("%d",i);  
}  
a) 4      b) 5      c) 6      d) 7
```

```
Void main () {  
    Int a = 0 , b = 10;  
    If(a=0)  
    {  
        Printf("true");  
    }  
    else  
    {  
        Printf("false");  
    }  
}  
a) true      b) false
```

```
#include <stdio.h>  
void main () {  
    int a=100;  
    if(a>10)  
        printf("M.S Dhoni");  
    else if(a>20)  
        printf("M.E.X Hussey");  
    else if(a>30)  
        printf("A.B de villiers");  
}  
1 . M.S Dhoni      2 . A.B de villiers  
3 . M.S Dhoni      4 . compilation error.More  
M.E.X Hussey      than one conditions are true  
A.B de villiers      5 . None of the above
```

```
Void main () {  
    Int a=2 , b=7 , c=10;  
    C=a==b;  
    Printf("%d",c);  
}  
a . 2      b . 7      c . 10      d . 0
```

1. A C variable cannot start with

- a. a digit
- b. a special symbol other than underscore
- c. a dollar symbol
- d. all of the above choices.

C中标识符由字母、数字、下划线组成  
有符号不能是数字  
只能是字母。  
下划线

2. static variable is initialized only once and remains into existence till the end of program?

Select one:

- a. static
- b. external
- c. automatic
- d. register

3. In C, parameters are always

- a. Non-pointer variables are passed by value and pointers are passed by reference
- b. Passed by value
- c. Passed by value result
- d. Passed by reference

4. In the below statement, ptr1 and ptr2 are uninitialized pointers to int i.e. they are pointing to some random address that may or may not be valid address.

int\* ptr1, ptr2; *ptr2是int型变量*

- a. False
- b. True

5. Is the NULL pointer same as an uninitialised pointer?

Select one:

- a. True
- b. False

6. A variable which is visible only in the function in which it is defined, is called

local variable

Select one:

- a. auto variable
- b. local variable
- c. static variable
- d. external variable

7. Which of the following is not a valid identifier?

- a. \_examVar ?应该是多选.
- b. examVar
- c. \$examVar Java 标识符可以\$开头
- d. 1examVar 所有标识符不能以数字开头

8. We use pointers in a C program because

- a. Pointers allow different functions to share and modify their local variables.
- b. To pass large structures so that complete copy of the structure can be avoided.
- c. Pointers enable complex "linked" data structures like linked lists and binary trees.
- d. All of the above

9. C programs are converted into machine language with the help of

- a. a compiler.
- b. an Editor.
- c. an operating system.
- d. None of these.

compilers are used to translate each source code instruction into the appropriate machine language instruction

10. A pointer variable can be \_\_\_\_\_.

Select one:

- a. All of the answers.
- b. Passed to a function as an argument.
- c. Returned by a function.
- d. Changed within function.

## 11. Special words

in programming languages are used to make programs more readable by naming actions to be performed.

- a. Special words
- b. Statements
- c. Operators
- d. Labels

12.

Direct descendants of C alongside with itself do not allow subprogram nesting.

- a. False, except C.
- b. True for all.
- c. True, except C.
- d. False for all.

除 C 之外，子程序可嵌套

13.

The scopes created by blocks, which could be nested in larger blocks, are treated exactly like those created by subprograms.

- a. It is a true sentence.
- b. It is a false sentence.
- c. Cannot be compared the two thing.
- d. It depends on the language.

14.

Coroutines can have multiple entry points, which are controlled by the coroutines themselves.

- 協程
- a. Coroutines, multiple, subroutines
  - b. Coroutines, multiple, coroutines
  - c. Subroutines, single, subroutines
  - d. Subroutines, multiple, subroutines

15.

Consider the following example:

```
void func( int i, int j)
```

- a. It is a subprogram header.
- b. It is a subprogram protocol.
- c. It is a subprogram profile.
- d. It is a subprogram definition.

16.

The referencing environment of a statement in a dynamically scoped language is the locally declared variables, plus the variables of all other subprograms that are currently active.

- a. referencing environment, static scoped
- b. referencing environment, dynamically scoped
- c. referencing block, dynamically scoped
- d. referencing block, static scoped

17.

In the below statement, ptr1 and ptr2 are uninitialized pointers to int i.e. they are pointing to some random address that may or may not be valid address.

```
int* ptr1, ptr2;
```

- a. True, because they are in the same statement.
- b. False, because they are different type variables.
- c. True, because they are not initialized.
- d. False, because they are pointing to the null address.

18.

Consider the following C function

```
void swap (int a, int b)
{
    int temp;
    temp = a;
    a = b;
    b = temp;
}
```

In order to exchange the values of two variables x and y:

- a. swap(x,y) cannot be used as the parameters are passed by value
- b. swap(x,y) cannot be used as it does not return any value
- c. Call swap (x, y)
- d. Call swap (&x, &y)

19.

Pure interpretation is often requires more space than compilation.

- a. True, because there is the intermediate code generator.
- b. True, because it translates programs into an intermediate form.
- c. True, because e.g. the symbol table need to be available.
- d. False, there is no need for more space.

20.

The name for a memory location that may hold data is \_\_\_\_\_.

- a. address
- b. variable
- c. pointer
- d. storage

21.

Are there any difference between variable declaration and variable definition?

- a. There is no difference between them.
- b. A declaration occurs once, but a definition may occur many times.
- c. Both can occur multiple times, but a declaration must occur first.  
*- YR13: Lập trình C và C++*
- d. A definition occurs once, but a declaration may occur many times.
- e. Both can occur multiple times, but a definition must occur first.

23. In order to support recursion for subprograms, local variables need to be Stack dynamic

- a. stack dynamic
- b. static
- c. heap dynamic
- d. implicit heap dynamic

24. Subprogram declarations provide the subprogram's protocol but do not include their bodies.

- a. profile, do
- b. protocol, do not
- c. protocol, do
- d. profile, do not

25. A \_\_\_\_\_ is a variable that is bound to a value only once.

- a. implicit heap dynamic variable
- b. named constant
- c. static variable
- d. literal constant

26. Functional side effect \_\_\_\_\_ ?不希望

- a. need to be achieved as much as you can.
- b. need to be detected by the interpreter and stop interpretation.
- c. need to be avoided as much as you can.
- d. need to be detected by the compiler and stop compilation.

26. The pseudocode below demonstrates a loop.  
The code segment is an example of which type of loop?

```
num = 10
while (num < 20) {
    print num
    num -= 1
}
```

- a. Statement-controlled loop X无
- b. Infinite loop
- c. Counter-controlled loop 计数控制循环：for, if/else
- d. Event-controlled loop

计数控制循环比逻辑控制循环更复杂.  
任何一个计数控制循环都可以用逻辑控制循环表示.  
反之不一定.

27. The pseudocode below demonstrates a loop.  
The code segment is an example of which type of loop?

```
num=10
while(num<20){
    print num;
    num += 1
}
```

- a. Counter-controlled loop
- b. Logically controlled loop 逻辑控制循环：while
- c. Statement-controlled loop
- d. Infinite loop

28. Consider the following C-program:

```
double foo (double); /* Line 1 */

int main()
{
    double da, db;

    // input da
    db = foo(da);

}

double foo(double a)
{
    return a;
}
```

The above code compiled without any error or warning.

If Line 1 is deleted, the above code will show:

- a. some compiler-warnings due to type-mismatch eventually leading to unintended results
- b. no compile warning or error
- c. compiler errors
- d. some compiler-warnings not leading to unintended results

29.

What will be the output of the following C code?

(Take care of scopeing!)

```
#include <stdio.h>
int main()
{
    int i;
    for ( i=0; i<5; i++ )
    {
        int i = 10;
        printf( "%d ", i );
        i++;
    }
    return 0;
}
```

- a. Compilation error
- b. 10 10 10 10 10
- c. 0 1 2 3 4
- d. 10 11 12 13 14

30.

What will be the result of given code?

```
main()
{
    int i = 1;
    for(;;)
    {
        printf("%d",i++);
        if(i > 5)
            break;
    }
}
```

Select one:

- a. error because condition in for loop is must
- b. error because of two semicolon inside for loop.
- c. error because of break inside for loop.
- d. 1 2 3 4 5

31.

Determine output:

```
void main()
{
    int i=10;
    i = !i>14;
    printf("i=%d", i);
}
```

- a. None of these.
- b. 14
- c. 1
- d. 0
- e. 10

32.

Output?

12.

```
int main()
{
{
    int var = 10;
}
{
    printf("%d", var);
}
return 0;
}
```

- a. Compiler Error
- b. 10
- c. Garbage Value

33.

Output?

```
int var = 20;
int main()
{
    {
        int var = var;
    }
    printf("%d", var);
}
return 0;
}
```

- a. Compiler Error
- b. 20
- c. Garbage Value
- d. 10

Predict the output

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

Run on IDE



Garbage Value

Mcqs:

Special words in programming languages are used to make programs more readable by naming actions to be performed.

Many other languages, including all of the direct descendants of C, do not allow subprogram nesting.

The scopes created by blocks, which could be nested in larger blocks, are treated exactly like those created by subprograms.

Coroutines can have multiple entry points, which are controlled by the coroutines themselves.

The referencing environment of a statement in a static-scoped language is the variables declared in its local scope plus the collection of all variables of its ancestor scopes that are visible.

Swap(x,y) in C: cannot be used, because parameters are passed by value. (If pointers are used in the code & can be used to achieve result of pass-by-reference)

Pure interpretation is that it often requires more space. In addition to the source program, the symbol table must be present during interpretation.

The name for a memory location that may hold data is Variables .

In C parameters are passed by value, but pointers are passed by reference.

Declaration occurs once, definition may occur many times.

Logically controlled loop: while(...)

A compiler takes the program code (source code) and converts the source code to a machine language.

Identifiers may only begin with a letter, the underscore or a dollar sign.

A subprogram declaration provides the protocol, but not the body.

What is the output?

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

```
int var = 20;
```

```
int main()
```

```
{
```

```
    int var = var ;
```

```
    printf("%d",var);
```

```
}
```

```
    return 0;
```

```
}
```

Answer: Garbage value

Static variable is initialized only once and remains into existence till the end of the program.

C programs are converted into machine language with the help of a

Answer:Compiler

A c variable cannot start with

A Number

**B. A special symbol other than underscore**

A pointer variable can be : Changed within a function

Is the NULL pointer same as an uninitialized pointer? False

a variable which is visible only in the function in which it is defined is called LOCAL VARIABLE.

w

What will be the result of the given code? ( for; ;)

What will be the result of given code?

```
main()
{
    int i = 1;
    for(;;)
    {
        printf("%d",i++);
        if(i > 5)
            break;
    }
}
```

Answer: 1 2 3 4 5

A pointer variable can be :

ANSWER: All of the answers ( Passed to a function as an argument, Changed within a function, Returned by a function)

We use c pointer in a program because : All of the above

Functional side effect NEED TO BE AVOIDED AS MUCH AS YOU CAN.

In order to support recursion for subprograms, local variables need to be STACK DYNAMIC.

Determine output (void)

```
void main()
{
    int i=10;
    i=!i>14;
    printf("i=%d", i);
}
```

- A. 10
- B. 14
- C. 0
- D. 1

**Answer:** Option C

#### Solution(By Examveda Team)

In the expression `!i>14`, NOT (!) operator has more precedence than ">" symbol. ! is a unary logical operator. !i (!10) is 0 (not of true is false). 0>14 is false (zero).

Consider the following C programme

```
double foo (double); /* Line 1 */

int main()
{
    double da, db;
    // input da
    db = foo(da);
}
```

```
double foo(double a)
{
    return a;
}
```

The above code compiled without any error or warning. If Line 1 is deleted, the above code will show:

Answer: Compile errors.

What will be the output of the following C code?

```
int main()
{
    int i ;
    for(i=0; i<5; i++)
    {
        int i = 10;
        printf("%d",i);
        i++;
    }
    return 0;
}
```

Answer: 10 10 10 10 10

In the below statement, ptr1 and ptr2 are uninitialized pointers to int i.e. they are pointing to some random address that may or may not be valid address.

```
int* ptr1, ptr2;
```

Answer: False

Output?

```
int main()
{
{
    int var = 10;

}
{
    printf("%d", var);

}
return 0;
}
```

ANSWER: Compiler error

1:Define Static and Dynamic Binding with Examples.

Static binding is done at compile time when a function is called in order to match it with the definition.

Dynamic binding is at run time where we can specify that the compiler matches a function call with the correct function definition at run time.

2:Define subprogram protocol.

Subprogram protocol is its parameter profile plus, if it is a function, its return type.

3:How can you describe a variable? What is a variable? List and explain all the attributes of a variable (excl. lifetime and scope).

A program variable is an abstraction of a computer memory cell or collection of cells. Programmers often think of variables as names for

memory locations, but there is much more to a variable than just a name.

- ❑ Name - not all variables have them
- ❑ Address - the memory address with which it is associated.
- ❑ Type - determines the range of values of variables and the set of operations that are defined for values of that type.
- ❑ Value - the contents of the location with which the variable is associated

4. What is the referencing environment of a statement?

Set of all names visible to the statement.

5. What does it mean by implicit heap-dynamic variables? (Explain its advantages and disadvantages)

Implicit heap-dynamic variables: Allocation and deallocation caused by assignment statements.

❑ Advantages:

❑ Disadvantage:

6. Define scope and lifetime. Explain C's scoping and lifetime in the following code fragment: Refer to Photo

❑ Scope: The range of statements in which the variable is visible. A variable is visible in a statement if it can be referenced in that statement.

❑ Lifetime: A time during which the variable is bound to a specific memory location. The lifetime begins when it is bound to a specific cell and ends when it is unbound from that cell.

The reference to count in the while loop is to that loop's local count.

In this case, the count of sub is hidden from the code inside the while loop. In general, a declaration for a variable effectively hides any declaration of a variable with the same name in a larger enclosing scope.

The scope of the variable sum is completely contained within the printhead function. It does not extend to the body of the function sub, although sub executes in the midst of the execution of printhead. However, the lifetime of printhead extends over the time during which sub executes.

7. What are the design issues for arrays?

What types are legal for subscripts?

- Are subscripting expressions in element references range checked?
- When are subscripts ranges bound?
- When does array allocation take place?
- Are ragged or rectangular multidimensional arrays allowed, or both?
- Can arrays be initialized when they have their storage allocated?
- What kinds of slices are allowed, if any?

8. What is unusual about C's multiple-selection statement?

¶ The C switch statement has virtually no restrictions on the placement of the case expressions, which are treated as if they were normal statement labels.

¶ This laxness can result in highly complex structure within the switch body.

You can only put expressions that result in integers instead of random boolean expressions, and these integers are used to calculate jumps from the top of the switch to the part that matches that value.

9. What are the two common problems with pointers? Explain your answers.

Dangling pointers (dangerous) – A pointer points to a heap-dynamic variable that has been deallocated

Lost heap-dynamic variable – an allocated heap-dynamic variables that is no longer accessible to the user program (often call garbage).

10. What are the three semantics models of parameter passing? Explain in detail what is pass-by-value.

in mode, out mode, and in-out mode.

Pass-by-value: the value of the actual parameter is used to initialize the corresponding formal parameter. This formal parameter is then used as a local variable in the subprogram. Since the subprogram is receiving data from the actual parameter, this is a model of in-mode semantics. In most cases, pass-by-value is implemented using copy where an actual value is copied then transmitted. However, it can be implemented by passing an access path to the value of the actual parameter. Copy implementation is more efficient, because when using an access path, the value is stored in a write protected cell that is not always simply enforced. An advantage of pass-by-value is its speed. Since the actual value is passed there is no need to go find it. A disadvantage is that if a copy is used then that copy must be stored, and that storage could be costly if using a large variable.

variable – name, address, value, type, lifetime, scope (attributes)  
preprocessor directive begins with #

#### 1) Block

Structure in which we can combine a bunch of statements. Statements in it are treated as if they were one statement.

Allows to localise parts of the program by introducing new data environments, or scopes.

#### 2) Scoping and lifetime

Scope of variable avg is completely contained in printavg() function. Doesn't extend to body of function sub(). Sub() executes in the midst of execution of printavg(). Lifetime of printavg() extends to time during which sub() executes.

#### 3) In what way are reserved words better than keywords?

Reserved words can't be used as names. They're better because ability to redefine keywords can be confusing.

Keywords have special meaning in language and are part of syntax.

Eg. Fortran has no reserved words, all keywords (if, then, etc) can be used as identifiers.

#### 4)

value: 1 list[0]: 2

list[0]: 3 list[1]: 2

value:2 listvalue: 1

In pass by reference, we pass the address/memory location of the variable. Any change that happens inside the function affects its value outside of it as well.

In pass by value, we pass the value of variable to function. Changes that happen to it remain within the scope of that function and don't affect the value of it outside.

## 5) Primary Task of Lexical Analyser

To read input characters in the code and produce tokens.

Syntax analyser takes lexical units from lexical analyser to construct hierarchical structures called parse trees (which represent syntactic structure of the program).

**Lexeme**- Lowest level syntactic units. Eg. Numeric literals, operators, special words.

**Token**- Category of language's lexemes. Eg. Lexeme / is token Div-OP

**Class**- Modelling tool to represent real world objects in a program.

**Instance**- Variation of an object created from a class.

## 6) Pass by Value and Pass by Reference

In pass by reference, we pass the address/memory location of the variable. Any change that happens inside the function affects its value outside of it as well.

In pass by value, we pass the value of variable to function. Changes that happen to it remain within the scope of that function and don't affect the value of it outside.

## 7) Operator precedence- Decides order of evaluation of operators in an expression

Associativity- Decides order in which operators of same precedence are evaluated in an expression.

For instance, it could be left to right.

## 8) Static ancestor of a program- Programs that completely contain the subprogram in question.

All the procedures in the program in which the procedure is defined. (definition of procedure is nested)

Dynamic ancestors of program- Programs that are called to reach the subprogram in question.

All procedures called before sub() during the execution of a program that haven't finished executing.

The procedures wait for sub() to finish before they can terminate.

Dynamic Scoping- This means that each identifier has a global stack of bindings, and the most recent binding is searched for occurrences of the identifier.

To put it another way, in dynamic scoping, the compiler searches the current block first, then all of the calling functions.

```
int x = 10;

// Called by g()
int f()
{
    return x;
}

// g() has its own variable
// named as x and calls f()
int g()
{
    int x = 20;
    return f();
}

main()
{
    printf(g());
}
```

Output: 20

**9) Difference between function and procedure.**

A function is compiled before being called. A procedure is compiled once and can be called over and over without being compiled again.

Functions return values and procedures do not.

**10) Coroutine**

Programs that allow execution to be suspended and resumed. Even though they might stop execution, they let us return to point from where we left off. It is a line of execution, with its own stack, its own local variables, and its own instruction pointer; but it shares global variables and mostly anything else with other coroutines.

Functions pass control back and forth. Function within a function usually has full control as to when parent function can continue running. In coroutine, that control is passed back and forth.

**11) Reference environment of a statement**

Set of all variables that are visible in the statement and can be referenced in it.

**12) How is C's for statement different?**

Each of the expressions can contain multiple expressions.

Loop of c is made of 3 parts which each define a task. (Index; condition; change to index after condition)

**MULTIPLE CHOICE QUESTIONS.**

1. The referencing environment of a statement in a dynamically scoped language is the locally declared variables.
2. Which term describes the mechanism of a function calling itself?  
Answer: Recursion.
3. When a function returns a value, the entire function call can be assigned to a variable. True or false?  
Answer: True
4. Which of the following converts source code into machine code at each runtime?  
Answer: Compiler.
5. Pure interpretation often requires more space than compilation:  
Answer: True, because e.g. the symbol table needs to be available.
6. A \_ variable is a variable that is bound to a value only once  
Answer: Named constant
7. Consider the following C function:  
Answer: swap(x,y) cannot be used as the parameters are passed by

```
void swap (int a, int b)
{
    int temp;
    temp = a;
    a = b;
    b = temp;
}
```

8. In the below statement, ptr1 and ptr2 are uninitialized pointers to int i.e. they are pointing to some random address that may or may not be valid address.

Answer: False, because they are different type variables.

9. when argument passing mechanism is "pass by value", the function works with the original arguments in the caller program unit. Is this True ?

Answer: False

10. Are there any difference between variable declaration and variable definition?

Answer: A declaration occurs once, but a definition may occur many times

11. In C, parameters are always

Answer: Passed by value

12. Which of the following commonly happens to variables (in most languages)? [3 answers]

Answers: declaration, assignment, initialization

13. In programming, a series of logically ordered steps that lead to a required result is called

Answer: an Algorithm

14. Assuming that = and / are the assignment and division operators, what will be the outcome of the following code in most programming languages:

x = 3

y = 7

z = x / (y-7)

Answer: Runtime error

15. Dynamic Ancestors of a subprogram-

All procedures called before sub() during the execution of a program that have not yet completed execution are dynamic ancestors of a subprogram sub(). These are the procedures that are awaiting the completion of procedure sub() before terminating. Simply stated, dynamic ancestors are those who are summoned to reach the desired subprogram

16. Consider the following function

Void func (int l, int j)

Answer: It is a subprogram header

17. how does operand evaluation order interact with functional side effects?

Answer: If the language does not allow functional side effects then the order of evaluating the operands has no effects on the value of the expression.

18. Define operator precedence and operator associativity?

Precedence: defines order and priority of the operator evaluation from different precedence levels.

Associativity: defines the order of operators evaluation when it is from the same precedence level.

19. Define functional side effects

. : when a function changes a two-way parameter or a non-local variable.

Example a:

a + abc(a)

If abc has no side effect of changing a, then order of evaluation has no effect on value of expression. But if abc has side effect, and for example a=5, but abc changes to 10, then if right associativity, the abc will be evaluated first and output will be 10 + abc(10)

If left associativity, abc will be evaluated second and output will be 5 +abc(5)

Example: a=20

Lab(a) takes a and returns 30 and changes a to 40

Why we avoid them?

- Side effects make it harder to reason about program behavior:
  - harder to debug
  - harder to prove that the program is correct

Define operand evaluation order:

The operands of any C operator are evaluated in an undetermined order (except where noted below). This includes the order in which function arguments in a function-call expression are evaluated, as well as the order in which subexpressions within any expression are evaluated. The compiler will evaluate them in any order, and when the same expression is evaluated again, it may choose a different order.

20. What is coercion?

Is an implicit type conversion that is initiated by the compiler

21. What is the purpose of a compound assignment operator?

A compound assignment operator is a short hand method of specifying a commonly needed form of assignment.

22. What is the associativity of C's unary arithmetic operators?

Associativity is to the right.

23. What are the primary tasks of a lexical analyzer?

Comments and white space outside lexemes are skipped during the lexical analysis process because they have no bearing on the program's meaning. In addition, the lexical analyzer adds lexemes for user-defined names to the symbol table, which is used by the compiler later on. Finally, lexical analysis detects and reports syntactic errors in tokens, such as incorrectly formed floating-point literals.

24. What is one possible disadvantage of treating the assignment operator as if it were an arithmetic operator?

It provides yet another side effect.

25. How does C support relational and Boolean expressions?

Answer: By using numeric types to express the Boolean value 0 for false and 1 for true.

C supports relational and Boolean expression by using the signs of >, >=, <, <= and == to express the relational and uses the variable type of Boolean to assign true or false or can be represented as 1 for true and 0 for false.

26. What is a generic subprogram?

A subprogram that takes generic parameters that are used in type expressions that describe the types of the parameters of the subprogram.

```
template <class Type>
void show(Type a) {
    cout<<a;
}
void main()
{
    show(10);
    show(10.2);
    show('c');
}
```

27. Explain for pass by value and pass by reference

```
28. int i=10 ;
29. program main ()
30. {
31.     int j = 20;
32.     i = 50;
33.     call f (i, j);
34.     print i, j;
35. }
36. procedure f (x, y)
37. {
38.     i = 20;
39.     x = 10;
40.     y = y + i ;
41. }
```

In pass by value, the values gained inside the function will not affect the values in the main function.

In pass by reference, the values gained inside the function will affect the actual perimeters in the main function.

28. Reserved words- In C, we have 32 keywords, which have their predefined meaning and cannot be used as a variable name. These words are also known as “reserved words”.

Cannot be used as an Identifier.

Example; **return** – This keyword is used for returning a value.

**break** – Used with any loop OR switch case.

**for, while, do** – types of loop structures in C.

**void** – One of the return type.

# HLPL1 THEORY MCQ'S

Q 1: \_\_\_\_\_ can have \_\_\_\_\_ entry points, which are controlled by the \_\_\_\_\_ themselves.....

Answer: coroutines, multiple, coroutines.

Q 2: The scopes created by blocks, which could be nested in larger blocks, are treated exactly like those created by subprograms.

Answer: This is true sentence.

Q 3: Direct descendants of C, do not allow subprogram nesting.

Answer: True for all.

Q 4: \_\_\_\_\_ in programming languages are used to make programs more readable by naming actions to be performed.

Answer: Special worlds.

Q 5: The \_\_\_\_\_ of a statement in a \_\_\_\_\_ language is the locally declared variables, plus the variables of all other subprograms that are currently active.

Answer: referencing environment , dynamically scoped.

Q 6: A \_\_\_\_\_ is a variable that is bound to a value only once.

Answer: Named constant.

Q 7: Pure interpretation is often requires more space then compilation.

Answer: True because e.g. the symbol table needs to be available.

Q 8: Consider the following example  
Void func(int I , int J)

Answer: subprogram header.

Q 9: Subprogram declarations provide the subprogram's \_\_\_\_\_ but \_\_\_\_\_ include their bodies.

Answer: Protocol , do not.

Q 10: In the below statement ptr1 and ptr2 are uninitialized pointers to int i.e. they are pointing To some random address that may or may not be valid address.  
Int\* ptr1 , ptr2;

Answer: False , because they are different type variables.

Q 11: Functional side effect \_\_\_\_\_.

Answer: Need to be avoided as much as you can.

Q 12: the name for a memory location that may hold data is.

Answer: Variable.

Q 13: Are there any difference between variable declaration and variable definition?

Answer: A declaration occurs once, but a definition may occur many time.

Q 14: In order support recursion for subprogram, local variables need to be\_\_\_\_\_.

Answer: Stack dynamic.

Q 15: Consider the following function .

```
Void swap( int a , int b )
{
    Int temp;
    Temp = a;
    a = b;
    b = temp;
}
```

In order to exchange the two values of x and y.

Answer: swap(x,y) can not be used as the parameter are passed by values.

Q 16: An \_\_\_\_\_ is one that has the same name as another subprogram in the same referencing environment.

Answer: overloaded subprogram.

Q 17: Which type of loop is this.

```
Num = 10;
While(num<20){
    Print num;
    Num -=1;
}
```

Answer: Infinite loop.

Q 18: Determine the OUTPUT.

```
Void msin()
{
    Int i=10;
    I = !i<14;
    Printf("%d", &i);
}
```

Answer: 1.

Q 19; Which type of loop is this.

```
Num = 10;
While(num<20){
    Print num;
    Num +=1;
}
```

}

Answer : Logically controlled loop.

Q 20; Determine the OUTPUT.

```
int var =20;
int main()
{
{
    int var = var;
}
{
    printf("%d",var);
}
return 0;
}
```

Answer: 20.

Q 21: In C parameter are always

Answer: Passed by values.

# Essay

Two types of subprograms:

1. Procedures: A sequence of instructions to do a task. They may take input but usually don't give output.
2. Functions: A function is a set of instructions that returns a value.

Q. What problems can arise from nested selection statements in C?

A.

1. Poor readability blah blah blah
2. Poor performance from unoptimized code blah blah blah

Emphasis on the GPT/google part, this exam has a lot of general knowledge parts so you shouldn't leave any doubts in your mind

There will be bullet point questions such as "What is the difference between switch statement in C and the one in Java?" where chatGPT is way better than google

There's also questions that ask you to do things like declare a variable on the heap using code in C++ or a specific language, gpt also works for that

# Programming Languages 1

## Part II. - Questions

- 1) Which produces faster program execution: a compiler or a pure interpreter? (1 point)  
[Explain your answer!] (3 points)
- 2) Define the meaning of syntax and semantics! (2 points)
- 3) How can you describe a variable? What is a variable? (1 point)  
List and explain all the attributes of a variable (excl. lifetime and scope)! (4 points)
- 4) What is the definition of block? (2 points)
- 5) What does it mean when a variable is static? (4 points)  
(Explain its advantages and disadvantages!)
- 6) Define scope and lifetime! (2 points)  
Explain C's scoping and lifetime in the following code fragment: (3 points)

```
void sub() {
    int count;
    ...
    while( ... ) {
        int count;
        count ++
        ...
    }
    ...
}
void printheader() {
    int sum;
    ...
    sub();
}
```

- 7) What are the design issues for multiple-selection statements? (2 points)  
[Explain your answers for C especially! ] (3 points)

## Evaluation

Part I. - Test : at least four (4) good answers required for grading

## Part II. - Questions:

0 – 17	failed
18 – 21	passed
22 – 26	average
27 – 30	good
31 – 36	excellent

# Programming Languages 1

## Part II. - Questions

- 1) Define the concept of type binding for static and dynamic cases!  
[Explain your answer with examples for each one! (expl./impl.; type inf.; compil.,intrpt)] (4 points)
- 2) Define steps to the semantics of a call to a "simple" subprogram!  
Show an example! (2 points)
- 3) How can you describe a variable? What is a variable?  
List and explain all the attributes of a variable (incl. lifetime and scope)! (1 point)  
(6 points)
- 4) What is a record as a data type?  
Compare it with arrays! (1 points)  
(2 points)
- 5) What does it mean when a variable is stack-dynamic?  
(Explain its advantages and disadvantages!) (3 points)
- 6) Define the two fundamental kinds of subprograms!  
What are the similarities and differences between them? (2 points)  
(3 points)
- 7) What is an associative array?  
[ Explain your answers for a chosen language!] (2 points)  
(2 points)
- 8) What could be a problem with nested selection statement in C based languages?  
(2 points)
- 9) What are special words?  
Explain your answer with its subtypes and examples! (1 points)  
(3 points)
- 10) Explain the Pass-by-result model in details with examples!  
Discuss the well-known potential problem with it! (4 points)

## Evaluation

Part I. - Test : at least four (4) good answers required for grading the exam!

## Part II. - Questions:

0 - 19	failed
20 - 24	passed
25 - 29	average
30 - 33	good
34 - 38	excellent

- 4) Define static scope!  
Show an example for nested scope in the C language! (2 points)
- 5) What does it mean when a variable is stack-dynamic?  
(Explain its advantages and disadvantages!) (3 points)
- 6) Define the two fundamental kinds of subprograms!  
What are the similarities and differences between them? (2 points)  
(3 points)
- 7) What is an associative array?  
(Explain your answers for a chosen language!) (2 points)  
(2 points)
- 8) What could be a problem with nested selection statement in C based languages?  
(2 points)
- 9) What are special words?  
Explain your answer with its subtypes and examples! (1 points)  
(3 points)
- 10) Explain the Pass-by-reference model in detail, with examples!  
Discuss the well-known potential problem with it! (4 points)

### Evaluation

Part I. - Test : at least four (4) good answers required for grading the exam!

### Part II. - Questions:

0 - 19	failed
20 - 24	passed
25 - 29	average
30 - 33	good
34 - 38	excellent



```
    }  
    ...  
}  
void printheader () {  
    int sum;  
    ...  
    sub ();  
}
```

- 7) What are the design issues for multiple-selection statements? (2 points)  
[ Explain your answers for C especially! ] (3 points)
- 8) How does C support relational and Boolean expressions? (2 points)
- 9) What are the two common problems with pointers? (2 points)  
Explain your answer with details! (2 points)
- 10) In what way is C's for statement more flexible than that of many other languages?  
[ Show examples! ] (3 points)

## **1. Describe lexical and syntax analysis with a few sentences!**

4.2&4.3

**Lexical analysis:** It is essentially a pattern matcher and serves as the front end of a syntax analyzer.

It collects lexemes (collects characters into logical groupings) from a given input string and produce the corresponding tokens (internal codes' categories).

Syntax analysis: the process of analyzing syntax.

It serves two purposes:

1. Make sure the syntax of the input program is syntactically correct.
2. If the grammar input is correct, it will generate a complete parse tree.

## **2. What are the primary tasks of a lexical analyzer? (3 points)**

4.2 Lexical Analysis

(Describe it with your own words, what steps are performed and what produced by it)

**Primary tasks:** It helps to convert a sequence of characters into a sequence of tokens.

- Steps:**
1. It collects characters into logical groupings and assigns internal codes to the groupings according to their structure.
  2. It extract lexemes from a given input string and produce the corresponding tokens.
  3. It skipping comments and white space outside lexemes.

## **3. What is a reserved word?**

5.2.3 Special Words

A reserved word is a special word of programming language which cannot be used as a name. In most languages, special words are classified as result words

## **4. What is a keyword?**

5.2.3 Special Words

A keyword is a word that is special only in certain contexts

## **5. What is the l-value of a variable? What is the r-value? (2 marks)**

5.3.2 Address

**l-value:** the address of this variable

**r-value:** the value of this variable

## **6. Describe the ways that aliases can occur with pass-by-reference parameters**

5.3.2 Address

Aliases are two or more variables bound to the same memory location.

An example: int a=0; int &a1=a; In this case, a and a1 are aliases.

## **7. Write a simple assignment statement with one arithmetic operator in some**

language you know. For each component of the statement, list the various bindings that are required to determine the semantics when the statement is executed. For each binding, indicate the binding time used for the language. (4 marks)

### **5.4 The Concept of Binding**

(C language) **int i=0; i=i+1;**

**The type for i:** bounded at compile time;

**Set of possible values of i:** bounded at compiler design time;

**The meaning of the operator +:** bounded at compile time;

**The internal representation of literal 1** is bound at compiler design time;

**The value of i:** bounded at execution time;

## 8. Explain why dynamic type bindings is closely related to implicit heap-dynamic variables, using the following code example: 为什么动态绑定与隐式堆动态变量密切相关

```
var a;  
...  
a = "Hello";  
...  
a = 123.456;
```

The variable a is bounded at runtime, in this case, firstly a is "Hello" and then it changes to "123.456".

And implicit heap-dynamic variables are bounded to heap storage only when they are assigned values, which is matched with dynamic type bindings.

## 9. Describe the case when a variable does not have a name 5.4.3.3 Explicit Heap- Dynamic Variables

**Explicit heap-dynamic variables** are nameless (abstract) memory cells. They often do not have identifiers associated with them. They can be referenced only by pointer or reference type variables.

### 10. Consider the following C program, which variable has the longest scope?

Explain your answer with details! [3 marks]

```
int a;  
int main()  
{  
    int b;  
    // ..  
    // ..  
}  
int c;
```

### 5.5 Scope

a has the longest scope.

a is accessible in all time in this process, b is limited in main (), c is accessible only after declaration in the last line.

### 11. 5.5 Scope

Explain C's scoping and lifetime for all avg variables in the code fragment:

```
void sub(){  
    int avg;  
    ...  
    for( ; ;){  
        int avg;  
        avg++  
        ...  
    }  
    ...  
}  
void printavg(){  
    int avg;  
    ...  
    sub();  
}
```

the first avg and third avg:

scope: all inside the function sub() / printavg()

lifetime: available util the function that calls it is done

the second avg:

scope: limited in the for loop

lifetime: available util the for loop is done

## 12. What is a static ancestor of a subprogram? Show an example (2 marks) 5.5.1 Static Scope

The static ancestor of a subprogram is its static parent and the static parent derived from it, until the largest one contains all the subprogram.

Example:

```
function() { //it is static ancestor of sub2
    function_sub1() {
        int x=1;
        function_sub2();
    }
    function_sub2() {
        int y=x;
    }
}
```

## 13. What is a block? 5.5.2 Blocks

It is the scope of local variables in a piece of code (this section of code between each pair of braces)

## 14. What is dynamic scoping and a dynamic ancestor of a subprogram? 5.5.6 Dynamic Scope

Show an example! (2 points)

**Scope** refers to the places in a program where a variable is visible and can be referenced.

Under **dynamic scoping**, a variable is bound to the most recent value assigned to that variable.

**Dynamic ancestors** of a subprogram are all the procedures called before it during the execution of a program.

Example:

```
func 1 () // which is a dynamic ancestor of func_sub1( )
{
    func_sub1 (){
        int i=1 ;} //it is a dynamic scoping
}
```

## 15. Define scope and lifetime. 5.6 Scope and Lifetime

**Scope:** The range of statements where the variable is visible.

**Lifetime:** The time when the variable is bound to a specific memory location.

## 16. What is the referencing environment of a statement? 5.7 Referencing Environments

The reference environment of a statement refers to the collection of all visible variables in this statement

## 17. 强制转换 Explain how the coercion rules of a language affect its error detection. (4 marks) 6.13 Strong Typing

It reduces the benefits of type checking.

Example: here are some codes:

```
int a, double b,c,d;
```

if the input should be: d=b\*c; but we make a mistake: d=b\*a; the compiler would not detect this as an error instead of coercing int a to double a.

## 18. What is type equivalence? 6.14 Type Equivalence

Two types are equivalent if an operand of one type in an expression can be substituted for one of the other types without coercion. It is a strict form of type compatibility.

## **19. How do parentheses affect the precedence rule? What will be the evaluation order?** 7.2.1.3 Parentheses

we can alter the precedence and associativity rules by using parentheses in expressions.

A parenthesized part of an expression has precedence over its adjacent unparenthesized parts.

## **20. Define functional side effect, important to show examples as well.** 7.2.2.1 Side Effects

If one function changes either its parameter or a global variable, it **has side effects**.

Here is an expression:  $a + \text{fun}(a)$ , if  $\text{fun}(a)$  has side effects, when function changes  $a$ , the order of evaluation of these two operands may be changed.

## **21. Explain why operator overloading increases the semantic capability of a language?** 7.3 Overloaded Operators

This multiple use of an operator is called operator overloading.

A single operator symbol has more than one meaning

## **22. How does short-circuit? 短路求值** 7.6 Short-Circuit Evaluation

**Definition:** A short-circuit evaluation of an expression is one in which the result is determined without evaluating all of the operands and/or operators.

For Boolean operators, the second argument is evaluated only if the first argument does not determine the value of the expression. Example:  $a \text{ AND } b$ :  $b$  is only evaluated if  $a$  is false;  $a \text{ OR } b$ :  $b$  is only evaluated if  $a$  is true.

## **23. In what way C's switch statement more different than that of many other languages? (2 marks)**

**show examples (2 marks)** 8.2.2.2 Examples of Multiple Selectors

The switch structure in C language hardly restricts the position of case expressions, which may be a highly complex structure in the switch structure.

**Example:**

```

switch (number)
default:
if (number)
    case 1: case 3: case 5:
        printf ("The number is odd"\n);
    case 2: case 4: case 6:
        printf ("The number is even"\n);

```

## **24. In what way is C's for statement more different than that of many other languages?**

**(Show examples!)** 8.3.1.2 The for Statement of the C- Based Languages

The general form of C's for statement:

```
for(initialization_exp ; loop_condition ; increment_exp)
```

    loop body; // it can be a single statement / a compound statement / a null statement;

Some features:

1. The three expressions in the C's for statement are all optional.
2. C's for statement does not require counting.
3. The last two loop parameters of the C's for statement are calculated every time the loop is repeated, and the variables in these parameter expressions can be modified in the loop body.

## 25. 9.5.2 Implementation Models of Parameter Passing

Consider the following program written in C syntax (only syntax - it is not a C code):

```
void swap(int a, int b) {
    int temp;
    temp = a;
    a = b;
    b = temp;
}

void main() {
    int value = 2, list[5] = {1, 3, 5, 7, 9};
    swap(value, list[0]);
    swap(list[0], list[1]);
    swap(value, list[value]);
}
```

For each of the following parameter-passing methods, what are all of the values of the variables value and list after each of the three calls to swap?

- a. Passed by value
- b. Passed by reference

- a. 2 1 3 5 7 9  
2 1 3 5 7 9  
2 1 3 5 7 9
- b. 1 2 3 4 7 9  
1 3 2 5 7 9  
2 3 1 5 7 9

26. Consider the program below in a hypothetical language which allows global variable and a choice of call by reference or call by value methods of parameter passing.

What will be the output of the program for the two parameter passing mechanisms?

Explain it for pass-by-value and pass-by-reference as well! [4 marks]

```
int i ;
program main ()
{
    int j = 60;
    i = 50;
    call f (i, j);
    print i, j;
}
procedure f (x, y)
{
    i = 100;
    x = 10;
    y = y + i ;
}
```

## 9.5.2 Implementation Models of Parameter Passing

**pass-by-value:** i=100, j=60 (执行了 i=100 的定义和函数操作, 但操作完的值不传回)

i is global declared, j is a local variable in main. At first i=50, j=60. When f called value, i and j are passed to it. And i=100 change the global i to 100, x=10 changes the local x from 50 to 10, and y=60+100=160. When returning back to main, i=100 and j=60;

**pass-by-reference:** i=10, j=70

When f called value, x and y point to the memory location of i and j respectively. So x=10 means x, that is, global i=10, y=y+i=60+10=70. When returning back to main, i will be 100 and j will be 60. When returning back to main, i=10 and j=70;

**27. What is the Pass-by-reference model of parameter passing? (describe it)** [9.5.2.4 Pass- by- Reference](#)

**Explain in your answer the parameter collision problem! (4 points )** [9.5.2.2 Pass- by- Result](#)

**Pass-by-reference** refers to passing the address of the actual parameter directly to the function when calling the function, and the modification of the parameter in the function will also affect the actual parameter.

**Parameter collision** occurs for the order can be implementation of expressions matter, so different implementations may produce different results.

**28. What is an overloaded subprogram?** [9.9 Overloaded Subprograms](#)

It is a subprogram which has the same name as other subprograms, but different in number, type order of parameters or return value in the same reference environment.

**29. What is a generic subprogram?**

**Explain it detailly (2 marks) and show an example (2 marks).** [9.10 Generic Subprograms](#)

Parameter polymorphism means that no specific types are specified during declarations and definitions, but these types are used as parameters, so that the definition is applicable to various specific types.

参数多态是指在声明和定义时不指定具体的类型，而是将这些类型作为参数使用，从而使定义适用于各种具体的类型

And a generic subprogram is a subprogram which have parametric polymorphism.

泛型子程序是具有参数多态性的子程序

**Example:** Generic functions in C++ is template function

```
template <typename T>
```

```
class List {
```

```
    // Class contents.
```

```
};
```

```
List<Number> list_of_numbers;
```

```
List<Car> list_of_cars;
```

**30. What is a coroutine?** [9.13 Coroutines](#)

it is a kind of computer program components that generalize subroutines for non-preemptive multitasking, by allowing execution to be suspended and resumed

它是一种计算机程序组件，通过允许暂停和恢复执行来概括非抢占式多任务处理的子程序

1.

What is the l-value of a variable?

What is the r-value?

(2 marks)

The address of a variable is sometimes called its **l-value**, because the address is what is required when the name of a variable appears in the left side of an assignment.

A variable's value is sometimes called its **r-value** because it is what is required when the name of the variable appears in the right side of an assignment statement. To access the *r*-value, the *l*-value must be determined first. Such determinations are not always simple. For example, scoping rules can greatly complicate matters, as is discussed in Section 5.5.

2.

What are the primary tasks of a lexical analyzer? (3 points)

[describe it with your own words, what steps are performed and what produced by it]

A lexical analyzer is essentially a pattern matcher. A pattern matcher attempts to find a substring of a given string of characters that matches a given character pattern. Pattern matching is a traditional part of computing. One of the earliest uses of pattern matching was with text editors, such as the ed line editor, which was introduced in an early version of UNIX. Since then, pattern matching has found its way into some programming languages—for example, Perl and JavaScript. It is also available through the standard class libraries of Java, C++, and C#.

A lexical analyzer serves as the front end of a syntax analyzer. Technically, lexical analysis is a part of syntax analysis. A lexical analyzer performs syntax analysis at the lowest level of program structure. An input program appears to a compiler as a single string of characters. The lexical analyzer collects characters into logical groupings and assigns internal codes to the groupings according to their structure. In Chapter 3, these logical groupings are named lexemes, and the internal codes for categories of these groupings are named tokens. Lexemes are recognized by matching the input character string against character string patterns. Although tokens are usually represented as integer values, for the sake of readability of lexical and syntax analyzers, they are often referenced through named constants.

Lexical analyzers extract lexemes from a given input string and produce the corresponding tokens. In the early days of compilers, lexical analyzers often processed an entire source program file and produced a file of tokens and lexemes. Now, however, most lexical analyzers are subprograms that locate the next lexeme in the input, determine its associated token code, and return them to the caller, which is the syntax analyzer. So, each call to the lexical analyzer

returns a single lexeme and its token. The only view of the input program seen by the syntax analyzer is the output of the lexical analyzer, one token at a time. The lexical-analysis process includes skipping comments and white space outside lexemes, as they are not relevant to the meaning of the program. Also, the lexical analyzer inserts lexemes for user-defined names into the symbol table, which is used by later phases of the compiler. Finally, lexical analyzers detect syntactic errors in tokens, such as ill-formed floating-point literals, and report such errors to the user.

Purpose of Lexical Analyzer  
z Main task: to read input characters and group them into “tokens.”  
z Secondary tasks: z Skip comments and whitespace; z Correlate error messages with source program (e.g., line number of error).

3.

**Explain how the coercion rules of a language affect its error detection. (4 marks)**

The coercion also results in a loss of one of the benefits of strong typing—error detection. For example, suppose a program had the `int` variables `a` and `b` and the `float` variable `d`. Now, if a programmer meant to type `a + b`, but mistakenly typed `a + d`, the error would not be detected by the compiler. The value of `a` would simply be coerced to `float`. So, the value of strong typing is weakened by coercion.

Languages with a great deal of coercion, like C, and C++, are less reliable than those with no coercion, such as ML and F#. Java and C# have half as many assignment type coercions as C++, so their error detection is better than that of C++, but still not nearly as effective as that of ML and F#. The issue of coercion is examined in detail in Chapter 7.

4.

**What is dynamic scoping and a dynamic ancestor of a subprogram?**

**Show an example! (2 points)**

**Dynamic scoping** is based on the calling sequence of subprograms, not on their spatial relationship to each other. Thus, the scope can be determined only at run time.

- ▶ Dynamic parent of a subprogram is the subprogram that call him. The dynamic parent of subprogram such as `a`, and its dynamic parent, and so forth up to the including the `main`, are called the dynamic ancestors of `a`.

Procedure `main()`

```

Var x : integer
Procedure ali
Begin
...
end
Procedure ahmed
begin
.....Ali().....
end
End
Begin main
..:Ahmed();
End
Dynamic ancestors of Ahmed are : main
Dynamic ancestors of Ali are : Ahmed and main

```

5.

#### What is a coroutine? (2 marks)

A **coroutine** is a special kind of subprogram. Rather than the master-slave relationship between a caller and a called subprogram that exists with conventional subprograms, caller and called coroutines are more equitable. In fact, the coroutine control mechanism is often called the **symmetric unit control model**.

6.

Consider the following C program, which variable has the longest scope?

Explain your answer with details! [3 marks]

```

int a;
int main()
{
    int b;
    // ..
    // ..
}
int c;

```

a is accessible everywhere.

b is limited to main()

c is accessible only after its declaration.

7.

Consider the program below in a hypothetical language which allows global variable and a choice of call by reference or call by value methods of parameter passing.

What will be the output of the program for the two parameter passing mechanisms?

Explain it for pass-by-value and pass-by-reference as well! [4 marks]

```
int i ;
program main ()
{
    int j = 60;
    i = 50;
    call f (i, j);
    print i, j;
}
procedure f (x, y)
{
    i = 100;
    x = 10;
    y = y + i ;
}
```

CALL BY VALUE :-  $i$  as global variable declared. Then in  $main()$  a local variable  $j$  as integer declared i.e  $j = 60$  And global variable  $i$  initialized to 50 by  $i = 50$ . Now procedure  $f$  called and values of  $i$  and  $j$  are passed to it. i.e., in  $f(i, j) \rightarrow f(x, y)$  content of memory location of  $i$  (here 50) is copied to memory location of  $x$  (which is different from  $i$ ) and content of memory location of  $j$  (here, 60) is copied to memory location of  $y$ . Then in  $f(x, y)$   $i = 100$  changes the global  $i$  to 100,  $X = 10$  changes the local  $X$  from 50 to 10 and  $Y = y + i$  means  $y = 60 + 100 = 160$ . Now when return back to main,  $i$  and  $j$  will be 100 and 60 respectively.

CALL BY REFERENCE:- Now procedure  $f$  called and passed reference of  $i$  and  $j$  to it. i.e., in  $f(i, j) \rightarrow f(x, y)$   $x$  and  $y$  are new names (aliases) pointing to the same memory location of  $i$  and  $j$  respectively. So,  $i = 100$  changes the global  $i$  to 100 and  $x = 10$  means  $x$  as well as global  $i = 10$  (as the  $i$  being passed is the global variable and  $x$  and  $i$  share the same address).

$y = y + i$  means  $y = 60 + 10 = 70$  and this changes the value of  $j$  also to 70 as  $j$  and  $y$  have the same address. Now when return back to main,  $i$  and  $j$  will be 10 and 70 respectively.

8.

Explain why dynamic type bindings is closely related to implicit heap-dynamic variables, using the following code example:

```
var a;  
....  
a = "Hello";  
....  
a = 123.456;
```

Variable a gets its type at runtime, i.e. dynamic type binding, when a value is assigned to it. Since variable a may often change the types of values it holds, hence the memory locations where the values are stored, a must be a reference variable and its dynamically assigned value must be in a dynamically allocated memory location, i.e. the heap.

9.

What is the Pass-by-Reference model of parameter passing? [describe it]  
Explain in your answer the parameter collision problem!

(4 points)

Pass by reference (also called pass by address) means to pass the reference of an argument in the calling function to the corresponding formal parameter of the called function so that a copy of the address of the actual parameter is made in memory, i.e. the caller and the callee use the same variable for the parameter.

Parameter collision problem occurs in pass-by-result model of parameter passing. When we have the same actual parameters in the calling function and different formal parameters in the called function, the order in which the actual parameters are copied determines the value.

10.

What is an overloaded subprogram? (2 marks)

Overloaded subprogram refers to a subprogram that has the same name as other subprograms but has different number, type, order of parameters or return value in the same programming circumstance.

11.

Write a simple assignment statement with one arithmetic operator in some language you know. For each component of the statement, list the various bindings that are required to determine the semantics when the statement is executed. For each binding, indicate the binding time used for the language. (4 marks)

```
count = count + 5;
```

Some of the bindings and their binding times for the parts of this assignment statement are as follows:

- The type of count is bound at compile time.
- The set of possible values of count is bound at compiler design time.
- The meaning of the operator symbol + is bound at compile time, when the types of its operands have been determined.
- The internal representation of the literal 5 is bound at compiler design time.
- The value of count is bound at execution time with this statement.

(C++)

```
int count;  
count = count + 5;
```

**Possible types for count:** set at language design time. **Type of count:** bound at compile time.

**Set of possible values of count:** bound at compiler design time. **Value of count:** bound at execution time with this statement. **Set of possible meanings for the operator symbol “+”**: \*bound at language definition time.\***Meaning of the operator symbol “+” in this statement:** bound at compile time.

**Internal representation of the literal “5”:** bound at compiler design time.

12. Type Equivalence

Two types are **equivalent** if an operand of one type in an expression is substituted for one of the other type, without coercion. Type equivalence is a strict form of type compatibility—compatibility without coercion.

- 1) Which produces faster program execution: a compiler or a pure interpreter? (1 point)  
[Explain your answer!] (3 points)
- 2) Define the meaning of syntax and semantics! (2 points)
- 3) How can you describe a variable? What is a variable? (1 point)  
List and explain all the attributes of a variable (excl. lifetime and scope) (4 points)
- 4) What is the definition of block? (2 points)
- 5) What does it mean when a variable is static? (Explain its advantages and disadvantages)  
(4 points)
- 6) Define scope and lifetime! (2 points)  
Explain C's scoping and lifetime in the following code fragment: (3 points)

```
void sub () {  
    int count;  
    ...  
    while( ... ) {  
        int count;  
        count ++  
        ...  
    }  
    ...  
}  
void printhead () {  
    int sum;  
    ...  
    sub ();  
}
```
- 7) What are the design issues for multiple-selection statements? (2 points)  
(Explain your answers for C's array!) (3 points)
- 8) How does C support relational and Boolean expressions? (2 points)
- 9) What are the two common problems with pointers? (2 points)  
Explain your answer! (2 points)
- 10) in what way is C's for statement more flexible than that of many other languages?  
[Show examples!] (3 points)

## Programming Languages 1 Part II. - Questions

1) Explain the three reasons why lexical analysis is separated from syntax analysis. (4 point)  
[ Explain your answer with examples for each one]

2) What is a reserved word? (1 points)

3) What is a static ancestor of a subprogram?

What is a dynamic ancestor of a subprogram? (2 points)

4) What are the advantages and disadvantages of dynamic type binding? (2 points)

5) What is the referencing environment of a statement? (2 points)

6) What is an overloaded operator? (2 points)

7) What does it mean when a variable is stack dynamic?

(Explain its advantages and disadvantage) (3 points)

8) Define scope and lifetime! (2 points)

Explain C's scoping and lifetime for all sum variables in the code Fragment: (3 points)

```
void sub () {  
    int sum;  
    ...  
    while(...) {  
        int sum;  
        sum ++  
        ...  
    }  
    ...  
}  
void printhead () {  
    int sum;  
    ...  
    sub ();  
}
```

9) What is the Pass-by-Result model of parameter passing? (6 points)

Explain in your answers the parameter collision problem!

10) How does short-circuit expression evaluation works? (2 points)

11) What is a generic subprogram? (2 points)

12) What is a coroutine? (2 points)

13) In what way is C's for statement more flexible than that of many other languages? [Show examples:] (3 points)

## 1. Define the concepts of type binding for static and dynamic cases!

[Explain your answer with examples for each one! (expl./impl.; type inf.; compil,intrprt)]

A binding is an association, such as between an attribute and an entity, or between an operation and a symbol.

- A binding is static if it first occurs before run time and remains unchanged throughout program execution. Example for c
- A binding is dynamic if it first occurs during execution or can change during execution of the program. Example for python.
- An explicit declaration is a program statement used for declaring the types of variables.
- An implicit declaration is a default mechanism for specifying types of variables (the first appearance of the variable in the program.)
- For type Inference Rather than by assignment statement, types are determined from the context of the reference.
- In Compilation, programs are translated into machine codes. Using a compiler like gcc, we use the command (gcc hello.c -o hello) then executing the generated binary file like (./hello)
- Pure Interpretation: Programs are interpreted by another program known as an interpreter.

## 2. Define steps to the semantics of a call to a “simple” subprogram!

1. Save the execution status of the current program unit.
2. Compute and pass the parameters.
3. Pass the return address to the caller.
4. Transfer control to the called.

```
int add_numbers(int num1, int num2) {  
    int result = num1 + num2;  
    return result;}  
  
int main() {  
    int x = 5;  
    int y = 10;  
    int sum_result = add_numbers(x, y);  
    printf("The sum is: %d\n", sum_result);  
    return 0;}
```

### 3- How can you describe a variable? What is a variable?

List and explain all the attributes of a variable (incl. lifetime and scope).

A variable is an abstraction of a memory cell(s).

1- Name: Is a sequence of characters used as an identifier.

2- Address: Is the memory address with which it is associated with.

3- Value: Is the content of the variable.

4- Type: Is the range of values that a variable can be stored.

5- Lifetime: Is the time duration in which a variable is bound to a specific memory location.

6- Scope: Defines where a variable can be accessed or referenced.

### 4- What is a record as a data type?

Compare it with arrays.

A record is a possibly heterogeneous aggregate of data elements in which the individual elements are identified by names. An array is a data type used to store a collection of elements of the same type in a contiguous memory block. In summary, arrays are used to store elements of the same type in a sequential manner, while records allow you to group elements of different types together into a single entity.

### 5- What does it mean when a variable is stack dynamic?

Explain its adv and dis.

Storage bindings are created for variables when their declaration statements are elaborated, but whose types are statically bound. Stack-dynamic variables are allocated from the run-time stack.

Adv- Allows recursion: each active copy of the recursive subprogram has its own version of the local variables.

Dis- Overhead of allocation and deallocation.

Subprograms cannot be history sensitive.

**6- Define the two fundamental kinds of subprograms!**

**What are the similarities and differences between them?**

**Procedures:** Subprograms that perform specific tasks or actions, typically used for executing a series of statements or operations without returning a value.

**Functions:** Subprograms that compute and return a value based on input parameters, used for calculating results that can be used in expressions or assigned to variables.

**Similarities:**

- Both are subprograms.
- Have a name and can be called.
- Can accept input parameters.

**Differences:**

- Procedures modify state, functions compute and return values.
- Procedures have no return value; functions have a return value.
- Functions are used in expressions or assigned to variables; procedures are invoked for their execution.

**7- What is an associative array?**

**Explain your answers for a chosen language.**

An associative array is an unordered collection of data elements that are indexed by an equal number of values called keys. Associative arrays are supported by the standard class libraries of Java and C++ and Perl. In Java for example, an associative array is called a "Map." It stores key-value pairs, where each key is unique. You can retrieve values by providing their keys.

**8- what could be a problem with nested selection statement in c-based languages?**

If the selection statement is nested in "then", it is not clear which if an "else" be associated with. Solution to this is to put an inner if compound.

```
If(num==1){  
    If(result == 1)  
        Return 1;  
    Else return 0;
```

## **9- What are special words?**

**Explain your answer with its subtypes and examples!**

Special words are used to separate the syntactic part of statements and program.

An aid to readability; used to delimit or separate statement clauses. A keyword is a word that is special only in certain contexts. **Ex: break, void, default, switch, case....**

**Ex: Fortran**

## **10- Explain the pass-by-result model in detail with examples!**

**Discuss the well-known potential problem with it!**

Pass-by-Result is an implementation model for out-mode parameters.

- When a parameter is passed by result, no value is transmitted to the subprogram.
- The corresponding formal parameter acts as a local var, but just before control is transferred back to the caller, its value is transmitted back to the caller's actual parameter, which must be a var.
- One problem with the pass-by-result model is that there can be an actual parameter collision, such as the one created with the call.

**Example:**

```
int func(int a, int b){  
    a = 5;  
    b = 10;  
    ...  
}  
int main(){  
    int x, y, z;  
    z = func(x, y)  
}
```

### **1- What does it mean when a variable is heap-dynamic?**

**Explain it's adv and dis.**

Nameless memory cells that are allocated and deallocated by explicit directives “run-time instructions”, specified by the programmer, which take effect during execution.

The heap is a collection of storage cells whose organization is highly disorganized b/c of the unpredictability of its use.

Adv- Provides for dynamic storage management.

Dis- Inefficient “Cost of allocation and deallocation” and unreliable “difficulty of using pointer and reference variables correctly.”

### **2- What is a union type?**

It is a type where variables are allowed to store different types of values at different times of execution.

Ex: c and c++ provide union constructs in which there is no language support for type checking, union in these languages is called free union.

### **3- What is a reserved word? Explain your answer with examples and indicating the differences from keywords!**

Reserved words are words in a programming language that are reserved and cannot be used as user-defined names. They are crucial for the language's syntax and grammar. Keywords are a subset of reserved words with specific meanings or functionalities within the language. In summary, while reserved words encompass all the words reserved by a programming language, keywords are a subset of reserved words with specific functionalities.

**4- Explain the pass-by-reference model in detail with examples!**

**Discuss the well-known potential problem with it!**

Pass-by-reference is a second implementation model for inout-mode parameters.

Rather than copying data values back and forth. This method transmits an access path, sometimes just an address, to the called subprogram. This provides the access path to the cell storing the actual parameter.

The actual parameter is shared with the called subprogram.

Problems: Access to the formal parameters will be slower than pass-by-value, because of the additional level of indirect addressing that is required.

Inadvertent and erroneous changes may be made to the actual parameter.

Example:

```
int func(int &a, int &b){  
...  
}  
int main(){  
int x, y, z;  
z = func(x, y)  
}
```

### 1- What is the definition of block?

A block is a compound statement that lets you group any number of data declarations into one statement using curly brackets {}.

Allows a section of code to have its own local vars whose scope is minimized.

### 2- What does it mean when a variable is static? (adv and dis.)

Static variables: variables that are bound to memory cells before program execution begins and remain bound to those same memory cells until program execution terminates.

Adv: memory efficiency, persistence, accessibility, can be addressed directly, no runtime overhead for allocation and deallocation.

Dis: reduced flexibility, can't be used for recursive subprograms, memory can't be carried.

### 3- How does C support relational and Boolean expressions?

For C it doesn't have built in Boolean expressions, but it supports it by integer values, relation operators and logical operators.

All operands with nonzero values are considered true, and zero is considered false.

example for relation operators in C: <, ==, >=

example for logical operators in C: &&, ||, !

### 4- Which produces faster program execution: a compiler or a pure interpreter?

[explain your answer!]

The compiler will produce faster program execution due to the fact that translating high level language to machine code is more complex. So, the compiler is faster as pure interpreter reads the source code line by line to check for errors and executes it as it goes while compiler takes the source code and converts it into machine code. Pure interpretation typically takes at least ten times as long as to execute equivalent machine code.

**5- What are the two common problems with pointers? Explain with details.**

- Dangling pointer: pointer that points to keep dynamic variable that have been deallocated the dangling pointer is dangerous because it is used to change the heap dynamic variable, the value of the heap dynamic variable will be destroyed.
- Lost heap dynamic variable: if the heap dynamic variable that is no longer accessible by the program. But may have some useful data. The variable at the time is often called garbage.

The process of LHDV is called memory leakage.

**6- Explain the pass-by-value model in detail with examples!**

**Is parameter collision present in this mode? Explain your answer!**

It is an implementation model of in-mode. Pass-by-value is implemented by copying, the original value remains unchanged. The value of actual parameter initializes formal parameter. Therefore, parameter collision is not present in this mode.

Advantage: for scalars it is fast.

Disadvantages: Additional storage required, storage and copy operations can be costly.

```
int func(int a, int b){  
...  
}  
int main(){  
int x, y, z;  
x = 5;  
y = 10;  
z = func(x, y)  
}
```

## **1- Define syntax and semantics!**

Syntax is the form of the expression, statements, and program units. It is a term we use to refer to grammar.

Semantics is the meaning of the expression, statements, and program units. It basically refers to the meaning of the sentence.

## **2- what are the design issues for multiple-select statements?**

[explain your answers for c especially].

- What is the form and type of the control expression?
- How are the selectable segments specified?
- Is execution flow through the structure restricted to include just a single selectable segment?
- What is done about unrepresented expression values?

The C switch statement has virtually no restrictions on the placement of the case expressions, which are treated as if they were normal statement labels.

## **3- what are the design issues for subprograms?**

1. What parameter passing methods are provided?
2. Are parameter types checked?
3. Are local variables static or dynamic?
4. Can subprogram definitions appear in other subprogram definitions?
5. What is the referencing environment of a passed subprogram?
6. Can subprograms be overloaded?
7. Are subprograms allowed to be generic?

**4- In what way is C's for statement more flexible than that of many other languages?  
[show examples!]**

C's for is more flexible because each of the expressions can comprise multiple statements, which in turn allow multiple loop vars that can be of any type.

example:

```
for (i = 0; i < 5; i++) {  
    printf("Hello, world!\n");}
```

This will print Hello, world 5 times and stop.

**What does it mean when a variable is explicit heap-dynamic?**

Explain its adv and dis.

Nameless memory cells that are allocated and deallocated by "run-time instructions".

Advantage: – Provides for dynamic storage management.

Disadvantage: – Inefficient "Cost of allocation and deallocation" and unreliable "difficulty of using pointer and reference variables correctly"

**What does it mean when a variable is implicit heap-dynamic?**

Explain its adv and dis.

Variables are bound to heap storage only when they are assigned value.

Advantage: – Flexibility allowing generic code to be written.

Disadvantages: Inefficient because all attributes are dynamic "run-time."

Loss of error detection by the compiler.

**What is a static Array?**

A static array is one in which the subscript ranges are statically bound and storage allocation is static (done before run time).

Advantages: efficiency "No allocation & deallocation."

Ex: Arrays declared in C & C++ function that includes the static modifier are static.

**What is coroutine?**

A coroutine is a subprogram that has multiple entries.

### **What is static ancestor?**

The static parent of subprogram Sub1, and its static parent, and so forth up to and including the main program, are called the static ancestors of Sub1.

### **What is a literal?**

### **Lexical analyzer, lexeme, and token!**

- lexical analysis: converts characters in the source program into lexical units.

The lexical units of a program are identifiers, special words operators, and punctuation symbols.

- Lexeme is the lowest level syntactic unit of a language. It includes identifiers, literals, operators, and special words. (e.g. \*, sum, begin) A program is strings of lexemes.

- A token is a category of lexemes.

### **Define the referencing environment!**

It is the collection of all names that are visible in the statement.

In a static-scoped language, it is the local variables plus all the visible variables in all the enclosing scopes.

In a dynamic-scoped language, the referencing environment is the local variables plus all visible variables in all active subprograms.

### **Overloaded subprograms!**

In overloaded subprogram is a subprogram that has the same name as another subprogram in the same referencing environment.

### **Overloaded operators!**

An overloaded operator is one that has multiple meanings.

### **Generic Subprograms!**

It takes parameters of different types on different activations.

PreExam - HL PL 1 - Essay - 2021.05.11. (page 1 of 11) - Google Chrome  
elearning.unideb.hu/mod/quiz/attempt.php?attempt=425339&cmid=406125

## High-level programming languages 1 - Joint lecture (INBPA0211E - INBMA0211E) 2021 Spring

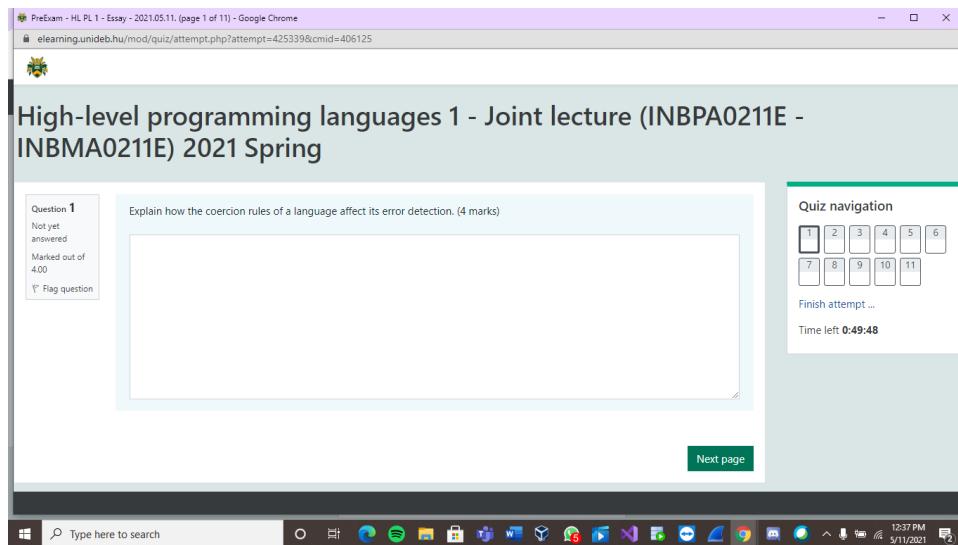
Question 1  
Not yet answered  
Marked out of 4.00  
Flag question

Explain how the coercion rules of a language affect its error detection. (4 marks)

Quiz navigation  
1 2 3 4 5 6  
7 8 9 10 11

Finish attempt ...  
Time left 0:49:48

Next page



PreExam - HL PL 1 - Essay - 2021.05.11. (page 2 of 11) - Google Chrome  
elearning.unideb.hu/mod/quiz/attempt.php?attempt=425339&cmid=406125&page=1

## High-level programming languages 1 - Joint lecture (INBPA0211E - INBMA0211E) 2021 Spring

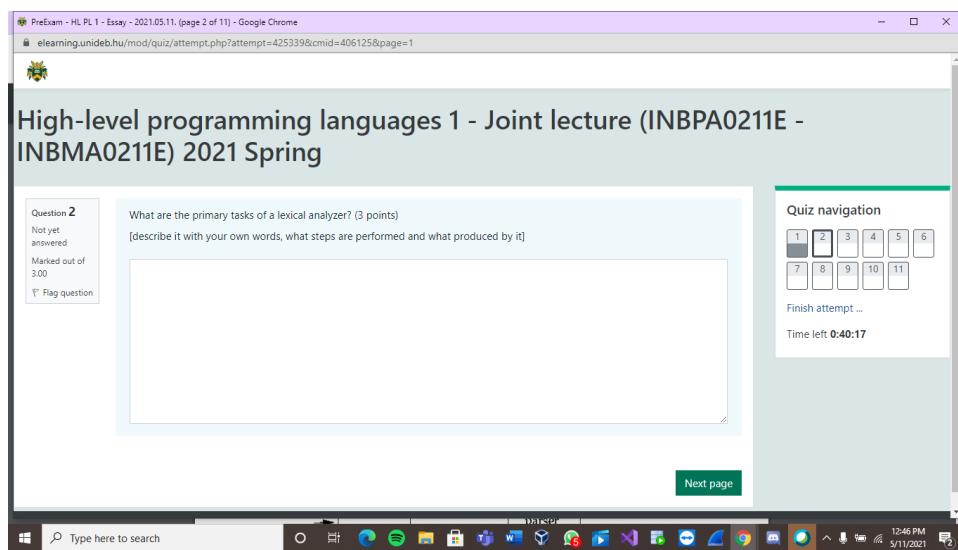
Question 2  
Not yet answered  
Marked out of 3.00  
Flag question

What are the primary tasks of a lexical analyzer? (3 points)  
[describe it with your own words, what steps are performed and what produced by it]

Quiz navigation  
1 2 3 4 5 6  
7 8 9 10 11

Finish attempt ...  
Time left 0:40:17

Next page



PreExam - HL PL 1 - Essay - 2021.05.11. (page 2 of 11) - Google Chrome  
elearning.unideb.hu/mod/quiz/attempt.php?attempt=425339&cmid=406125&page=1

## High-level programming languages 1 - Joint lecture (INBPA0211E - INBMA0211E) 2021 Spring

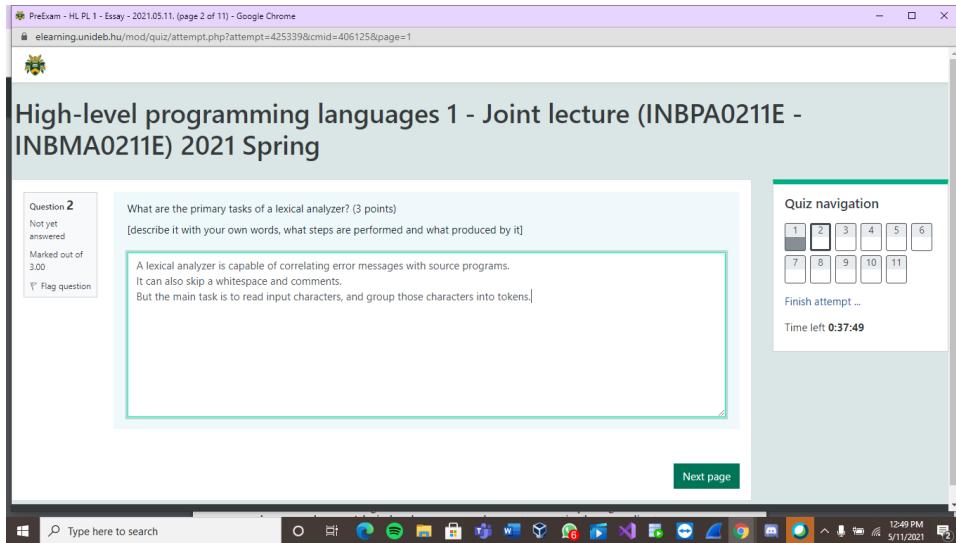
Question 2  
Not yet answered  
Marked out of 3.00  
Flag question

What are the primary tasks of a lexical analyzer? (3 points)  
[describe it with your own words, what steps are performed and what produced by it]

A lexical analyzer is capable of correlating error messages with source programs.  
It can also skip whitespace and comments.  
But the main task is to read input characters, and group those characters into tokens.

Quiz navigation  
1 2 3 4 5 6  
7 8 9 10 11  
Finish attempt ...  
Time left 0:37:49

Next page



PreExam - HL PL 1 - Essay - 2021.05.11. (page 4 of 11) - Google Chrome  
elearning.unideb.hu/mod/quiz/attempt.php?attempt=425339&cmid=406125&page=3

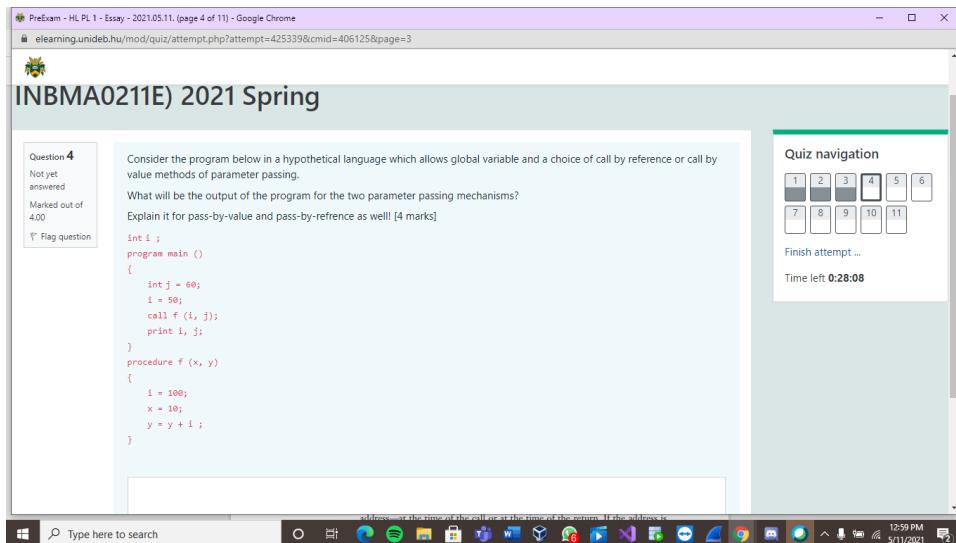
## INBMA0211E) 2021 Spring

Question 4  
Not yet answered  
Marked out of 4.00  
Flag question

Consider the program below in a hypothetical language which allows global variable and a choice of call by reference or call by value methods of parameter passing.  
What will be the output of the program for the two parameter passing mechanisms?  
Explain it for pass-by-value and pass-by-reference as well! [4 marks]

```
int i ;  
program main ()  
{  
    int j = 60;  
    i = 50;  
    call f (i, j);  
    print i, j;  
}  
  
procedure f (x, y)  
{  
    i = 100;  
    x = 10;  
    y = y + i ;  
}
```

Quiz navigation  
1 2 3 4 5 6  
7 8 9 10 11  
Finish attempt ...  
Time left 0:28:08



PreExam - HL PL 1 - Essay - 2021.05.11. (page 5 of 11) - Google Chrome  
elearning.unideb.hu/mod/quiz/attempt.php?attempt=425339&cmid=406125&page=4

## INBMA0211E) 2021 Spring

Question 5  
Not yet answered  
Marked out of 3.00  
Flag question

Consider the following C program, which variable has the longest scope?  
Explain your answer with details! [3 marks]

```
int a;
int main()
{
    int b;
    ...
}
int c;
```

The variable a. This is because b is limited in main function, and c has to be declared at the end of the program.

Next page

1:12 PM 5/11/2021

PreExam - HL PL 1 - Essay - 2021.05.11. (page 7 of 11) - Google Chrome  
elearning.unideb.hu/mod/quiz/attempt.php?attempt=425339&cmid=406125&page=6

## High-level programming languages 1 - Joint lecture (INBPA0211E - INBMA0211E) 2021 Spring

Question 7  
Not yet answered  
Marked out of 2.00  
Flag question

Dynamic type binding is closely related to implicit heap-dynamic variables. Explain this relationship. (2 marks)

A *V* **B** *I* *U* *S* *I*

Finish attempt ...  
Time left 0:13:07

1:14 PM 5/11/2021

PreExam - HL PL 1 - Essay - 2021.05.11. (page 9 of 11) - Google Chrome

learning.unideb.hu/mod/quiz/attempt.php?attempt=425339&cmid=406125&page=8



## High-level programming languages 1 - Joint lecture (INBPA0211E - INBMA0211E) 2021 Spring

**Question 9**  
Not yet answered  
Marked out of 2.00  
 Flag question

What is the l-value of a variable?  
What is the r-value?  
(2 marks)

**Quiz navigation**

1	2	3	4	5	6
7	8	9	10	11	

[Finish attempt ...](#)  
Time left 0:09:17

[Next page](#)



PreExam - HL PL 1 - Essay - 2021.05.11. (page 10 of 11) - Google Chrome

learning.unideb.hu/mod/quiz/attempt.php?attempt=425339&cmid=406125&page=9



## High-level programming languages 1 - Joint lecture (INBPA0211E - INBMA0211E) 2021 Spring

**Question 10**  
Not yet answered  
Marked out of 4.00  
 Flag question

Write a simple assignment statement with one arithmetic operator in some language you know. For each component of the statement, list the various bindings that are required to determine the semantics when the statement is executed. For each binding, indicate the binding time used for the language. (4 marks)

**Quiz navigation**

1	2	3	4	5	6
7	8	9	10	11	

[Finish attempt ...](#)  
Time left 0:07:53

[Next page](#)



PreExam - HL PL 1 - Essay - 2021.05.11. (page 11 of 11) - Google Chrome  
elearning.unideb.hu/mod/quiz/attempt.php?attempt=425339&cmid=406125&page=10



## High-level programming languages 1 - Joint lecture (INBPA0211E - INBMA0211E) 2021 Spring

**Question 11**  
Not yet answered  
Marked out of 2.00  
[Flag question](#)

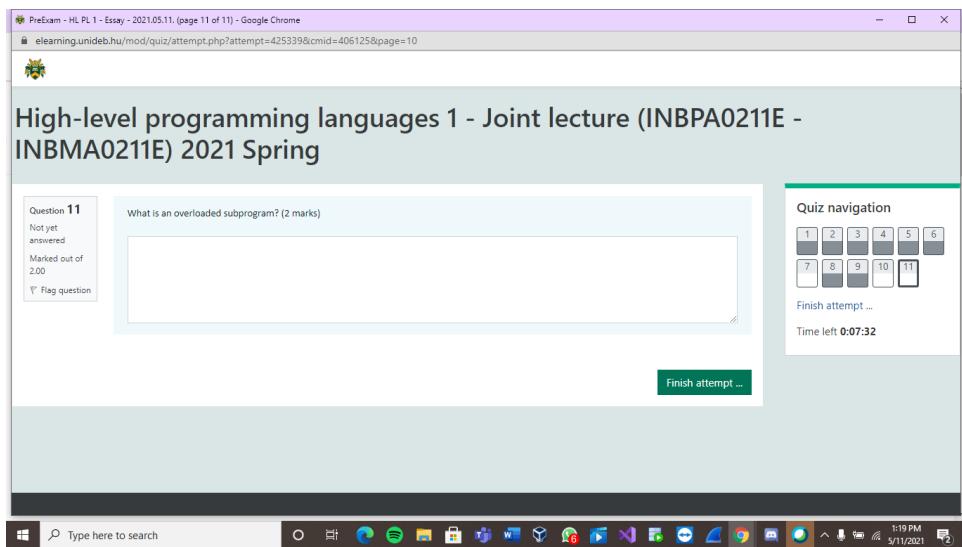
What is an overloaded subprogram? (2 marks)

**Quiz navigation**

1	2	3	4	5	6
7	8	9	10	11	

[Finish attempt ...](#)

Time left 0:07:32



# **HLPL**

## **Chapter 1 review questions:**

1. Why is it useful for a programmer to have the ability to learn new languages, even though he or she may have a good knowledge of a number of programming languages?

Increased capacity to express ideas, better understanding of the significance of implementing, better use of languages already known, overall advancement of computing

2. Why is it essential to choose an appropriate programming language for a specific software solution?

The right selection of a programming language yields solutions that are concise, easy to debug, easy to extend, easy to document, and easy to fix

3. Which programming language for scientific applications was the first to be used successfully?

Fortran

4. Which is the first successful high-level programming language for business?

COBOL

5. In which programming language were most of the AI applications developed prior to 1990?

LISP

6. Which is the most popular markup language for Web development?

HTML

7. Why is a list of programming language evaluation criteria for the development of software controversial?

Criteria is better precisely defined nor exactly measurable

8. How does the overall simplicity of a programming language affect its readability?

Number of basic constructs/features, feature multiplicity, operator overloading.

**9. Why is the VAX instruction design orthogonal?**

Because the instruction for 32 bit integer addition of the form where each of the op's can refer to either a register or a memory location.

**10. Why does too much orthogonality cause problems?**

Intuition leads one to ascribe certain advantages to orthogonality: the reduction in the number of special rules or exceptions to rules should make a language easier "to describe, to learn, and to implement" — in the words of the Algol 68 report. On the other hand, strict application of the orthogonality principle may lead to constructs which are conceptually obscure when a rule is applied to a context in an unusual combination. Likewise the application of orthogonality may extend the power and generality of a language beyond that required for its purpose, and thus may require increased conceptual ability on the part of those who need to learn and use it.

**11. Explain how "writability" is used as a measure of how easily language can be used to create programs.**

The easier a program is to write, the more likely it is to be correct. Readability affects writability in both development and maintenance phases of the software cycle.

**12. Why is too much orthogonality a detriment to "writability"?**

Too much orthogonality can be detrimental: if all combinations are meaningful, errors are difficult to detect

**13. Give an example of expressivity in a language.**

It means that there are very powerful operators that allow a great deal of computation to be accomplished with a very small program. in C, the notation count++ is more convenient and shorter than count = count + 1

**14. What is type checking?**

Type checking is a program analysis that verifies something about the types that are used in the program

**15. Give an example of how the failure to type check, at either compile time or run time, can lead to countless program errors**

If and \*int\* is passed in a \*float\*'s stead the result will be non-sense

**16. How is the total cost of a programming language calculated?**

Smalltalk, developed by Goldberg and Robson in 1989.

- Training programmers
- writing programs
- compiling

- executing
- implementation system
- reliability (or lack of)
- maintenance

17. What is portability of a language?

A programming language capable of developing software for more than one computer system.

18. What is the use of the well-definedness criterion?

the completeness and precision of the language's official defining document

19. How does the execution of a machine code program on a von Neumann architecture computer occur?

it occurs in a process called the fetch-execute cycle for the programs reside in memory but are executed in the CPU.

20. What two programming language deficiencies were discovered as a result of the research in software development in the 1970s?

Type checking and control statements

21. What are the three fundamental features of an object-oriented programming language?

Polymorphism, inheritance, and encapsulation

22. What language was the first to support the three fundamental features of object-oriented programming?

Smalltalk

23. What is an example of two language design criteria that are in direct conflict with each other?

the ease with which programs that work on one platform can be modified to work on another.  
This is strongly influenced by to what degree a language is standardized.

24. What are the three general methods of implementing a programming language?

Three general methods of implementing a programming language are compilation, pure interpretation, and hybrid implementation.

25. Which produces faster program execution, a compiler or a pure interpreter?

A compiler will produce faster program execution due to the fact that translating high level language to machine code is substantially more complex.

26. What role does the symbol table play in a compiler?

The symbol table serves as a database for the compilation process in a compiler.

27. What does a linker do?

A program used with a compiler or assembler to provide links to the libraries needed for an executable program.

28. Why is the von Neumann bottleneck important?

Keeps us thinking in a word-at-a-time terms in much the same way the computer's bottleneck does

29. What are the advantages in implementing a language with a pure interpreter?

One of the advantages of implementing a language with a pure interpreter is easy implementation of many source-level debugging operations, because all run-time errors can refer to the source-level units. An example of this is an array index is found to be out of range.

# HLPL

## Chapter 3 review questions:

1. Define *lexeme* and *token*.

A Lexeme is a string of characters that is a lowest-level syntactic unit in the programming language.

A Token is a syntactic category that forms a class of lexemes.

2. How are programming languages formally defined?

Two distinct ways: by recognition and by generation

3. In which form is the programming language syntax commonly described?

by context-free grammars.

4. What is a metalanguage?

A language that is used to describe another language

5. What is a derivation in the context of grammar?

A derivation is a repeated application of rules, starting with the start symbol and ending with a sentence (all terminal symbols)

[copy from the book ^^^ !!! ]

6. What is an ambiguous grammar?

A grammar that generates a sentential form for which there are two or more distinct parse trees

**7. What is a left-recursive grammar?**

When a grammar rule has its LHS also appearing at the beginning of its RHS, the rule is said to be left recursive

**8. Explain the use of metasymbols in EBNFs.**

Notational tools and non terminal symbols in the syntactic entities they help describe . These include brackets, braces, and parentheses in the EBNF extensions.

**9. What is the purpose of a predicate function?**

States the static semantic rules of the language and associated with grammar rules.

**10. When can the parse tree be called fully attributed?**

If all the attribute values in the parse tree have been computed

**11. How is the order of evaluation of attributes determined for the trees of a given attribute grammar?**

With the construction of a dependency graph to show all attribute dependencies.

**12. What is the use of intrinsic attributes?**

Can be used to compute the remaining attribute values.

**13. What is meant by decorating a parse tree?**

The process of computing the attribute values of a parse tree.

**14. Why can machine languages not be used to define statements in operational semantics?**

First the individual steps in the execution of machine language and the resulting changes to the state of the machine are too small and too numerous. Second the storage of a real computer is too large and complex.

15. Describe the two levels of uses of operational semantics.

At the highest level, the interest is in the final result of the execution of a complete program. This is called **natural operational semantics**. At the lowest level, operational semantics can be used to determine the precise meaning of a program through an examination of the complete sequence of state changes that occur when the program is executed. This use is called **structural operational semantics**.

16. In denotational semantics, what are the syntactic and semantic domains?

The domain is called the syntactic domain, because it is syntactic structures that are mapped. The range is called the semantic domain.

17. What is stored in the state of a program for denotational semantics?

The values of all of the program's variables.

18. What is an assertion in axiomatic semantics?

Describes the constraints on the program variables at that point in the program.

19. What two things must be defined for each language entity in order to construct a denotational description of the language?

Both a mathematical object and a function that maps instances of that language entity onto instances of the mathematical object.

20. Which part of an inference rule is the antecedent?

The top part of an inference rule.

21. What is the weakest precondition?

The least restrictive precondition that will guarantee the validity of the associated postcondition.

22. What is an inference rule?

A method of inferring the truth of one assertion on the basis of the values of other assertions.

23. Give an example of a logical pretest loop.

M<sub>sl</sub> and M<sub>b</sub>, that map statement lists and states to states and Boolean expressions to Boolean values (or error), respectively.

24. On what branch of mathematics is denotational semantics based?

Recursive function theory.

25. What is the problem with using a software pure interpreter for operational semantics?

**The detailed characteristics of the particular computer would make actions difficult to understand.** Such a semantic definition would be machine-dependent.

26. Explain what the preconditions and postconditions of a given statement mean in axiomatic semantics.

The precondition statement indicates what must be true before the function is called. The postcondition statement indicates what will be true when the function finishes its work.

27. What is the difference between total correctness and partial correctness with regard to loop termination?

A distinction is made between partial correctness, which requires that if an answer is returned it will be correct, and total correctness, which additionally requires that the algorithm terminates.

28. When is a function called a predicate transformer?

When it takes a predicate, or assertion, as a parameter and returns another predicate.

29. In what fundamental way do operational semantics and denotational semantics differ?

Unlike operational semantics, denotational semantics does not model the step- by-step computational processing of programs.

# HLPL

## Chapter 4 review questions:

1. What are the reasons why using BNF is advantageous over using an informal syntax description?  
(1) Syntax description with context free grammars are precise and unambiguous. This is easy to understand for humans and software systems.  
(2) The formal description of the syntax, CFG or BNF are both used as the direct basis of syntax analyzer  
(3) BNF are easy to implement in modules.
2. How does a lexical analyzer serve as the front end of a syntax analyzer?

The syntax analyzer takes the lexical units from the lexical analyzer and uses them to construct hierarchical structures called parse trees.

3. Define *finite automata* and *regular grammar*.

State diagrams of the form used for lexical analyzers are representations of a class of mathematical machines called finite automata.

Finite automata can be designed to recognize members of a class of languages called regular languages.

regular grammar: generative devices for regular languages.

4. How can you construct a lexical analyzer with a state diagram?

Idk :/

5. Describe briefly the three approaches to building a lexical analyzer.

1. Write a formal description of the token patterns of the language using a descriptive language related to regular expressions.
2. Design a state transition diagram that describes the token patterns of the language and write a program that implements the diagram.
3. Design a state transition diagram that describes the token patterns of the language and hand-construct a table-driven implementation of the state diagram.

6. What are the different grammar symbols for formal languages?
  1. Terminal symbols—lowercase letters at the beginning of the alphabet (a, b, . . . )
  2. Nonterminal symbols—uppercase letters at the beginning of the alphabet (A, B, . . . )
  3. Terminals or nonterminals—uppercase letters at the end of the alphabet (W, X, Y, Z)
  4. Strings of terminals—lowercase letters at the end of the alphabet (w, x, y, z)
  5. Mixed strings (terminals and/or nonterminals)—lowercase Greek letters (a, b, d, g)
7. Why are character classes used, rather than individual characters, for the letter and digit transitions of a state diagram for a lexical analyzer?  
a lexical analyzer is interested  
only in determining that it is a name and is not concerned with which specific name it happens to be.

Therefore, we define a character class named LETTER  
for all 52 letters and use a single transition on the first letter of any name.

8. What are the two distinct goals of syntax analysis?  
First, syntax analyzer must check the input program to determine whether it is syntactically correct.  
  
The second goal is to produce a complete parse tree, or at least trace the structure of the complete parse tree, for syntactically correct input.
9. Describe the complexity of parsing algorithms.  
the complexity of such algorithms is  $O(n^3)$ , which means the amount of time they take is on the order of the cube of the length of the string to be parsed.

This relatively large amount of time is required because these algorithms frequently must back up and reparse part of the sentence being analyzed.

Reparsing is required when the parser has made a mistake in the parsing process.

Backing up the parser also requires that part of the parse tree being constructed (or its trace) must be dismantled and rebuilt.  $O(n^3)$  algorithms are normally not useful for practical processes, such as syntax analysis for a compiler, because they are far too slow.

10. Describe the recursive-descent parser.

a coded version of a syntax analyzer based directly on the BNF description of the syntax of language.

11. What do the two Ls in LL algorithm specify?

The first L in LL specifies a left-to-right scan of the input;

the second L specifies that a leftmost derivation is generated.

12. State and explain the convention followed for writing a recursive-descent parsing subprogram

Each one leaves the next token of input in nextToken

13. Why are named constants used, rather than numbers, for token codes?

(1) Better code readability

(2) It's better to return a name for the lexeme identified (name has more information than a number)

14. Describe how a recursive-descent parsing subprogram is written for a rule with a single RHS..

(1) For each terminal symbol in the RHS, the terminal symbol should be compared with the "nextToken"

(2) If no matches happen, syntax error reported

(3) If match happens, lexical analyzer called (for "nextToken")

(4) For each non-terminal, the parsing subprogram for that non-terminal is called.

15. Explain the two grammar characteristics that prohibit them from being used as the basis for a top-down parser.

(1) Grammar with direct or indirect left recursion.

(2) Grammar which fails pairwise disjointness test.

16. What is direct left recursion?

The left recursion in the rule  $A \rightarrow A + B$  is called direct left recursion, because it occurs in one rule.

17. Describe the pairwise disjointness test.

A test used to check whether there is a left recursion existing in a grammar or not. If the test is passed, it's non left recursive grammar.

18. What is the limitation of left factoring?

RHS has to fail the pairwise disjointness test

19. What is a phrase of a right sentential form?

The string of terminals collected from the leaves of the partial parse tree.

20. What is the difference between a simple phrase and a phrase of the right sentential form?

Simple phrases can be derived in one step, while phrase can take one or more steps.

21. What is the feature of the handle of a right sentential form?

The handle of any rightmost sentential form is its leftmost simple phrase.

22. What is the mathematical machine on which both top-down and bottom-up parsers are based?

A PDA (PushDown Automation)

23. What is the disadvantage of an LR parser?

it is difficult to produce by hand

the parsing table for a given grammar for a complete programming language.

24. Why is a bottom-up parser often called a shift-reduce algorithm?  
because shift and reduce are the two most common actions they specify. An integral part of

every bottom-up parser is a stack. As with other parsers, the input to a bottom-up parser is the stream of tokens of a program and the output is a sequence of grammar rules.

25. Describe the purpose of a parse stack in an LR parser.

A stack is used to contain a sequence of grammar symbols with a \$ at the bottom of the stack.

26. Describe the properties of the variations on the canonical LR table construction process.

- (1) They require far less computer resources to produce the required parsing table than the canonical LR algorithm, and
- (2) they work on smaller classes of grammars than the canonical LR algorithm.

27. Why is every parser for a programming language a pushdown automaton?

because a PDA is a recognizer for a context-free language.

# HLPL

## Chapter 5 review questions:

1. What is a reserved word?

A special word of a programming language that can- not be used as a name.

2. In Java and C#, how long can a name be?

As long as possible.

3. What is the address of a variable?

the machine memory address with which it is associated.

4. How can a variable be characterized?

as a sextuple of attributes: (name, address, value, type, lifetime, and scope)

5. What is type inference? Give an example.

An implicit type declaration in which context is used to determine the data type. Example:  
`var sum = 0;`

6. What is deallocation of a memory cell?

the process of placing a memory cell that has been unbound from a variable back into the pool of available memory.

7. After language design and implementation, what are the four times bindings can take place in a program?

Before program execution begins, when a variable's declaration statement is being elaborated, when instruction written by the programmer to explicitly allocate memory is reached, and when a variable is assigned values.

8. What is the lifetime of a variable?

the time during which the variable is bound to a specific memory location.

9. What is the use of a stack-dynamic variable?

They allow subprograms to have dynamic local storage so each can have their own version of local variables.

10. What are the advantages and disadvantages of dynamic type binding?

Advantage: more program flexibility. Disadvantages: causes programs to be less reliable because the error-detection capability of the compiler is diminished relative to a compiler for a language with static type bindings and cost of implementation on execution time.

11. Define *static*, *stack-dynamic*, *explicit heap-dynamic*, and *implicit heap-dynamic variables*. What are their advantages and disadvantages?

#### **static variable**

variables that are bound to memory cells before program execution begins and remain bound to those same memory cells until program execution terminates.

#### **stack-dynamic variable**

variables whose storage bindings are created when their declaration statements are elaborated, but whose types are statically bound.

### **explicit heap-dynamic variable**

variables that are nameless (abstract) memory cells that are allocated and deallocated by explicit run-time instructions written by the programmer.

### **Implicit heap-dynamic variables**

Variables bound to heap storage only when they are assigned values. In fact, all their attributes are bound every time they are assigned.

## **12.What is a block-structured language?**

A programming language that permits the creation of blocks, including blocks nested within other blocks

## **13.How is a reference to a nonlocal variable in a static-scoped program connected to its definition?**

Nonlocal variables can be statically determined (prior to execution)

## **14.What is the general problem with static scoping?**

It allows more access to both variables and subprograms than is necessary.

## **15.What is the referencing environment of a statement?**

the collection of all variables that are visible in the statement

## **16.What is a static ancestor of a subprogram? What is a dynamic ancestor of a subprogram?**

The static parent (subprogram that holds the program) of the subprogram, and its static parent, and so forth up to and including the largest enclosing subprogram, are called the static ancestors. Dynamic ancestors are the same thing, but they are parents based on if they call the subprogram.

17.What is a block?

a section of code to have its own local variables whose scope is minimized.

18.What is the purpose of the let constructs in functional languages?

To mimic blocks in imperative languages.

19.What is the difference between the names defined in an ML let construct from the variables declared in a C block?

The names defined in an ML let construct can not have their values changed.

20.Describe the encapsulation of an F# let inside a function and outside all functions.

The scope of a name defined with let inside a function definition is from the end of the defining expression to the end of the function. The scope of let can be limited by indenting the following code, which creates a new local scope.

21.When can we call a subprogram active?

if its execution has begun but has not yet terminated.

22.What is variable initialization?

The binding of a variable to a value at the time it is bound to storage

# HLPL

## Chapter 6 review questions:

1. What is a data type?

It defines a collection of data values and a set of predefined operations on those values.

2. What are the advantages of the data structure of COBOL over the data structure of Fortran I?

The data structures of COBOL took the first step away from the Fortran I model by allowing programmers to specify the accuracy of decimal data values, and also by providing a structured data type for records of information.

3. What is an abstract data type?

a user-defined type.

4. Describe the different uses of the type system of a programming language

The most practical of these is error detection.

The assistance it provides for program modularization.

It defines how a type is associated with each expression in the language and includes its rules for type equivalence and type compatibility.

5. Define *descriptor* and *object*.

A descriptor is the collection of the attributes of a variable. In an implementation, a descriptor is an area of memory that stores the attributes of a variable.

The word object is often associated with the value of a variable and the space it occupies. In this book, however, we reserve object exclusively for instances of user-defined and language-defined abstract data types.

6. What are the two most common structured data types in the imperative languages?

The two most common structured (nonscalar) data types in the imperative languages are arrays and records.

7. What mechanism is used to store negative integers in a computer?

A negative integer could be stored in sign-magnitude notation, in which the sign bit is set to indicate negative and the remainder of the bit string represents the absolute value of the number. Sign-magnitude notation, however, does not lend itself to computer arithmetic.

8. What are the four signed integers supported by Java?

byte, short, int, and long

9. Define *static*, *fixed stack-dynamic*, *fixed heap-dynamic*, and *heap-dynamic arrays*. What are the advantages of each?.

**Static:** subscript ranges are statically bound and storage allocation is static (before run-time)  
Advantage: efficiency (no dynamic allocation).

**fixed stack dynamic:** subscript ranges are statically bound, but the allocation is done at declaration time. Advantage: space efficiency

**fixed heap-dynamic:** similar to fixed stack-dynamic; storage binding is dynamic but fixed after allocation (i.e., binding is done when requested and storage is allocated from heap, not stack).

**heap-dynamic:** binding of subscript ranges and storage allocation is dynamic and can change any number of times. Advantage: flexibility (arrays can grow or shrink during program execution).

10.What is the disadvantage of ones-complement notation?

two representations of zero

11.How are floating-point types represented?

model real numbers; approximations for many real values

12.What are the different representations for floating-point values supported by Java?

float and double

13.What languages support complex data type?

Fortran and Python

14.What is the disadvantage of a decimal type?

range of values is restricted because no exponents are allowed

15.Define *precision* and *range*.

Precision: the accuracy of the fractional part of a value, measured as the number of bits

Range: a combination of the range of fractions and the range of exponents

16.What is the use of a Boolean type?

used to represent switches or flags in programs

17.What are the required entries in a Java array descriptor, and when must they be stored (at compile time or run time)?

In Java all arrays are fixed heap-dynamic arrays. Once created, these arrays keep the same subscript ranges and storage. Secondarily, Java supports jagged arrays and not rectangular arrays. Being a fixed heap-dynamic array the entries will be established and fixed at run time.

18.What is a substring reference?

a reference to a substring of a given string

substring references called slices

19.What two classes support strings in Java?

StringBuffer and StringBuilder

20.Define *limited dynamic length strings* and *dynamic length strings*.

allow strings to have varying length up to a declared and fixed maximum set by the variables definition

21.What is an immutable object of Java's String class?

static length string

22.Define *tuple*.

a data type that is similar to a record, except that the elements are not named

23.What are the design issues for enumeration types?

(1) allowed to appear in more than one type definition?

(2) values coerced to integer?

(3) types coerced to an enumeration type?

24.In what primarily imperative language do lists serve as arrays?

Python and C#

25.What method is used to fetch the internal numeric value of an enumeration variable?

ordinal method

26.What is the difference between C# enumeration types and C++ enumeration types?

In C and C++, an enum defines a value type that represents a restricted set of values. ... In C#, an enum defines a class (i.e. types are classes), meaning it can be declared only at the namespace or class level; it cannot be defined in a statement block within a method as it can in C++

27.In what way does Scheme's CDR function modify its parameter?

returns its parameter list minus its first element

28.What is the mechanics of a list comprehension in Python?

a function is applied to each of the elements of a given array and a new array is constructed from the results

29.Define *union*, *free union*, and *discriminated union*.

**Union:** a type whose variables may store different type values at different types during program execution

**free union:** union without a tag

programmers allowed complete freedom from type checking in their use

30.Are the unions of F# discriminated?

Yes?

31.What are the advantages of the enumeration types of F#?

(1) no arithmetic operations are legal on enumeration types

(2) no enumeration variable can be assigned a value outside its defined range

32.What is the design issue for associative arrays?

form of references to their elements

33.Why are associative arrays called hashes in Perl?

elements are stored and retrieved with hash functions

34.What are generic arrays in Java?

arrays whose elements are references to objects, through their class libraries

35.Why are reference variables in C++ better than pointers for formal parameters?

provide two-way communication between caller function and called function

36.What advantages do Java and C# reference type variables have over the pointers in other languages?

the reference Java and C# provide some of the flexibility and the capabilities of pointers,

37.Describe the lazy and eager approaches to reclaiming garbage.

reclamation is incremental and is done when inaccessible cells are created, and mark-sweep, in which reclamation occurs only when the list of available space becomes empty.

38.Why wouldn't arithmetic on Java and C# references make sense?

Java and C# references refer to class instances and not memory addresses. This immediately prevents arithmetic on these references from being sensible.

39.What is the data type of the subscript of an array in Java?

int

40.Define *record*.

an aggregate of data elements in which the individual elements are identified by names and accessed through offsets from the beginning of the structure

41.Define strongly typed.

type errors are always detected

42.Why is Java not strongly typed?

there are no implicit ways type errors can go undetected

43.What is the syntax of array references?

universal; the array name is followed by the list of subscripts, which is surrounded by either parentheses or brackets

44.What languages have no type coercions?

C and C++

45.Why are the unions in C and C++ called free unions?

programmers are allowed complete freedom type checking in their use

46.What is name type equivalence?

two variables have equivalent types if they are defined either in the same declaration or in declarations that use the same type name

47.What is a dangling pointer?

a pointer that contains the address of a heap-dynamic variable that has been deallocated

48.What is the primary advantage of name type equivalence?

easy to implement

49.What is the primary disadvantage to structure type equivalence?

restrictive

50.For what types does C use structure type equivalence?

C and C++ use structural equivalence except for structs and classes (where name equivalence is used). For arrays, size is ignored.

51.What set operation models C's **struct** data type?

Cartesian (cross) product; union

1. Define operator precedence and operator associativity.

The operator precedence rules for expression evaluation define the order in which "adjacent" operators of different precedence levels are evaluated.

Associativity- when an expression contains two adjacent occurrences of operators with the same level of precedence

2. What is a unary operator? /  
A unary operator has one operand.
3. What is an infix operator? /  
Binary operators

An infix operator, or infix notation of an operator (the notation doesn't change what the operator does) is simply an operator that is written in between the operands

4. What operator usually has left associativity?  
In order to reflect normal usage, addition, subtraction, multiplication, and division operators are usually left-associative
5. When do we call operators "adjacent"?  
If operators are separated by a single operand
6. What associativity rules are used by Java?  
Left to right
7. What is the difference between the way exponentiation operators are implemented in Fortran and Ruby?

Right associative?

8. How do parentheses affect the precedence rule?  
an expression with parentheses has precedence over expressions without parentheses
9. How is a Lisp statement declared?  
subprograms are explicitly called  
ex:  $a + b c \rightarrow (+ a ( b c ))$

10. Give a solution to the problem of operand evaluation order and side effects.

Solution:

1. Write the language definition to disallow functional side effects.
2. Write the language definition to demand that operand evaluation order be fixed

Side effects:

*Functional side effects:* when a function changes a two-way parameter or a non-local variable.

11. What is an overloaded operator?  
multiple use of an operator
12. Define narrowing and widening conversions.  
Narrowing:  
converts a value to a type that cannot store even approximations of all of the values of the original type

Widening:

converts a value to a type that can include at least approximations of all of the values of the original type

13. In JavaScript, what is the difference between == and ===?

loose equality vs strict equality

It is the same but === prevent its operands from being coerced.

14. What is a mixed-mode expression?

An expression that has operands of different types

15. How is referential transparency related to functional side effects?

If you write a function that has side-effects and then substitute the entire function with its return value, any side-effects from executing the function would be lost — a function with side-effects breaks referential transparency and you lose the ability to use that function in any context.

16. What are the advantages of referential transparency?

The semantics of such programs is much easier to understand than the semantics of programs that are not referentially transparent.

17. How does operand evaluation order interact with functional side effects?

If the language does not allow functional side effects then the order of evaluating the operands has no effects on the value of the expression

18. What is short-circuit evaluation?

An expression in which the result is determined without evaluating all of the operands and/or operators

19. Name a language that always does short-circuit evaluation of Boolean

expressions. Name one that never does it.

C, C++, and Java always use short-circuit evaluation

Ada

20. How does C support relational and Boolean expressions?

a shorthand method of specifying a commonly needed form of assignment

21. What is the purpose of a compound assignment operator?

Compound-assignment operators provide a shorter syntax for assigning the result of an arithmetic or bitwise operator. They perform the operation on the two operands before assigning the result to the first operand.

22. What is the associativity of C's unary arithmetic operators?

Operators Associativity is used when two operators of same precedence appear in an expression. Associativity can be either Left to Right or Right to Left. For example: '\*' and '/' have same precedence and their associativity is Left to Right, so the expression "100 / 10 \* 10" is treated as "(100 / 10) \* 10".

23. What is one possible disadvantage of treating the assignment operator as if it were an arithmetic operator?

expression side effect may arise

24. What two languages include multiple assignments?

**Perl, Ruby, and Lua**

25. What mixed-mode assignments are allowed in Java?

All of them; only if the required coercion is widening

26. What mixed-mode assignments are allowed in ML?

None of them.

Because error detection is reduced when mixed-mode expressions are allowed

27. What is a cast?

explicit type conversions

# HLPL

## Chapter 8 review questions:

1. Define selection statement.

Selection statements allow a program to test several conditions, and execute instructions based on which condition is true. That is why selection statements are also referred to as conditional statements

2. Mention one common misuse of the Böhm and Jacopini result.

One obvious misuse of the Böhm and Jacopini result is to argue against the inclusion of any control structures beyond selection and pretest logical loops.

3. What is the general form of a two-way selector?

`if control_expression then clause else clause`

4. What mechanism does Python use to specify compound statements?

Python uses indentation to specify compound statements. For example,

```
if x > y :  
    x = y  
    print "case 1"
```

5. Why is a static semantics rule used instead of a syntactic entity in many languages?

When there is no syntactic indicator to specify a matching of the else clause to one of the then clauses, a static semantics rule, rather than a syntactic entity, is used to provide the disambiguation.

6. Give an example of a language that does not allow else-less **if** statements.

ML

7. Under what circumstances can arithmetic expressions be used as control expressions?

Languages that do not have a Boolean data type (i.e. C89)

8. What is the limitation of building a multiple selector from two-way selectors and gotos?

Resulting structures are cumbersome, unreliable, and difficult to write and read.

9. What are the two general categories of a selection statement?

two-way

n-way

10. Between what two language characteristics is a trade-off made when deciding whether more than one selectable segment is executed in one execution of a multiple selection statement?

In Ada, the choice lists of the case statement must be exhaustive, so that there can be no unrepresented values in the control expression.

In C++, unrepresented values can be caught at run time with the default selector. If there is no default, an unrepresented value causes the whole statement to be skipped.

11.What is the role of the **default** segment in a **switch** statement?

unrepresented values of the control expression

12.What functional languages do not have statements as selectors?

In the functional languages **ML, F#, and LISP**, the selector is not a statement; it is an expression that results in a value.

13.Explain how C#'s switch statement is safer than that of C.

First, C# has a static semantics rule that disallows the implicit execution of more than one segment. The rule is that every selectable segment must end with an explicit unconditional branch statement: either a break, which transfers control out of the switch statement or a goto, which can transfer control to one of the selectable segments (or virtually anywhere else).

The other way C#'s switch differs from that of its predecessors is that the control expression and the case statements can be strings in C#.

14.Under what circumstances can a linear search on a table of cases and labels be accepted?

H   ????

15.What mechanism does a compiler follow when the number of cases in a selection statement is 10 or greater to optimize the time required to execute?

the compiler can build a hash table of the segment labels

16.What is a loop variable? What is stepsize?

loop variable - a counting iterative control statement has a variable in which the count is maintained

stepsize - the difference between sequential loop variable values

17.What is the difference between a pretest version and a posttest version of a logical loop?

Pretest - while; the statement is executed as long as the expression evaluates to true

Posttest - do; the loop body is executed until the expression evaluates to false

18.In C, what is the significance of a **for** loop without the second expression?

the 'for' is terminated

19.What does the **range** function in Python do?

simple counting loops

20.What alternative is provided for goto in Java?

break

21.What are the design issues for logically controlled loop statements?

Should the control be pretest or posttest?

Should the logically controlled loop be a special form of a counting loop or separate statement?

22.What is the main reason user-located loop control statements were invented?

so a programmer can choose a location for loop control other than the top or bottom of a loop body

23.What predefined iterator for standard data structures is available in Perl?

foreach

24.What is the use of a **break** statement in switch statements?

exiting switch statements

25.What are the differences between the **break** statement of C++ and that of Java?

C++: unconditional unlabeled exit

Java: unconditional labeled exit

26.What are fatbars in guarded clauses?

small blocks used to separate the guarded clauses and allow the clauses to be statement sequences

27.What is the limitation of a guarded command?

there is considerably increased complexity in the implementation of the guarded commands over their conventional deterministic counterparts.

28.How does a functional language implement repetition?

recursion

29. How are iterators implemented in Ruby?

using blocks

block: sequence of code, delimited by braces or the 'do' and 'end' reserved words

30. What language predefines iterators that can be explicitly called to iterate over its predefined data structures?

PHP defines current() and next()

31. What common programming language borrows part of its design from Dijkstra's guarded commands?

CSP and Ada

Haskell

1. What are the three general characteristics of subprograms?

- Each subprogram has a single entry point.
- The calling program unit is suspended during the execution of the called subprogram, which implies that there is only one subprogram in execution at any given time.
- Control always returns to the caller when the subprogram execution terminates.

2. What is a subprogram call?

A subprogram call is an explicit request that the called subprogram be executed.

3. What is a subprogram definition?

A subprogram definition is a description of the actions of the subprogram abstraction.

4. What characteristic of Python subprograms sets them apart from those of other languages?

One characteristic of Python functions that sets them apart from the functions of other common programming languages is that function def statements are executable. When a def statement is executed, it assigns the given name to the given function body. Until a function's def has been executed, the function cannot be called. Consider the following skeletal example:

5. How do Ruby methods differ from the subprograms of other programming languages?

Ruby methods are often defined in class definitions but can also be defined outside class definitions, in which case they are considered methods of the root object, Object.

Such methods can be called without an object receiver, as if they were functions in C or C++.

If a Ruby method is called without a receiver, self is assumed. If there is no method by that name in the class, enclosing classes are searched, up to Object, if necessary.

6. What is the feature of Lua functions?

All Lua functions are anonymous (although they can be defined using syntax that makes it appear as though they have names).

7. What are function declarations called in C and C++? Where are the declarations often placed?

They are called prototypes.

Declarations are often placed in header files.

8. Name one pure functional programming language that does not have mutable data.

Haskell

9. What are positional parameters?

Parameters where the correspondence (or binding) between actual and formal parameters is done by position.

10. Give an example of a language that allows positional parameters in addition to keyword parameters.

Python

11. What is the use of a default value in a formal parameter?

They are used if no actual parameter is passed to the formal parameter in the subprogram header.

12. What is the rule of using a default parameter in C++?

The default parameters must appear last because parameters are positionally associated. Once a default parameter is omitted in a call, all remaining formal parameters must have default values.

13. What is the rule for accepting variable parameters in C# methods?

The call can send either an array or a list of expressions, whose values are placed in an array by the compiler and provided to the called method.

14. What language allows array formal parameters?

Ruby

15. What is an ellipsis?

Lua's mechanism for supporting a variable number of parameters.

16. What are the modes, the conceptual models of transfer, the advantages, and the disadvantages of pass-by-value, pass-by-result, pass-by-value result, and pass-by-reference parameter-passing methods?

Pass by Value (In Mode):

Can be implemented by transmitting an access path but not recommended (enforcing write protection is not easy)

Disadvantages (if by physical move): additional storage is required (stored twice) and the actual move can be costly (for large parameters)

Disadvantages (if by access path method): must write-protect in the called subprogram and accesses cost more (indirect addressing)

Pass by Value Result (inout Mode):

A combination of pass by value and pass by result

Sometimes called pass by copy

Formal parameters have local storage

Disadvantages:

Those of pass by result

Those of pass by value

Pass by Result (Out Mode):

When a parameter is passed by result, no value is transmitted to the subprogram; the corresponding formal parameter acts as a local variable; its value is transmitted to caller's actual parameter when control is returned to the caller, by physical move.

It requires extra storage location and copy operation.

Potential problems:

sub(p1, p1); whichever formal parameter is copied back will represent the current value of p1

sub(list[sub] sub); compute address of list[sub] at the beginning of the subprogram or end?

Pass by Reference (Inout Mode):

Pass an access path

Also called pass by sharing

Advantage: passing process is efficient (no copying and no duplicated storage)

Disadvantages

Slower accesses (compared to pass by value) to formal parameters

Potentials for unwanted side effects (collisions)

Unwanted aliases (access broadened)

fun(total, total); fun(list[i], list[j]); fun(list[i], i);

17. Describe the ways that aliases can occur with pass-by-reference parameters.

Pass-by-reference makes

access paths available to the called subprograms, thereby providing access to nonlocal variables.

18. What is the difference between the way original C and C89 deal with an actual parameter whose type is not identical to that of the corresponding formal parameter?

In the original C, neither the number of parameters nor their types were checked. In C89, the formal parameters of functions can be defined in two ways:

They can be defined as in the original C, where the names of the parameters are listed in parentheses and the type declarations for them follow.

Alternatively, C89 also has the "prototype method", in which the formal parameter types are included in the list.

19. Name some languages that support procedures.

## Fortran, Ada

20. Describe the problem of passing multidimensioned arrays as parameters.

It does not allow a programmer to write a function that can accept matrices with different numbers of columns; a new function must be written for every matrix with a different number of columns. This, in effect, disallows writing flexible functions that may be effectively reusable if the functions deal with multidimensional arrays.

21. What is the name of the parameter-passing method used in Ruby?

Pass-by-assignment

22. What are the two issues that arise when subprogram names are parameters?

The first issue that arises is type checking the parameters of the activation of the subprogram that was passed as a parameter.

The second complication appears in languages that allow nested subprograms. There is another issue related to subprogram names that are passed as parameters.

The question is what referencing environment for executing the pass subprogram should be used.

23. Define shallow and deep binding for referencing environments of subprograms that have been passed as parameters.

Shallow binding: The subprogram runs in the environment of the call statement that enacts the passed subprogram

Deep binding: The subprogram runs in the environment of the definition of the passed subprogram

24. What is a generic subprogram?

A subprogram that takes generic parameters that are used in type expressions that describe the types of the parameters of the subprogram.

25. What is ad hoc binding?

Ad hoc binding: The subprogram runs in the environment of the call statement that passed the subprogram as an actual parameter

26. What causes a C++ template function to be instantiated?

All of the methods stored in a delegate instance are called in the order in which they were placed in the instance called a multicast delegate.

27. In what fundamental ways do the generic methods of Java 5.0 differ from those of C# 2005?

Ad hoc polymorphism is an overloaded subprograms provide a particular kind of polymorphism.

28. If a Java 5.0 method returns a generic type, what type of object is actually returned?

The generic type casted to the proper type.

29. If a Java 5.0 generic method is called with three different generic parameters, how many versions of the method will be generated by the compiler?

30. When does a variable have unlimited extent?

when in a program's execution a variable has a (meaningful) value

31. What is subtype polymorphism?

The ability to use a subclass where a superclass is expected

32. What is a multicast delegate?

A Multicast Delegate is a delegate that holds the references of more than one function. When we invoke the multicast delegate, then all the functions which are referenced by the delegate are going to be invoked. If you want to call multiple methods using a delegate then all the method signature should be the same.

33. What is the main drawback of generic functions in F#?

34. What is a coroutine?

Coroutines are methods that essentially break up their execution into multiple parts. Each time you call the coroutine, the next part of the task is performed. They essentially can break in the middle of execution and return to where they left off the next time they are called.

35. What are the language characteristics that make closures useful?

Language in which a subprogram can access variables in nesting scopes and it can be called from anywhere.

36. What languages allow the user to overload operators?

Ada, C++, C#, Ruby, and Python allow operator overloading.

37. What is a symmetric unit control model?

-Symmetric coroutine (unit control model) facilities provide a single control-transfer operation that allows coroutines to explicitly pass control between themselves.-

# Extra from Internet

# C Programming

## Multiple Choice Questions







- 55.) Advantage of UNION over STRUCTURE is \_\_\_\_\_.  
a.) Memory Storage                           c.) Memory Screen  
b.) Memory Location                       d.) None of these
- 56.) Maximum number of elements in the array declaration " int arr[5][8]; " is \_\_\_\_\_.  
a.) 28                                   c.) 35                           d.) 40
- 57.) Array subscripts in C always start at \_\_\_\_\_.  
a.) 1                                   c.) as per programmer                   d.) 0
- 58.) Which is the correct way to declare a pointer?  
a.) int ptr;                           b.) \*int \*ptr;                           c.) int \*ptr;                           d.) \*int ptr;
- 59.) Output of the printf("2.3f\n",17.23478) will be \_\_\_\_\_.  
a.) 17.23478                           c.) 17.2348  
b.) 17.235   d.) 17.23
- 60.) What will be the value of the following:  
1. floor(5.8)  
2. floor(-5.8)  
a.) -5,-6                                   c.) -5,6  
b.) 5, -6   d.) 5,6
- 61.) What would be the value of X after execution of the following statements?  

```
int x,y=10;
char z='a';
x=y+z;
```

a.) Invalid                                   c.) 107  
b.) 17   d.) 10a
- 62.) What will be the output the following:  

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int *i, *j, a=12, b=2, c ;
    c= (a=a+b, b=a/b, a=a*b, b=a-b);
    i= &c;
    printf ("%d",--(*i));
}
```

a.) 93   c.) 91  
b.) 92   d.) 90
- 63.) Study the following program:  

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int a=7,b=5;
    switch(a/a%b)
    {
```

```

        case 1: a=a-b;
        case 2: a=a+b;
        case 3: a=a*b;
        case 4: a=a/b;
        default: a=a;
    }
    printf("%d", a);
}

```

What will be the output?

- a.) 7              b.) 5              c.) 2              d.) None of above
- 64.) A \_\_\_\_\_ operator applies to two operands, while a \_\_\_\_\_ operator applies to a single operand.
- a.) Binary, Unary              c.) Unary, Ternary
  - b.) Ternary, Binary              d.) None of these
- 65.) A \_\_\_\_\_ is a place where we can store values. Size of INT is \_\_\_\_\_ bits.
- a.) int, 2              c.) variable, 16
  - b.) variable, 2              d.) int, 16
- 66.) Variable consist of letters, numbers and \_\_\_\_\_. You can't use \_\_\_\_\_.
- a.) Underscore, keyword              c.) Identifier, keyword
  - b.) Alphabets, underscore              d.) Underscore, identifier
- 67.) Study the following C program:
- ```

#include<stdio.h>
#include<conio.h>
void main()
{
    char *str1,*str2,*str3;
    str1="India is my";
    str2="!!yrttnuoC";
    str3="Gujarat";
    clrscr();
    strncat(str1,strrev(str2),strlen(str3));
    puts(str1);
}

```
- What will be the value of STR1 after execution of the above program?
- a.) India is my Country!!
  - b.) India is my !!yrttnuoC
  - c.) India is my Country
  - d.) None of these
- 68.) Study it:
- ```

#include<stdio.h>
#include<conio.h>
void main()
{
    int count=1,digit=0;
    while(digit<=9)

```

```

    {
        ++count;
        ++digit;
    }
    printf("%d", count);
}
What will be the output?
a.) 10          b.) 9          c.) 11          d.) 12

```

- 69.) What will be the output of following program?

```

#include<stdio.h>
#include<conio.h>
void main()
{
    int i, j, k;
    j=5;
    i= 2*j/2;
    k=2*(j/2);
    printf("i=%d \n k=%d", i, k);
}
a.) i=5, k=5          c.) i=5, k=4
b.) i=4, k=4          d.) i=4, k=5

```

- 70.) Study the following program:

```

#include<stdio.h>
#include<conio.h>
void main()
{
    int a, b, c;
    a=2;
    b=2*(a++);
    c=2*(++a);
    printf("b=%d \n c=%d", b, c);
}
a.) b=4,c=6          c.) b=3, c=6
b.) b=3,c=8          d.) b=4, c=8

```

- 71.) Study the following program:

```

#include<stdio.h>
#include<conio.h>
void main()
{
    int *ptr, i;
    i=12;
    *ptr=i * i;
    ++i;
    printf("%d %d", i, *ptr);
}
a.) 12, 144          b.) 13, 144          c.) 13, 0          d.) None of these

```

- 72.) Study the following program:

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int a=3;
    switch(a)
    {
        case 1:      printf("One");
        case 3:      printf("Three");
        case 5:      printf("Five");
        case 7:      printf("Seven");
        default:     printf("Odd");
        break;    }
}
```

What will be the output of above program?

- a.) Three  
b.) Three Five Odd  
c.) Three Five Seven Odd  
d.) Three Five Seven

- 73.) Study the following program:

```
#include<stdio.h>
#include<conio.h>
union example
{
    int i;
    float f;
    double d;
};
```

```
void main()
{
```

```
union example e;  
printf("\n Size of union is: %d", size);  
}  
What will be the output of this program?
```

What will be the output of this program?  
a) 8      b) 6      c) 10

- a.) 8                  b.) 6                  c.) 10                  d.) 14

- 74.) Study the following program:

```
#include<stdio.h>
#include<conio.h>
void main()
```

```
void main()
{
```

{

```
int i=0, x=0;  
do  
{  
    if(i%5==0)  
    {  
        X++  
    }  
    ++i;  
} while (i < 20);
```

}

- What will be the output of this program?

- 75.) Study the following program:

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int i=0, X=0;
    while (i<20)
    {
        if(i%5==0) {    X+=i; }
        i++;
    }
    printf("X=%d", X);
}
```

What will be the output of above program?



- 76.) Study the following program:

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int i, n=2;
    for(i=0; i<2; i++)
    {
        if(!(i<=n) && (++n==i))
            n=n+2;
        else
            n=n-2;
    }
}
```

What will be the output of this?

- What will be the output of this program?  
a) 3      b) 4      c) 2      d) None of these

- 77.) Symbol of Magnetic Tape is

- a.)  b.)  c.)  d.) 

- 78.) Which is not a symbol of Flow-Chart.?

- a.)  b.)  c.)  d.) None of above

- 79.) Pre-programming technique is.??

- 80.) Which is the step by step representation of program creation?  
a.) Algorithm  
b.) Dry Run  
c.) Flow Chart  
d.) None of Above

81.) Dry Run is a \_\_\_\_\_.  
a.) Tabular Development  
b.) Graphical Representation  
c.) Step by Step Development  
d.) None of Above

82.) 'B' language is developed by \_\_\_\_\_.  
a.) Martin Richards  
b.) Dennis Ritchie  
c.) Ken Thompson  
d.) Berian Kerningham

83.) Which of the following symbol is use for single line comment?  
a.) //  
b.) /\* ..... \*/  
c.) \\  
d.) ;

84.) Which is not the graphical representation of program creation?  
a.) Algorithm  
b.) Dry Run  
c.) Flow Chart  
d.) (A) and (B) both

85.) ANSI C permits the use of as many as \_\_\_\_\_ case labels.  
a.) 258  
b.) 259  
c.) 257  
d.) 256

86.) Structure template is terminated by symbol \_\_\_\_\_.  
a.) Semicolon ( ; )  
b.) Comma ( , )  
c.) Space ( )  
d.) Underscore ( \_ )

87.) ?: is \_\_\_\_\_ operator.  
a.) Logical  
b.) Bitwise  
c.) Assignment  
d.) Conditional

88.) Which symbols used to define an array?  
a.) [ ]  
b.) < >  
c.) { }  
d.) ( )

89.) Find the error:  

```
void main()
{
    int a=11 ;
    if(a>0);
        printf("Its POSITIVE");
    else
        printf("Its NEGATIVE");
}
```

  
a.) getch(); is missing  
b.) statement missing  
c.) misplaced else  
d.) None of above

90.) Which is a relational operator?  
a.) &&  
b.) >=
c.) ?:  
d.) /\* ..... \*/



```
b=a++  
printf("\n\t B: %d", b);  
getch();  
}
```

- a.) Error                  b.) 12                  c.) 11                  d.) None of above

104.)  $a -= b$  is equivalent to \_\_\_\_\_.

- a.)  $a = b - a$                   c.)  $b = b - a$   
b.)  $a = a - b$                   d.)  $b = a - b$

105.) {} are used to group statements together as in a function, or in the body of a loop. Such a grouping is known as \_\_\_\_\_.

- a.) Statement                  c.) Loop  
b.) Block                  d.) Process

106.) Which is the Escape Sequence character?

- a.) \n                  c.) \a  
b.) \o                  d.) (A) and (B) both

107.) Null character is represented by \_\_\_\_\_.

- a.) NULL                  c.) \0  
b.) \o                  d.) All of above

108.) \_\_\_\_\_ Symbol behaves as STYLE while using with one variable and as REMINDER while using with two variable/values.

- a.) &                  b.) %                  c.) \*                  d.) #

109.) \_\_\_\_\_ and \_\_\_\_\_ are used to test more than one condition.

- a.) &&, ||                  b.) <=, >=                  c.) ==, !=                  d.) <, >

**God Bless You**

# **Fundamentals of Computer Programming-I**

## **Multiple Choice Questions**

### **Question Bank**

# PART A

## UNIT -I Introduction to Computers

**Q.1 ALU stands for**

- (a) Arithmetic Logic Unit
- (b) Array Logic Unit
- (c) Application Logic Unit
- (d) None of above

**Q.2 The brain of any computer system is**

- (a) ALU
- (b) Memory
- (c) CPU
- (d) Control Unit
- (d) None of above

**Q.3 What difference does the 5<sup>th</sup> generation computer have from other generation computers?**

- (a) Technological advancement
- (b) Scientific code
- (c) Object Oriented Programming
- (d) All of the above
- (e) None of the above

**Q.4 Which of the following computer generation uses concept of artificial intelligence?**

- (a) First Generation
- (b) Second Generation
- (c) Third Generation
- (d) Forth Generation

**Q.5 When a key is pressed on keyboard, which standard is used for converting the keystroke into the corresponding bits**

- (a)ANSI
- (b) ASCII
- (c) EBCDIC
- (d) ISO

**Q.6 Which device is used as the standard pointing device in a Graphical User Environment**

- (a) Keyboard
- (b) Mouse
- (c) Joystick
- (d) trackball

**Q.7 Which of the following is valid storage type?**

- (a) CPU
- (b) Keyboard
- (c) Pen Drive
- (d) Track Ball
- (e) None of the above

**Q.8 The section of the CPU that is responsible for performing mathematical operations**

- (a)Memory
- (b) Register Unit
- (c)Control Unit
- (d)ALU
- (e) None of the above

**Q.9 Any storage device added to computer beyond the immediately usable main storage is known as:**

- (a)Floppy disk
- (b)Hard Disk
- (c)Backing store
- (d) Punched Card
- (e) None of the above

**Q.10 The list of coded instructions is called**

- (a)Computer Program
- (b)Algorithm
- (c) Flowchart
- (d)Utility Program
- (e) None of the above

**Q.11 Source code is available to view, modify and redistribute in**

- (a)Open Source
- (b) Closed
- (c)Proprietary
- (d) Licensed
- (e) None of the above

Source

**Q.12 Which of the following is not input device**

- (a) Touch Pad
- (b) Mouse
- (c)Printer
- (d) Joystick

**Q.13 MS Word is example of Closed Source Software**

- (a) True
- (b) False

**Q.14 Software required to run the hardware is known as**

- (a)Task Manager
- (b) Task Bar
- (c) Program Manager
- (d) Device Driver

**Q.15 Which the following is application software?**

- (a)Compiler
- (b)Power Point
- (c)Debugger
- (d) None of the above

**Q.16 Which of the following is system software?**

- (a)Linux
- (b)Word
- (c)Excel
- (d)Tally

**Q.17 The Programs which are as permanent as hardware and stored in ROM is known as**

- (a)Hardware
- (b)Software
- (c)Firmware
- (d)ROMware

**Q.18 Which of the following is invalid type of memory**

- (a)RAM (Random Access Memory)
- (b)ROM (Read Only Memory)
- (c)PRAM (Programmable Read Access Only Memory )
- (d)EPROM (Erasable Programmable Read Only

**Q 19 =C"s were used in \_\_\_\_\_ of computers**

(a) A First Generation      (b) Second Generation      (c) Third Generation      (d) Fifth Generation

**Q.20 which of the following is not component of computer system?**

- (a) Input Device      (b) Stepper Motor      (c) Memory      (d) None of the above

**Q.21 Which of the following is not output device?**

- (a) Printer      (b) VDU      (c) Scanner      (d) All

**Q.22 Joystick is used for \_\_\_\_\_**

- (a) Gaming      (b) Weather forecast      (c) Word Processing      (d) All

**Q.23 Trackball is output device**

- (a) True      (b) False

**Q.24 ALU is part of Memory**

- (a) True      (b) False

**Q.25 CPU consist of**

- (a) ALU+CU      (b) ROM+ALU      (c) RAM+ROM      (d) None

**Q.26 \_\_\_\_\_ is Volatile Memory**

- (a) ROM      (b) EPROM      (c) RAM      (d) None

**Q.27 \_\_\_\_\_ is non volatile memory**

- (a) RAM      (b) EERAM      (c) ROM      (d) PROM

**Q.28 Which of the following is the Valid Measurement unit of memory**

- (a) GB      (b) MB      (c) KB      (d) All

**Q.29 Hardware can work without device driver**

- (a) True      (b) False

**Q.30 Which of the following if not OS**

- (a) Android      (b) MAC      (c) Samsung      (d) LINUX

**Q.31 DVD is having more storage capacity than CD**

- (a) True      (b) False

**Q.32 Tally is \_\_\_\_\_**

- (a) Open S/W      (b) Closed S/W      (c) Application s/w      (d) System s/w

**Q.33 VLSI is used in \_\_\_\_\_ Generation of computer**

- (a) First      (b) Second      (c) Third      (d) Fourth

**Q.34 \_\_\_\_\_ Interprets and executes set of instruction**

- (a) CPU      (b) VDU      (c) Printer      (d) Scanner

**Q.35 \_\_\_\_\_ consists of set of instruction**

- (a) Software      (b) Hardware      (c) Program      (d) None of this

**Q.36 \_\_\_\_\_ consists of set of program s**

- (a) Scanner      (b) VDU      (c) Software      (d) None of this

**Q.37 Paint brush is text editor**

- (a) True      (b) False

**Q.38 Notepad is text editor**

- (a) True      (b) False

**Q.39 Keyboard Converts typed in character to**

- \_\_\_\_\_ code  
(a) EBCDIC      (b) ASCII      (c) Decimal      (d) Binary

**Q.40 \_\_\_\_\_ Unit control the operation of CPU**

- (a) ALU      (b) RAM      (c) CU      (d) BU

**Q.41 Which one of the following is not a feature of third generation languages?**

- (a) They need to get translated      (b) They are faster than MLLs  
(c) They are easy to use than MMLs      (d) They use compilers and interpreters

**Q.42 Which is the type of memory for information that does not change on your computer?**

- (a) RAM      (b) RAM      (c) ERAM      (d) RW/RAM

**Q.43 LSI,VLSI & ULSI chips were used in which generation?**

(a)First      (b)Second      (c)Third      (d)Fourth

**Q.44 Which characteristics of computer distinguishes it from electronic calculation?**

(a)Accuracy      (b)Storage      (c)Versatility      (d)Automatic

**Q.45 Which of the following is not the classification of computers based on application?**

(a) Electronic Computers      (b)Analog Computers  
(c)Digital Computers      (d)Hybrid Computers

## Answer Keys

QUE NO	ANS								
1	A	2	C	3	D	4	D	5	B
6	B	7	C	8	D	9	A	10	A
11	A	12	C	13	C	14	D	15	B
16	A	17	C	18	C	19	B	20	B
21	C	22	A	23	B	24	B	25	A
26	C	27	C	28	D	29	B	30	C
31	A	32	C	33	D	34	A	35	C
36	C	37	B	38	A	39	B	40	C
41	B	42	D	43	C	44	B	45	B

## UNIT –II Introduction to Open Source Operating Systems

**Q.1 Source code is not available for user in**

(a)Open Source      (b)Bharat OS      (c)Linux OS      (d)None

**Q.2 Linux is closed source**

(a)True      (b)False

**Q.3 Bash is the \_\_\_\_\_**

(a)Shell      (b)Compiler      (c)None

**Q.4 BOSS is developed by**

(a) NASA      (b)IUCCA      (c)C-DAC      (d)NASDAQ

**Q.5 BOSS is developed over**

(a)Mandrake      (b)SUSE      (c)Fedora      (d)Debian

**Q.6 Android is desktop OS**

(a)True      (b)False

**Q.7 3DBlender is close source**

(a)True      (b)False

**Q.8 Mozilla Firefox is \_\_\_\_\_**

(a)Browser      (b)Editor      (c)Compiler      (d)None of Above

**Q.9 Google chrome is \_\_\_\_\_**

(a)Compiler      (b)OS      (c)Editor      (d) None of Above

**Q.10 Windows OS is a closed source**

(a)True      (b)False

**Q.11 Microsoft .NET is \_\_\_\_\_**

(a)Open Source      (b)Closed Source      (c)Browser      (d)All of above



[Type text]

# Answer Key

QUE NO	ANS										
1	B	7	B	13	D	19	C	25	B	31	A
2	B	8	A	14	A	20	A	26	A	32	B
3	A	9	D	15	B	21	A	27	B	33	D
4	C	10	A	16	B	22	B	28	D		
5	D	11	B	17	B	23	D	29	A		
6	B	12	B	18	C	24	A	30	A		

## **UNIT –I Part –III Eclipse**

**Q.1**Eclipse is an IDE developed in \_\_\_\_\_.



**Q.2. IDE consists of:**






(a) true  
Q4. Ellipse is an

- Q.4. Eclipse is an \_\_\_\_\_.**

(a) compiler      (b) Debugger      (c) IDE      (d) Interpreter



(d) true



Q.7.CDT does not provide content assistant provider.



#### **Q.8.Eclipse supports programming in PHP.**



**Q.9.Eclipse supports programming in Ruby.**



**Q.10.Eclipse supports programming in Perl.**

- (a) true

## Answer Key

- (b) false

QUE	ANS								
NO									
1	C	3	B	5	A	7	B	9	A
2	B	4	C	6	B	8	A	10	A

## **Unit –I Part IV Programming Languages**

**Q.1 What is the older high-level(non-assembler) programming language?**

- (a)C      (b)Lisp      (c)Fortran      (d)Basic

**O.2 The primary progenitor of COBOL was the programming language known as:**

[Type text]

**Q.3 The C++ programming language is very popular because it is:**

- (a) backward compatible with C.      (b) object-oriented.  
(C) widely available.      (d) all of the above.

**Q.4** Java and perl are examples of \_\_\_\_\_ languages.

- (a) compiled      (b) Interpreted      (c) Hybird      (d) script

**Q.5** What innovation introduced in ALGOL68 is often credited to pascal?



**Q.6 The most common programming languages ,dating back to the 1940s are called:**

- (a)functional
  - (b)object-oriented
  - (c)rule-based.
  - (d)imperative.

**Q.7 When did the first ANSI programming language standard come out?**

- (a) 1949      (b) 1975      (c) 1958      (d) 1966

#### **Q.8 List in chronological order, when these languages become officially recognized as a standard**

- (i)ANSI C                   (ii)ANSI COMMON LISP                   (iii)ANSI Cobol                   (iv)ANSI ADA  
 (a)i. ii. iii. iv           (b)i. iii. ii. iv                   (c)iv. iii. i. ii                   (d)i. iv. iii. ii

### **O.9 Who uses languages which are**

standards?



**Q-10 What features make C++ so powerful?**

- Q.16** What features make C++ so powerful?

  - (a) Easy Implementations
  - (b) Reusing the old code
  - (c) Easy Memory Management
  - (d) All of the above

(c) Easy Memory Management      (d) None of the above  
Q. 11 What makes OOP so popular?

- (i) Data abstraction
  - (ii) Easily reusable
  - (iii) Easily modifiable
  - (iv) None of the above

(a) i only

(b) ii & iii only

(c) i, ii & iii only

(d) iv only

**Q. 12 What is/are the main differences between the 3rd and 4th generation languages?**

- Q. 12 What is/are the main differences between the 3rd and 4th generation languages?

  - (i) Both follow procedural code.
  - (ii) Third generation *languages* are mostly compiled languages.
  - (iii) Fourth generation languages are *in-line* with the minimum work and skill concept.
  - (iv) Third generation languages are user friendly and have intelligent default options.

(a) i & iv only	(b) ii & iii only
(c) c. i & iii only	(d) d. none of the above

O. 13 Which of the features below would make the next generation of PL popular?

- Q. 10 Which of the features below would make the next generation of PL

  - (i) They are highly portable and are offered on a wide range of systems .
  - (ii) They are suitable for development of programs of arbitrary size and complexity.
  - (iii) They are reasonably stable during changes in hardware and system software.
  - (iv) They both have procedural and non-procedural features.

(a) i & ii only.	(b) i, iii & iv
(c) iii & iv only	(d) All of the above

[Type text]

[Type text]

**Q. 15 Which of the following languages is the newest?**



**Q. 16 Which of the following languages is the oldest?**



**Q. 17 Which of the following languages introduced the notion of inheritance?**



**Q. 18 Which of the following are language processors?**

- (a) Assembler      (b) Compiler      (c) Interpreter      (d) All of the above

**Q. 19 A program in execution is called**

- (a) process      (b) function      (c) CPU      (d) Memory

**Q. 20 An assembly language is a**

- (a) low level programming language      (b) Middle level programming language  
(c) High level programming language    (d) Internet based programming language

**Q. 21 An assembler is**

- (a) Programming language dependent.      (b) Syntax dependant.  
(c) Machine dependant.      (d) Data dependant

**Q. 22** Translator for low level programming language were termed as



**Q. 23 What is the name of the category of programming languages whose structure is dictated by the von Neuman computer architecture?**



**Q. 24** A paradigm that allows specification of what has to be computed rather than just how a computation is to be carried out.

- (a) Imperative
  - (b) Denotational
  - (c) Functional
  - (d) Non-procedural
  - (e) Constraint
  - (f) Object-oriented

**Q. 25 A paradigm incorporating encapsulation, Inheritance and dynamic type binding.**

- (a) Imperative
  - (b) Denotational
  - (c) Functional
  - (d) Non-procedural
  - (e) Constraint
  - (f) Object-oriented

**Q. 26 Which language is considered to be the first fully object-oriented language?**

- |             |               |
|-------------|---------------|
| (a) FORTRAN | (b) COBOL     |
| (c) LISP    | (d) C         |
| (e) JAVA    | (f) SMALLTALK |

### **Q.27 In what language is UNIX written?**

- |             |               |
|-------------|---------------|
| (a) FORTRAN | (b) COBOL     |
| (c) LISP    | (d) C         |
| (e) JAVA    | (f) SMALLTALK |

**Q. 28 What programming language has dominated scientific computing over the past 35 years?**

- |             |                |
|-------------|----------------|
| (a) FORTRAN | (b) COBOL      |
| (c) LISP    | (d) C          |
| (e) JAVA    | (f) SMALL TALK |

**Q. 29 What programming language has dominated artificial intelligence programming over the past 35 years?**



[Type text]

[Type text]

(  
b  
)

C  
O  
B  
O  
L

(  
d  
)

C  
(f) SMALL TALK

[Type text]





[Type text]

**Q.61 The tool used by a programmer to convert a source program to a machine language object module is a**



## ANSWER KEY

## **Unit -I Part -IV Documentations**

**Q.1** is part of agreement between customer and company which describes needs of the customer

- (a)cost Estimate      (b)Requirement document  
(c)patent                (d)Need document

### **Q.2 Product brief is for**

- (a)Users                    (b)Coders                    (c)Mangers                    (d)Marketing & sales people

### **Q.3 Technical documentation is prepared by**

- Q.5 Technical documentation is prepared by  
(a) Users                    (b) Coders                    (c) Managers                    (d) Marketing & sales people

**Q 4 Cost estimate is part of agreement between customer and company which describes**



**Q.5** User manual of software can be found in the menu option \_\_\_\_\_.



**Q6** Quality of software is documented in \_\_\_\_\_ phase of software development.

- Q.6 Quality of software is documented in \_\_\_\_\_ phase of software development.**

- Q.7 Inventor is given special Rights using document \_\_\_\_\_



(c)Architecture Document



(c) Letter typing

- Q.10 Latex automatically generates \_\_\_\_\_

(a)Indexes (b)Bibliography

- Q.11 First command in LATEX for any document is \_\_\_\_\_**

(a)\begin{document} (b)\documentclass{article} (c)\begin{document}



[Type text]

This is example for creating new paragraph in latex. It simply requires to add an extra newline. Because of this simple way the writer does have to worry about indentation.

This is now bit complex. I have left an extra line, you can see the effect in output for the same. try to read the section properly in the chapter, you will get the clue. This will again make you think.

#### ANSWER KEY

<u>QUE NO</u>	<u>ANS</u>						
1	B	6	A	11	B	16	B
2	D	7	A	12	D	17	C
3	B	8	C	13	D	18	D
4	C	9	A	14	A	19	C
5	D	10	C	15	D	20	A
21	C	22	C				

## Unit II- Algorithm & Programming Concepts

**Q.1 Macro flowchart is also called as**

- |                              |                          |
|------------------------------|--------------------------|
| (a)Simple detailed flowchart | (b)Less Detail flowchart |
| (c)More detail flowchart     | (b)None                  |

**Q.2 GUI stands for**

- |                             |                             |
|-----------------------------|-----------------------------|
| (a)Graphical User Interface | (b) Graph Under Instruction |
| (c)Graphical input Unit     | (d)None                     |

**Q.3 Terminal symbol in a flowchart indicates**

- |                     |               |
|---------------------|---------------|
| (a)End              | (b)processing |
| (c)Input and Output | (d)Decision   |

**Q.4 Continue Statement**

- |   |  |
|---|--|
| (a)Without Executing remaining statements takes control back to starting loop |  |
| (b)Take control outside the loop  |  |
| (c)Continues to program end   |  |
| (d)None   |  |

**Q.5 Structured Programming is**

- |  |                                    |
|--|------------------------------------|
| (a)Dividing the program into different program modules | (b)Using Structures in the program |
| (c)Using classes in the program                        | (d)None                            |

**Q.6 Pseudocode is used for**

- |                              |                             |
|------------------------------|-----------------------------|
| (a)Denoting the program Flow | (b) To make structure chart |
| (c) For coding the program   | (d)To write program steps   |

**Q.7 Macro flowchart shows the**

- |                           |                   |
|---------------------------|-------------------|
| (a)Outline of the program | (b)Program code   |
| (c)Program Detail         | (d)Both (a) & (c) |

**Q.8 Indentation in a program**

- |  |                   |
|--|-------------------|
| (a) Improves its readability and understanding | (b) Is compulsory |
| (c)Both  | (d)None           |

**Q.9 Which of the following is used to avoid infinite loops?**

- |             |        |          |             |
|-------------|--------|----------|-------------|
| (a)Sentinel | (b)For | (c)While | (d)Do while |
|-------------|--------|----------|-------------|

**Q.10 Which of the following is not necessarily a characteristic of a program module?**

- |                              |                                    |
|------------------------------|------------------------------------|
| (a)It performs a single task | (b)It contains Several sub modules |
| (c)It is self-contained      | (d)It is relatively small in size  |

**Q.11 Which of the following is not a benefit of modular programming?**

- |  |   |
|--|---|
| (a)It increases program readability  | (b)It increases programmer productivity |
| (c)It allows for the creation of a library of common programming task        |   |
| (d)It allows one programmer to do the job of many in the same amount of time |   |

**Q.12 The main module of a program contains the following sequences of statements**

Call Module A

---

Call Module B

---

Call Module C

---

Which of the following statements is executed after Call Module B?

- |                                    |                  |
|------------------------------------|------------------|
| (a)Call Module A                   | (b)Call Module C |
| (c)The first Statement in Module B | (d)None          |

[Type text]

**Q.13 Which of the following statements is executed after all statements in ModuleB have been carried out in above Q 12?**

- (a)Call Module A
- (b)Call Module C
- (c)The first statement in Module C
- (d)None

**Q.14 Which of the following is not a principle of structured programming?**

- (a)Design the program in top-down manner
- (b)Write each program module as a series of control structures
- (c)Code the program so that it runs correctly without testing
- (d)Use good programming

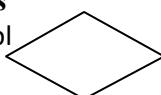
**Q.15 The flowchart symbol bellow**

- (a)Process symbol
- (b)Input/output symbol
- (c)Decision symbol
- (d)Terminator symbol



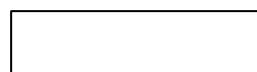
**Q.16 The flowchart symbol shown below is**

- (a)Process symbol
- (b)Input/output symbol
- (c)Decision symbol
- (d)Terminator symbol



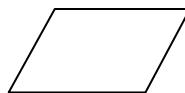
**Q.17 The flowchart symbol shown below is**

- (a)Process symbol
- (b)Input/output symbol
- (c)Decision symbol
- (d)Terminator symbol



**Q.18 The flowchart symbol shown below is**

- (a)Process symbol
- (b)Input/output symbol
- (c)Decision symbol
- (d)Terminator symbol



**Q.19 Which of the following is not a basic control structure?**

- (a)The process
- (b)The Loop
- (c)The decision
- (d)The sequential

**Q.20 Which of the following is not a principle of good programming style?**

- (a)Use descriptive variable names
- (b)Provide a welcome message
- (c)Identify using text the numbers that are output
- (d)Test the program

**Q.21 Method which uses a list of well defined instructions to complete a task starting from a given initial state from a given initial state to end state is calls as**

- (a)Program
- (b)Flowchart
- (c)Algorithm
- (d)A & B

**Q.22 The chart that contains only function flow and no code is called as**

- (a)flowchart
- (b)Structure chart
- (c)Both A and B
- (d)None

**Q.23 Which of the following is a program planning tool?**

- (a)Sequential
- (b)decision
- (c)Pseudo code
- (d)Both B and C

**Q.24 Which of the following structures are used in computer programs?**

- (a)sequential
- (b)decision
- (c)Timesharing
- (d)None

**Q.25 Execution of two or more programs by a single CPU is known as**

- (a)Multiprogramming
- (b)Multiprocessing
- (c)Timesharing
- (d)None

**Q.26 A structured chart is**

- (a)A statement of information processing requirements
- (b)A document of what has to be accomplished
- (c)A hierarchical Partitioning of the program
- (d)Beginners all purpose
- (e)All

**Q.27 In structure charts modules are described as**

- (a)Circle
- (b) Triangles
- (c)Rectangle
- (d)Ellipse

**Q.28 The sequence logic will not be used while**

- (a)Accepting input from user
- (b)Comparing two sets of data
- (c)Giving output to the user
- (d)Adding two numbers

**Q.29 Flowcharts and Algorithms are used for**

- (a) Better Programming
- (b) Efficient Coding
- (c) Easy testing and Debugging
- (d) All

**Q.30 An Algorithm represented in the form of programming languages is \_\_\_\_\_**

- (a) Flowchart
- (b) Pseudo code
- (c) Program
- (d) None

**Q.31 Which of the following is a pictorial representation of an algorithm?**

- (a) Pseudo code
- (b) Program
- (c) Flowchart
- (d) Algorithm

**Q.32 Which of the following symbol in a flowchart are used to indicate all arithmetic processes of adding, subtracting, multiplying and dividing ?**

- (a) Input/output
- (b) terminal
- (c) Processing
- (d) Decision

**Q.33 A flowchart that outlines the main segments of program is called as**

- (a) Micro flowchart
- (b) Macro flowchart
- (c) Flowchart
- (d) Algorithm

**Q.34 A flowchart that outlines with all detail is called as**

- (a) Micro flowchart
- (b) Macro flowchart
- (c) Flowchart
- (d) Algorithm

**Q.35 Pseudo code is also known as**

- (a) Program Design Language
- (b) Software Language
- (c) Hardware Language
- (d) Algorithm

**Q.36 Pseudo code emphasizes on**

- (a) Development
- (b) Coding
- (c) Design
- (d) Debugging

**Q.37 In which of the following pseudo code instructions are written in the order or sequence in which they are to be performed?**

- (a) Selection Logic
- (b) Sequence Logic
- (c) Iteration Logic
- (d) Looping Logic

**Q.38 Which of the following logic is used to produce loops in program logic when one or more instruction may be executed several times depending on some conditions?**

- (a) Iteration Logic
- (b) Selection Logic
- (c) Sequence Logic
- (d) Decision Logic

**Q.39 Selection logic also called as**

- (a) Decision Logic
- (b) Iteration Logic
- (c) Sequence Logic
- (d) Looping Logic

**Q.40 Which of the following program planning tool allows the programmers to plan program logic by writing program instruction in an ordinary language?**

- (a) Flowchart
- (b) Pseudo code
- (c) Program
- (d) Looping

**Q.41 Which logic is used to select the proper path out of two or more alternative paths in program logic**

- (a) Looping Logic
- (b) Sequence Logic
- (c) Iteration Logic
- (d) Selection Logic

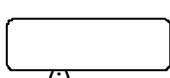
**Q.42 Which of the following control structures are used in iteration logic**

- (a) if then if then else
- (b) do which
- (c) do which repeat until
- (d) do while if else

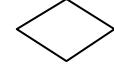
**Q.43 To write the correct and effective program we must first**

- (a) Draw a flowchart
- (b) Plan its logic
- (c) Write pseudo code
- (d) Use iterations

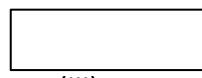
**Q.44 Match the following**



(i)



(ii)



(iii)



(iv)



(v)

- (a) Connecting
- (b) Input/Output
- (c) Processing
- (d) Terminal
- (e) Decision

**ANS=i-(d),ii-(e),iii-(c),iv-(a);**

**Q.45 Which of the following file contains the programmer's original program code?**

- (a) Application file
- (b) Executing
- (c) Object file
- (d) Source file

**Q.46 Algorithm is**

- (a)step by step execution of program      (b)Executable file  
(c)Object file      (d)Source file

**Q.47 Kite box in flow chart is used for**

- (a)Connector      (b)Decision      (c)Statement      (d) All of the above

**Q.48 Which of the following is not a characteristic of good algorithm?**

- (a)Precise      (b)Finite number of steps  
(c)Ambiguous      (d)Logical flow of control

**Q.49 Diagrammatic representation of an algorithm is**

- (a)Flowchart      (b)Data flow Diagram      (c)Algorithm design      (d) Pseudo code

**Q.50 Goto statement is ?**

- (a)Used to jump the control of program      (b)Same as switch case statement  
(c)Used for user defined iteration      (d)None of above

**Q.51 After a programmer plans the logic of a program ,she /he will next \_\_\_\_\_**

- (a)Understand the problem      (b)Test the program  
(c)Translate the program      (d)Code the program

**Q.52 What symbol is used to represent output in a flowchart?**

- (a)Square      (b)Circle      (c)Parallelogram      (d)Triangle

**Q.53 What is the standard terminal symbol for flowchart?**

- (a)Circle      (b)Parallelogram      (c)Diamond      (d)Square

**Q.54 The following pseudo code is an example of \_\_\_\_\_ structure:**

Get number

While number is positive

Add to sum

- (a)Sequence      (b)Decision      (c)Loop      (d)Nested

**Q.55 The following pseudo code is an example of \_\_\_\_\_ structure:**

Get number

Get another number

If first number is greater than second then

Print first number

Else

print second number

- (a)Sequence      (b)Decision      (c)Loop      (d)Nested

**Q.56The following pseudo code is an example of \_\_\_\_\_ structure:**

Get number

Get another number

Multiply numbers

Print result

- (a)Sequence      (b)Decision      (c)Loop      (d)Nested

**Q.57structured program can be easily broken down into routines or \_\_\_\_\_ that can be assigned to any number of programmers**

- (a)Segments      (b)Modules      (c)Units      (d)Sequences

**Q.58 In a case structure of the loop, the loop body continues to execute as long as the answer to the controlling question is yes, or true.**

- (a)Else      (b)Then      (c)Default      (d)Loop

**Q.59 In which of the following loop ,the loop body continues to executes as long as the answer to the controlling question is yes, or true.**

- (a)do-then      (b)do-when      (c)do-until      (d)do-while

**Q.60 Which of the following statement cause program control to end up almost anywhere in the program?**

- (a)go to      (b)for      (c)while      (d)do while

**Q.61 Which of the following statement allows us to make a decision from the number of choices?**

- (a)break      (b)Switch      (c)for      (d)go to

**Q.62 Which of the following keyword is followed by an integer or character constant?**

- (a)switch      (b)case      (c)for      (d)void

**Q.63 Which of the following enhances the versatility of the computer to perform a set of instructions repeatedly?**

- (a)Function      (b)Loop      (c)header files      (d)statement

**Q.64 Which of the following contains parenthesis after the „while“ loop?**

- (a)Condition      (b)statement      (c)count      (d)value

**Q.65 The condition being tested within the loop may be relational or relational or logical operations**

- (a)while      (b)switch      (c)break      (d)continue

**Q.66 Which of the following loop uses three things initialization, condition to terminate loop and increasing the value of loop counter?**

- (a)for      (b)while      (c)goto      (d)switch

**Q.67 The three things inside the for loop are separated by**

- (a)colon      (b)comma      (c)semicolon      (d)hyphen

**Q.68 Which of the following statement associated with an „if“?**

- (a)switch      (b)goto      (c)break      (d)do while

**Q.69 „do while“ loop is useful when we want that statement within the loop must be executed**

- (a)Only Once      (b)At least Once      (c)More than once      (d)None of above

**Q.70 Which of the following statement allows the programmer to make the control to the beginning of the loop ,without executing the statement inside the loop?**

- (a)while      (b)continue      (c)go to      (d)if

**Q.71 Which of the following can be replaced by if**

- (a)switch      (b)while      (c)continue      (d)for

**Q.72 Which of the following statement is useful while writing menu driven programs**

- (a)while      (b)break      (c)switch      (d)if

**Q.73 Which of the following is self contained block of statements that perform a coherent task of some kind?**

- (a)function      (b)loop      (c)statement      (d)body of program

**Q.74 The function gets called when the function name is followed by**

- (a)colon      (b)semicolon      (c)statement      (c)braket

**Q.75 The mechanism used to convey information to the function is the**

- (a)Argument      (b)commands      (c)loops      (d)statements

**ANSWER KEY :**

QUE NO	ANS								
1	B	16	C	31	C	46	D	61	B
2	A	17	A	32	C	47	B	62	B
3	A	18	B	33	B	48	C	63	B
4	A	19	A	34	A	49	A	64	A
5	A	20	B	35	A	50	A	65	A
6	D	21	C	36	C	51	D	66	A
7	A	22	B	37	B	52	C	67	C
8	A	23	D	38	A	53	A	68	C
9	A	24	D	39	A	54	C	69	B

[Type text]

10	D	25	B	40	B	55	B	70	B
11	D	26	C	41	D	56	A	71	A
12	C	27	B	42	C	57	B	72	A
13	B	28	B	43	B	58	A	73	A
14	A	29	D	44		59	D	74	B
15	D	30	C	45	D	60	A	75	A

**UNIT –III Introduction to C**

**For Programs Students are Instructed to follow the following :**

***Consider every program has a main()***

***Consider Void→ void, Main→main, Printf→printf, Scanf→ scanf, Int→ int , Float→float***

**Q.1 Which of the following is not a type of computer programming language?**

- (a) Natural language      (b)Machine Language      (c)High-level language      (d)Binary languages

**Q.2 The programming language that closely resembles the machine language is**

- (a)High-level languages      (b)C language      (c)FORTRAN      (d)Assembly language

**Q.3 The tool used to convert a „C” program to machine language is called as**

- (a)Linker      (b)Language translator      (c)Compiler      (d)Preprocessor

**Q.4 The programmer original program code is called as**

- (a)Object file      (b)Source file      (c)Executable file      (d)Application file

**Q.5 The diagrammatic flow of the program is represented by**

- (a)flowchart      (b)Program map      (c)Pseudo code      (d)Water fall mode

**Q.6 C- language**

**is**

- (a)Assembly level Language      (b)Low level Language  
(c)High level Language      (d)All of above

**Q.7 What is a program**

- (a)A set of instruction      (b)A set of algorithm      (c)A set of pseudo code      (d)All of above

**Q.8 Who developed the C language**

- (a)Dennis Ritchie      (b)Ken Thompson      (c)Matrin Richards      (d)Patric Naughton

**Q.9 Which year was C developed in?**

- (a)1975      (b)1980      (c)1972      (d)1971

**Q.10 The C language has been developed at**

- (a)AT & T Bell Labs      (b)IBM      (c)Borland International      (d)Sun Microsystems

**Q.11 The C programs are stored with \_\_\_\_\_ extension**

- (a).obj      (b).bak      (c).c      (d).cpp

**Q.12 Every statement in C program is to be terminated by a \_\_\_\_\_**

- (a).dot(.)      (b).semi-colon(;)      (c).colon(:)      (d).Question mark(?)

**Q.13 The escape sequence „\b” is a**

- (a)back space      (b)next line      (c)tab      (d)none of the above

**Q.14 Which OS (Operating System) supports C?**

- (a)DOS only      (b)Linux only      (c)window only      (d>All of the above

**Q.15 The real numbers (numbers with decimal fractional value) in C can be expressed which of the following forms?**

- (a)Fractional form only      (b)ASCII  
(c)Exponent form only      (d)Both fractional and Exponetial

**Q.16 A character variable can store how many characters at a time?**

- (a) 1 character      (b)8 characters      (c)255 character      (d)None

**Q.17 What will be stored in the variable „ch” if we write the statement char ch="z"?**

- (a)ASCII value of Z      (b)Z along with the single inverted commas  
(c)The character Z      (d)None of above

**Q.18 What is the maximum value that an signed integer constant can have?**

- (a)32768      (b)32767      (c)1.7014e+38      (d)256

**Q.19 An identifier in C cannot start with?**

- (a)A number      (b)An Alphabet





- (a)Modem      (b)Monitor      (c)Disk      (d)Printer

**Q.45**The default standard input device for C++ program is

- (a)Mouse      (b)Scanner      (c)Keyboard      (d)None of above

**Q.46**When requesting multiple inputs from the user,they must be separated by

- (a)a space      (b)a tab character  
(c)a new line character      (d)any of the above

**Q 47** The “return 0 ”statement in main function indicates

- (a)The program did nothing i.e. completed zero tasks  
(b)The program will be executed without any error  
(c)The program has not yet completed the execution  
(d)None of the above

**Q.48** What value must be returned to the operating system on the successful completion of a program?

- (a)0      (b)-1      (c)1      (d)Programs should not return a value

**Q.49** What is the only function all programs must contain ?

- (a)start()      (b)system()      (c)main()      (d)program

**Q.50** What is the function from where C programs begins their execution?

- (a)start()      (b)begin()      (c)main()      (d)program()

**Q.51**What punctuation is used to indicate the start and end of code blocks?

- (a){and}      (b)<and >      (c)[and]      (d)(and)

**Q.52** Which of the following is the correct way of writing comments?

- (a)/\*comments/\*      (b)/\*comment\*/  
(c)\*\*comment\*\*      (d){comment}

**Q.53** Which of the following is not a name of data type in C?

- (a)double      (b)floa  
(c)int      t      (d)real

**Q.54**Which relational operator is used for comparison?

- (a):=      (b)==      (c)equal      (d)=

**Q.55** Which is the Boolean operator logical AND ?

- (a)&      (b)|      (c)&&      (d)||

**Q.56**Evaluate !(1 && !(0||1))

- (a)True      (b)False      (c)Error      (d)Cannot be evaluated

**Q.57**What is the result of 16>>2?

- (a)1      (b)8      (c)2      (d)4

**Q.58** Find the output of the following program?

```
#include<stdio.h>
Void main()
{
char letter=' '
printf("\n%c" letter)
}
(a)A      (b)65      (c)Error      (d)Garbage value
```

**Q.59** Find the output of the following program

```
#include<stdio.h>
void main()
```

```
{
Int a;
Printf("%d" a^a)
}
(a)1 (b)0 (c)infinite (d)Error
```

**Q.60 find the output of the following program?**

```
#include<stdio.h>
void main()
{
int x=0,y=0;
x=(y=75)+9;
printf("\n%d %d" x y)
}
(a)75,9 (b)75,84 (c)84,75 (d)None of above
```

**Q.61Find the output following C program?**

```
#include<stdio.h>
#define a 5+2
int main()
{
int ans;
ans=a*a*a;
printf("%d" ans)
return 0;
}
(a)133 (b)343 (c)27 (d)None of above
```

**Q.62 Find the output the following C program?**

```
#include<stdio.h>
int main()
{
char x=65;
x=x+10;
printf("%d" x)
return 0;
}
(a)21 (b)18 (c)15 (d)None of above
```

**Q.63 Find the output of the following c****program?**

```
#include<stdio.h>
{
Int i=4,ans;
ans=++i+ ++i + ++i;
printf("%d" ans)
return 0;
}
(a)21 (b)18 (c)15 (d)None of the above
```

**Q.64 Find the output of the following c program?**

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

```
{  
Int xa=10;  
printf("%d%d%d" x x++ ++x)  
return 0;  
}
```

- (a)11 11 11                         (b)12 10 10                         (c)12 11 10                         (d)12 11 11

**Q.65 Find the output of the following C program?**

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

```
int main()  
{  
Printf("%d" sizeof(3 3))  
Return 0;  
}  
(a)2                                 (b)4                                 (c)8                                 (d)compiler error
```

**Q.66 Find the output of the following C program?**

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

```
int main()  
{  
int i=32,j=32,k,l,m;  
k=i | j;  
l= i & j;  
m=k^l;  
printf("%d %d %d %d %d\n" = j k = m)  
return 0;  
}  
(a)0,0,0,0,0                         (b)0,32,32,32,32  
(c)32,32,32,32,0                     (d)32,32,32,32,32
```

**Q.67 What are the different type of real data type in C?**

- (a)float,double,char                     (b)short int,double,long int  
(c)float,double,long double             (d)double,long int,float

**Q.68 Which of the following is not logical operator ?**

- (a)&                                     (b)&&                                     (c)||                                     (d)!

**Q.69 What is the output following C program?**

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

```
int main()  
{  
int k,num=30;  
k=(num < 10) ? 100:200;  
printf("%d%d" num k)  
return 0;  
}
```

- (a)200 30                             (b)30 200                             (c)100 200                             (d)500 500



(a) 2 bytes      (b) 4 bytes      (c) 8 bytes      (d) 10 bytes

**Q.88** The memory space taken for a signed char type data is

(a) 2 bytes      (b) 4 bytes      (c) 8 bytes      (d) 10 bytes

**Q.89** Which of the following is not an escape sequence

(a) \n      (b) \b      (c) \c      (d) \a

**Q.90** Which of the following is an escape sequence

(a) \d      (b) \e      (c) \f      (d) \g

**Q.91** Which of the following is not an escape sequence

(a) \\      (b) \?      (c) \'      (d) \;

**Q.92** Which of the following is an escape sequence

(a) \:      (b) \+      (c) \'      (d) \;

**Q.93** The space taken for a unsigned char type data is

(a) 2 bytes      (b) 4 bytes      (c) 8 bytes      (d) 1 Byte

**Q.94** The space taken for a unsigned int type data is

(a) 2 bytes      (b) 4 bytes      (c) 8 bytes      (d) 10 bytes

**Q.95** Match the column

i) \n	(a) back space
ii) \t	(b) tab
iii) \b	(c) beep sound
iv) \a	(d) new line

(a) i-A, ii-B, iii-C, iv-D      (b) i-D, ii-B, iii-A, iv-C

(c) i-D, ii-B, iii-C, iv-A      (d) i-D, ii-C, iii-B, iv-A

**Q.95** Match the column

i) \v	(a) carriage return
ii) \t	(b) back space
iii) \b	(c) horizontal tab
iv) \r	(d) vertical tab

(a) i-A, ii-B, iii-C, iv-D      (b) i-D, ii-B, iii-A, iv-C

(c) i-D, ii-B, iii-C, iv-A      (d) i-D, ii-C, iii-B, iv-A

**Q.97** Suppose the following statements are written :

Int i=9,j=6;

Float x=0.5,y=0.5;

Char a='a' b='b'

**Find the values of the following expression**

$(3*i-2*j)%(2*a-b)$

(a) 10      (b) 15      (c) 11      (d) 16

**Q.98** Suppose the following statements are written :

Int i=9,j=6;

Float x=0.5,y=0.5;

Char a='a' b='b'

**Find the values of the following expression**

$2 * (j/5) + (4 * (j-3)) % (i+j-2)$

(a) 7      (b) 15      (c) 14      (d) 16

**Q.99** Suppose the following statements are written:

Int i=9,j=6;  
Float x=0.5,y=0.5;

Char a='a' b='b'

Find the values of the following expression

$(x>y) \&\& (i>0) \&\& (j>5)$

- (a)-1                    (b)0                    (c) 1                    (d)2

**Q.100 Suppose the following statements are written:**

Int i=9,j=6;  
Float x=0.5,y=0.5;  
Char a='a' b='b'

Find the values of the following expression

$((x>y) \&\& (i>0)) \mid\mid (j>3)$

- (a)-1                    (b)0                    (c) 1                    (d)2

**Q.101 Suppose the following statements are written:**

Int i=9,j=6;  
Float x=0.5,y=0.5;  
Char a='a' b='b'

Find the values of the following expression

A==99

- (a)-1                    (b)0                    (c) 1                    (d)2

**Q.102 Suppose the following statements are written:**

Int i=9,j=6;  
Float x=0.5,y=0.5;  
Char a='a' b='b'

Find the values of the following expression

++i

- (a)10                    (b)11                    (c) 9                    (d)8

**Q.103 Suppose the following statements are written:**

Int i=9,j=6;  
Float x=0.5,y=0.5;  
Char a='a' b='b'

Find the values of the following expression

i++

- (a)10                    (b)11                    (c) 9                    (d)8

**Q.104 Suppose the following statements are written:**

Int i=9,j=6;  
Float x=0.5,y=0.5;  
Char a='a' b='b'

Find the values of the following expression

$!(b==98)$

- (a)0                    (b)1                    (c)- 1                    (d)98

**Q.105 Find the output of the following program**

#include<stdio.h>

```

Void main()
{
int a=2,b=3,ab=4;
int i;
int in='2'*2
char ch='c'
printf("%c %c\n" ch ++ch)
printf("%c %c\n" b ++b)
printf("%c %c%c\n" ab ab++ab)
printf("%c %c\n" a !!a)
}
(a)dd
  3 3 3
  4 4 4
  5 5 5
  3 1
(c)d c
  3 2 2
  4 3 3
  5 4 4
  3 1
(b)c d
  2 2 3
  3 3 4
  4 4 5
  3 0
(d) None of the above

```

**Q.106 Find the output of the following program.**

```

#include<stdio.h>
Void main()
{
int x=4,y=9;
int z;
z=(x++)+(-y)+y;
printf("Value=%d\n" z)
}
(a)value=22          (b)value=19
  Value=17          value=16
(c)value=22          (d)value=20
  Value=18          value=16

```

**Q.108 Find the output of the following program**

```

#include<stdio.h>
Void main()
{
int a,b,c;
a=2;b=5;c=10;
printf("value=%d\n" (a+b*c));
printf("value=%d\n" (-c/b*c-a));
printf("value=%d\n" (-a+ ++b %a));
}
(a)value=-70          (b)value=-48
  Value=-18          value=-22
  Value=0            value=0

```

- |  |                                      |
|--|--------------------------------------|
| (c) value=-48<br>Value=-18<br>Value=-2 | (d) value=20<br>value=16<br>value=-2 |
|--|--------------------------------------|

**Q.109 Find the output of the following program.**

```
#include<stdio.h>
Void main()
{
int a=5,b=3;
float c;
c=a/b;
printf("%d\n" c)
}
```

(a)0                (b)1                (c)-1                (d)None of the above

**Q.110 Find the output of the following program**

```
#include<stdio.h>
Void main()
{
clrscr();
int a=10,b,c;
c=b=a;
b-=a--;
c---a;
a---a;
a---a-a--;
printf("a=%d\nb=%d\nc=%d\n" a b c)
}
Output:
(a)a=7                (b)a=5
    b=1                b=-1
    c=3                c=1
(c)a=6                (d)None of the above
    b=6
    c=2
```

**Q.111 Find the output of the following program**

```
#include<stdio.d>
Void main()
{
int k=3,l=4,m;
m=++k +l--;
printf("Value of m %d\n" m)
m=k++ + --l;
printf("Value of m %d\n" m)
}
(a)Value of m 7                (b)Value of m 8
    Value of m 6                value of m 6
(c)value of m 7                (d)None of the above
```

Value of m 6

**Q.112 Find the output of the following program.**

```
#include<stdio.h>
Void main()
{
int a=1,b=2,c=3,d=4.75,x; x=++a +
b++ * ++c % d++;
printf("%d%d%d%d%d" a b c d x)
}
(a)2 3 4 5 2          (b) 2 3 4 1
(c) 1 2 3 4 2         (d) 1 2 3 4 5
```

**Q.113 Find the output of the following program**

```
#include<stdio.h>
Void main()
{
int x=1;
printf("%d%d%d\n" x (x=x+2) (x 2))
x<<2;
printf("%d%d%d\n" ++x x++ ++x)
}
(a)334                  (b)433
       644
446                      (d)None of the above
(c)343
       464
```

**Q.114 Find the output of the following program?**

```
#include<stdio.h>
Void main()
{
char letter=' '
printf("\n%d" letter)
}
(a) 's ascii value      (b)68        (c)Error      (d)Garbage value
```

**Q.115 Find the output of the following program?**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i=4,z=12;
clrscr();
if(i=5 || z>50)
    printf("\n Samosa");
else
    printf("\n Dosa");
getch();
}
```

- a) Samosa      b)Dosa      c) Error      d) None of above

**Q.116 Find the output of the following program?**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i=4,z=12;
clrscr();
if(i==5 && z>50)
    printf("\n Let us C");
else
    printf("\n Let us Not C");
getch();
}
```

- a) Let us C      b)Let us Not C    c) Error      d) None of above

**Q.117 Find the output of the following program?**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int p=8,q=20;
if(p==5 && q>5)
    printf("\n Why not C");
else
    printf("\n Why C");
getch();
}
```

- a) Why not C    b) Why C      c) Why Not C & Why C    d) None

**Q.118 Find the output of the following program?**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int j=4,k;
k=!5 &&j;
printf("\n k= %d",k);
}
```

- a) 4    b)5    c)0    d)45

**Q.119 Find the output of the following program?**

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

```
#include<conio.h>
void main()
{
int i=0;
clrscr();
for(;i<=2;)

printf("%d",++i);
getch();
}
a) 1 2 3      b) 0 1 2      c) 2 3 4      d) error
```

**Q.120 Find the output of the following program?**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i=4; clrscr();
printf("%d\t%d\t%d\t",i,i--,--i);
getch();
}
a) 2 3 3      b) 4 3 2      c) 2 2 2      d) 3 3 3
```

**Q.121 Find the output of the following program?**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i=4;
clrscr();

printf("%d",i);
printf("\n%d",i--);
printf("\n%d",--i);
getch();
}
a) 4 4 2      b) 4 4 3      c) 4 4 4      d) 4 3 2
```

**Q.122 Find the output of the following program?**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i=4; clrscr();
printf("%d\t%d\t%d\t",i,i++,++i);
getch();
```

a)

6 5 5

b) 6 5 4

c) 6 6 6

d) 5 5 5

**Q.123 Find the output of the following program?**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int x=5,y; y=x++;
printf("%d%d",x,y);
getch();
}
```

a) 6 5

b) 5 6

c) 6 6

d) 5 5

**Q.123 Find the output of the following program?**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int x=5;
if(i=0)
{
    Printf(" = am in Zero")
}
Else
{
    Printf(" = am in :ero")
}
getch();
}
```

a) I am in Hero      b) I am in Zero    c) Error d) None of Above

**Q. 124 Why this program runs infinite times**

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int i;
    for(i=32200;i<=32768;i++)
    {
        printf(" The Value I %d",i);
    }
}
```

a) The range of Integer      b) It will not infinite    c) Error      d) None of above

**Answer Key :**

Que No	Ans								
1	A	31	B	61	C	91	D	121	A
2	D	32	B	62	B	92	C	122	A
3	C	33	C	63	A	93	D	123	A
4	B	34	A	64	D	94	A	124	A
5	A	35	B	65	D	95	B		
6	C	36	C	66	C	96	D		
7	A	37	D	67	C	97	B		
8	A	38	C	68	A	98	C		
9	C	39	D	69	B	99	C		
10	A	40	C	70	B	100	C		
11	C	41	A	71	B	101	B		
12	B	42	A	72	B	102	A		
13	A	43	B	73	B	103	C		
14	D	44	B	74	C	104	A		
15	D	45	C	75	C	105	A		
16	A	46	D	76	B	106	C		
17	B	47	B	77	A	107	D		
18	B	48	A	78	B	108	D		
19	A	49	C	79	B	109	B		
20	D	50	C	80	D	110	C		
21	B	51	A	81	D	111	B		
22	B	52	B	82	B	112	A		
23	A	53	D	83	D	113	A		
24	D	54	B	84	A	114	A		
25	C	55	C	85	B	115	A		
26	A	56	A	86	D	116	A		
27	C	57	D	87	B	117	B		
28	A	58	A	88	D	118	A		
29	A	59	B	89	C	119	A		
30	C	60	B	90	C	120	A		

## Loops in C

**Q.1 What is the final value of x when the code int x; for(x=0;x<=10;x++){} is run ?**

- (a)10                    (b)9                    (c)0                    (d)11

**Q.2 When does the program following while (x<100) {} execute?**

- (a)When x is greater than 100                    (b)when x is greater than or equal to 100  
 (c)when x is less than 100                    (d)When x is less than or equal to 100

**Q.3 Which of the following is not a loop statement in c?**

- (a)repeat until                    (b)do while                    (c)while                    (d)for

**Q.4 Which of the following loops will definitely execute atleast once even if the condition is not satisfied**

- (a)for                    (b)while                    (c)do-while                    (d)None of the above

**Q.5 What is the error in following code?**

```
If(z=100)
Printf("z is 100")
(a)100 should be written in double quotations in the first line
(b)variable z should be inside double quotations in the first line
(c)Mistakes in the equals to operator
(d)There is no semicolon (;) at the end of first line
```

**Q.6 Looping in a program means**

- (a)Branching to be specified branch or label in the program  
 (b)repeating a given set of instruction  
 (c)Both of above  
 (d)None of above

**Q.7 The difference between while and do-while statements is**

- (a)In the while statement the control first enters into the loop then condition is tested at the end of first iteration  
 (b)In do while the condition is tested in first iteration and if the condition is true ,it enters into the loop  
 (c)The do-while statement's condition is used to decide whether to enter the loop or not whereas the while statement's condition is used to decide whether to exit the loop or not  
 (d)The while statement's condition is used to decide whether to enter the loop or not whereas the do-while statement's condition is used decide whether to exit the loop or not

**Q.8 Which of the following is not a branching statement in C?**

- (a)exit                    (b)break                    (c)goto                    (d)switch

**Q.9 Which of the following is a decision statement in C?**

- (a)if-else                    (b)switch-case                    (c)both a&b                    (d)do-while

**Q.10 Which of the following is a selection statement in C++?**

- (a)for                    (b)switch-case                    (c)while                    (d)do-while

**Q.11 The continue statement is used to:**

- (a) resume the program when it is hanged
- (b)resume the program if a break statement is given
- (c)skip the test of the statements of the loop in the current iteration
- (d) none of the above

**Q.12 Observe the following block of code and determine what happens when x=2?**

```
Switch(x)
{
Case 1 printf("x is 1")
    Break;
Case 2:
Case 3 printf("x is 3")
    break;
default:
    printf("X is not within the range")
}
```

- (a)Program jumps to the end of switch statement since there is nothing to do for x=2
- (b)The code inside default will run since there is no task for x=2,so
- (c)Will display x is 3, and then come outside the switch statement
- (d)None of above

**Q 13 Which of the following is false for a “switch” statement in C?**

- (a)break statement is false is compulsory after each case
- (b)default statement is compulsory
- (c)There is a limit on the maximum number of cases
- (d)None of the above

**Q.14 Find the output of following code**

```
#include<stdio.h>
Void main()
{
Int s=0;
While(s++<10)
{
If(s>3 && s<10)
Continue;
Printf("\n%d\t", s)
}
}
(a)1 2 3 4 5 6 7 8 9          (b)1 2 3 10
(c)4 5 6 7 8 9 10           (d)4 5 6 7 8 9
```

**Q.15 Find the output of following c code?**

```
#include<stdio.h>
Void main()
{
int a=2;
if(a==2)
{
```

```
a=a+2;
printf("%d" a)
}
Else
{
Break;
}
}
```

- (a)It will printing nothing (b)-3 (c)4 (d)Compile error

**Q.16 Find the output of the following c code**

```
#include<stdio.h>
#include<string.h>
void main()
{
int i=0;
for(;i<=2;)
printf("%d" ++i)
}
```

- (a)0 1 2 (b)1 2 3 (c)0 1 2 3 (d)Infinite loop

**Q.17 Find the output of following c code**

```
#include<stdio.h>
void main()
{
Int x;
For(x=1;x<=5;x++)
printf("%d" x)
}
```

- (a)1 2 3 4 5 6 (b)6 (c)1 2 3 4 5 (d)5

**Q 18 :ow many times “C” is get printed?**

```
#include<stdio.h>
Void main()
{
Int x;
for(x=0;x<=10;x++)
{
    If(x<5)
        Continue;
Else
    break;
printf("C")
}
```

- (a) 5 times (b)11 times (c)0 times (d)10 times

**Q.19 Find the output of the following program**

```
#include<stdio.h>
void main()
{
int j=1;
while(j<=255)
{
    printf("%d\n" j)
}
j++;
}
```

- (a) 0 times      (b) 254 times      (c) 255 times      (d) 256 times

**Q.20 Find the output of the following program**

```
#include<stdio.h>
void main()
{
int i=0;
for(;i<=5;i++);
printf("%d " i)
}
```

- (a) 0,1,2,3,4,5      (b) 5  
 (c) 1,2,3,4      (d) 6

**Q.21 find the output of the following program**

```
#include<stdio.h>
void main()
{
int x=500,y=100,z;
if(!x>=400)
y=300;
z=200;
printf("y=%d z=%d\n" y z)
}
```

- (a) y=100 z=200      (b) y=300 z=garbage  
 (c) y=100 z=garbage      (d) y=300 z=200

**Q.22 find the output of the following program**

```
#include<stdio.h>
void main()
{
int x=4;
float y=4.0;
if(x==y)
printf("x and y are equal")
else
printf("x and y are not equal")
}
```

- (a) x and y are equal      (b) x and y are not equal

- (c)Unpredictable                      (d) No output

**Q.23 find the output of the following program**

```
#include<stdio.h>
void main()
{
float a=0.7;
if(a==0.7)
printf(":i")
else
printf(":ello")
}
(a)Hi                                (b) Hello
(c)Hi Hello                        (d) None of above
```

**Q.24 find the output of the following program**

```
#include<stdio.h>
void main()
{
int i=5;
while(i-- >=0)
printf("%d "i)
printf("\n")
while(i-- >=0)
printf("%i" i)
i=5;
printf("\n")
while(i-- >=0)
printf("%d" i)
return 0;
}
(a)4,3,2,1,0,-1    4,3,2,1,0,-1
(b)5,4,3,2,1,0    5,4,3,2,1,0
(c)Error
(d) 5,4,3,2,1,0 5,4,3,2,1,0 5,4,3,2,1,0
```

**Q.25 find the output of the following program**

```
#include<stdio.h>
void main()
{
int i=1;
switch(i)
{
printf(":ello\n")
case 1:
printf(":i\n")
case 2:
printf("\nBye\n")
break;
```

```
}
```

**Q. 26 find the output of the following program**

```
#include<stdio.h>
void main()
{
char j=1;
while(j<5)
{
    printf("%d ", j++);
}
printf("\n")
```



**Q.27 To repeat a set of the statements for 25 times ,which kind of statement will be required?**



**Q.28 To perform one of the many operations selected based on a condition, which kind of statement will be required?**



**Q.29 =initializaions in the “for” loop are optional**



**Q 30** The maximum number of initializations allowed in a “for” loop are \_\_\_\_\_



**Q 31** The minimum number of initializations allowed in a “for” loop are \_\_\_\_\_



**Q 32** The maximum number of conditions allowed in a “for” loop are \_\_\_\_\_



**Q 33** The minimum number of conditions allowed in a “for” loop are \_\_\_\_\_



**Q 34** The maximum number of update/increment/decrement allowed in a “for” loop are \_\_\_\_\_



**Q 35 The minimum number of update/increment/decrement allowed in a “for” loop are \_\_\_\_\_**

(a)1                    (b)2                    (c)3                    (d)None of above

**Q.36 The for loop execution has statements inside the loop executed before checking the condition for the first time**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q 37 The " while" loop can be replaced by "for" loop in all the cases**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q 38 The" while" loop is an entry controlled loop**

(c)Depends on the condition                    (d)None of the above

(a)True                    (b)False

**Q 39 The" do-while" loop is an entry controlled loop**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q 40 The" while" loop is an exit controlled loop**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q 41 The "do- while" loop is an exit controlled loop**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q.42 There is no semicolon (;) after the condition in the syntax of the "while" loop**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q 43 There is no semicolon ( ) after the condition in the syntax of the "do-while" loop**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q.44 In the "if-else" statement "else " is optional**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q.45 There can be a condition in the brackets associated with the switch statement**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q.46 Only expression or a variable is allowed in the brackets associated with the switch statement**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q 47 "break" statement is compulsory after every case in the "switch-case" statement**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q 48 "default" statement is compulsory after every case in the "switch-case" statement**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q.49 The label in "switch-case" statement can be a condition or expression**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q 50 The label in "switch-case" statement can be only a value**

(a)True                    (b)False                    (c)Depends on the condition                    (d)None of the above

**Q 51 "break " statement when executed the control is transferred \_\_\_\_\_**

- (a)Outside the loop ,to the next statement after the loop
- (b)beginning of the loop i.e. to the first statement in the loop
- (c)outside the function, to the next function in the program
- (d)beginning of the function i.e. to the first statement in the function

**Q 52 "continue " statement when executed the control is transferred \_\_\_\_\_**

- (a)Outside the loop ,to the next statement after the loop
- (b)beginning of the loop i.e. to the first statement in the loop
- (c)outside the function, to the next function in the program
- (d)beginning of the function i.e. to the first statement in the function

**Q 53 "goto " statement transfers the control to\_\_\_\_\_**

- (a)Outside the loop ,to the next statement after the loop
- (b)beginning of the loop i.e. to the first statement in the loop
- (c)Label specified with the statement
- (d)None of the above

**Q.54 Find the output of the following program**

```
#include<stdio.h>
void main()
{
int i,j;
for(i=1;i<=2;i++)
{
    for(j=1;j<=2;j++)
    {
        printf(":ello")
    }
    printf(":i\n")
}
}
```

- (a)Hello Hello Hi  
    Hello Hello Hi
- (b) Hello Hello Hi  
        Hello Hello Hi  
        Hello Hello Hi
- (c) Hello Hello Hi
- (d)Hello Hi  
        Hello Hi  
        Hello Hi

**Q.55 Find the output of the following program**

```
#include<stdio.h>
void main()
{
int i,j;
for(i=1;i<=2;i++)
{
    for(j=1;j<=3;j++)
{
```

```

printf(":ello")
}
printf("\n")
}
}

(a)Hello Hello Hello
    Hello Hello Hello
    Hello Hello Hello
(c) Hello Hello
    Hello Hello

```

- (b) Hello Hello Hello  
Hello Hello Hello
- (d)Hello Hello  
Hello Hello Hello

**Q.56 Find the output of the following program**

```

#include<stdio.h>
void main()
{
int i,j;
for(i=1;i<=5;i++)
{
    for(j=1;j<=i;j++)
    {
        printf("1")
    }
    printf("\n")
}
}

```

- (a) 1  
11  
111  
1111  
111111
- (b) 1  
1  
1  
1  
1
- (c) 11111  
1111  
111  
11  
1

**Q.57 Find the output of the following program**

```

#include<stdio.h>
void main()
{
int i,j;
for(i=1;i<=5;i++)
{
    for(j=1;j<=i;j++)
    {
        printf("*")
    }
    printf("\n")
}

```

```

}
(a)*          (b)*****
(c)*          (d)*****
    **
    ***
    ***
    *****

```

**Q.58 Find the output of the following program**

```
#include<stdio.h>
void main()
{
int n=400;
if(n%10==0)
{
printf("Yes")
}
else
{
    printf("No")
}
}
(a)Yes          (b)No
(c)Compilation Error  (d)None of the above
```

**Q.59 Find the output of the following program**

```
#include<stdio.h>
void main()
{
int i=1,j=1;
for(;;)
{
    if(i>3) break;
    else j+=i;
    printf("%d\n" j)
    i+=j;
}
}
(a)Compile error      (b)2
(c)2                  (d)2

```

5  
3

**Q.60Find the output of the following program**

```
#include<stdio.h>
void main()
{
int i;
for(i=0;i<=8;i++)
```

```

{
    if(i%2==0)
        printf("%d\n" i+1)
    else if(i%3==0)
        continue;
    else if(i%5==0)
        break;
    printf("\nEnd of the program\n")
}
printf("\nEnd of program\n")
}

```

- (a) 1  
 End of program  
 End of program  
 3  
 End of program  
 5  
 End of program  
 End of program

- (b) 1  
 End of program  
 2  
 End of program  
 3  
 End of program  
 4  
 End of program  
 5  
 End of program

(c) Error

(d) None of the above

#### **Q.61 Select the correct answer**

- (a) I=10  
 do  
{  
 do something  
}while(I<10);  
(b)"do something" will not be executed at all  
(c)do-while loop is not a valid loop.  
(d)None of the above

#### **Q.62 Find the output**

```

void main()
{
int i=1,j=2,k=3;
if(i==1)
if(j==2)
if(k==3)

```

```

{
    printf("ok")
    break;
}
else
printf("continue")
printf("bye")
}
(a)ok          (b)okbye
(c)Misplaced break (d)None of there

```

**Q.63 Find the output**

```

void main()
{
int I,j=6;
for(;i=j;j-=2)
printf("%d" j)
}
(a)Error          (b)Garbage value
(c)642           (d)6420

```

**Q.64 Select the correct statement if „n“ is the number of times the loop is executed**

- (a)In a while loop the control conditional check is performed n times.
- (b)In a do-while loop the control conditional check is performed n+1 times.
- (c)Break is a keyword used with if and switch case.
- (d)None of these

**Q.65 Find output**

```

void main()
{
Float x=2.8,y=4;
if(x%y)
printf("Both are equal")
else
printf("Not equal")
}
(a)Both are equal      (b)Not equal
(c)Error               (d)None of these

```

**Q.66 Find the correct output**

```

void main()
{
int a=2,b=0,c=-2;
if(b,a,c)
    printf("True")
else
    printf("False")
}
(a)True            (b)False

```

(c)Compile time error      (d)Run time error

**Q.67 The break statement is used to exit from a \_\_\_\_\_**

- (a)DO loop                        (b)FOR loop  
(c)SWITCH statement              (d)all of above

**Q.68 In which statements, does a CONTINUE statement cause the control to go directly to the test condition and then continue the looping process?**

- (a)FOR and WHILE                 (b)WHILE and IF-ELSE  
(c)DO-WHILE AND IF-ELSE        (d)While and DO-WHILE

**Q.69 Find the output of following program**

```
#include<stdio.h>
void main()
{
int i;
for(i=0;i<10;i++)
printf("%d" i)
}
(a)0 1 2 3 4 5 6 7 8 9                        (b)Compile Error
(c)Run Time Error                                    (c)9
```

**Q.70 Find the following program**

```
#include<stdio.h>
void main()
{
int i=2,j=2;
while(i+1?—i:j++)
printf("%d" j)
}
(a)1                                                    (b)2                                                    (c)3                                                    (d)4
```

**Q.71 Find the following program**

```
#include<stdio.h>
void main()
{
int x=011,i;
for(i=0;i<x;i+=3)
{
printf("Error")
continue;
printf("Exit")
}
}
(a)EnterExitEnterExitEnterExit                        (b)EnterEnterEnter
(c)EnterEnterEnterExit                                    (d)None of the above
```

**Q.72 Find the output of following program**

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

```
void main()
{
int i,j;
i=j=2;
while(--i&&j++)
printf("%d%d" = j)
}
```

- (a)1 30 4                    (b)1 3                    (c)Error                    (d)None of the above

**Q.73 Find the following program**

```
#include<stdio.h>
void main()
{
int x=1;
for(;x<5;x++)
printf("%d" ++x)
}
```

- (a)1234                    (b)123456                    (c)135                    (d)24

**ANSWER KEY**

Que No	Ans								
1	D	16	B	31	A	46	A	61	A
2	C	17	B	32	A	47	B	62	C
3	A	18	C	33	A	48	B	63	C
4	C	19	C	34	D	49	B	64	D
5	C	20	A	35	D	50	A	65	C
6	B	21	A	36	B	51	A	66	A
7	D	22	A	37	A	52	B	67	D
8	A	23	B	38	A	53	C	68	A
9	C	24	A	39	B	54	A	69	D
10	B	25	A	40	B	55	B	70	B
11	C	26	D	41	A	56	A	71	B
12	C	27	A	42	A	57	C	72	B
13	C	28	B	43	B	58	A	73	D
14	B	29	A	44	B	59	B		
15	D	30	D	45	B	60	A		

## Function String Array etc.

**Q.1 Find the output of the following program**

```
#include<stdio.h>
int X=40;
void main()
{
int X=20;
printf("%d\n" X)
}
(a)20          (b)40          (c)60          (d)Error
```

**Q.2 Find the output of the following program**

```
#include<stdio.h>
void main()
{
int fun(float); int
a; a=fun(3.14);
printf("%d\n" a)
}
int fun(int aa)
{
return(int)++aa;
}
(a)3          (b)4          (c)0          (d)Error
```

**Q.3 Find the output of the following program**

```
#include<stdio.h>
void main()
{
int a[5]={2,3};
printf("%d%d%d\n" a*a+ a*2+ a*4+)
}
(a)Garbage Values      (b)2,3,3      (c)3,2,2      (d)0,0,0
```

**Q.4 Find the output of the following program**

```
#include<stdio.h>
void main()
{
diplay();
}
void display()
{
printf("=ndiaB=X")
}
(a)No Error      (b)display doesn't get invoked      (c)display() is called before it is defined
```

(d) None of the above

**Q.5 A function cannot be overloaded only by its return type**



**Q.6 A function can be overloaded with a return type if it has all the parameters same.**



**Q.7** Inline functions involves some additional overhead in running time.



**Q.8 A Function that calls itself is known as**

- (a)Inline Function
  - (b)Nested Function
  - (c)Overloaded Function
  - (d)recursive Function

**Q.9 The return type of a function that does not have any return type is declared as**



**Q.10 Parameters passed to a function are separated with**



**Q.11** Variables declared inside the parenthesis of a function have \_\_\_\_\_ visibility.



**Q.12 According to the following declaration of a function, which of the statement given below is true?**

true int function(int a,int b=2)

- (a) Variable b is of integer type and will always have value 2
  - (b) Variable a and b are of int type and the initial value of both variables is 2
    - (c) Variable b is global scope and will have value 2
    - (d) Variable b will have value 2 if not specified when calling function

**Q.13 String is an array of character arrays terminated with \_\_\_\_\_**

- (a)\n (b)\t (c)\0 (d)\1

**Q.14 The void specifier is used if a function does not have return type.**



**Q.15 According to the following statements, select the best suitable statement**

```
int x=5,y=3,z;
```

a=add(x,y)

- (a) The function add is called by passing the values
  - (b) The function add is called by passing reference
  - (c) Both (a and b) of above
  - (d) None of above

**Q.16 According to the following code, select the best suitable statement**

```
int x=5,y=3,z;
```

a=add(&x,&y)

- (a)The function add is called by passing the values
- (b) The function add is called by passing reference
- (c) Both (a and b) of above
- (d)None of above

**Q.17 In case of arguments passed by values when calling a function such as z=add(x,y)**

- (a) ny modifications to the variables x and y from inside the function “add” will not have any effect on the variables outside the function.
- (b)The variables x y will be updated when any modification is done in the function “add”
- (c)Yhe variable x y will be modified as per modification in the function “add” but the Variable y will not be updated as per the variation in the function “add”
- (d)None of the above

**Q.18 If the type specifier of parameters of a function call is followed by an ampersand (&) and then the variable names, that function call is**

- (a)pass by value
- (b)pass by reference
- (c)pass by variables
- (d)none of above

**Q.19 In case of pass by reference**

- (a)The values of those variables are passed to the function so that it can manipulate them
- (b)The address of variable in memory is passed to the function so that it can use the same memory area for its processing
- (c)Both of above
- (d)None of above

**Q.20 When an array is passed to a function, it can said that \_\_\_\_\_ is passed**

- (a)Address of the array
- (b)Value of the first element of the array
- (c)Address of the first element of the array
- (d)Number if elements in the array

**Q.21 Find the output of the following program?**

```
#include<stdio.h>
void main()
{
char *str="Hello world"
printf("%s" str)
}
(a)Hello world
(b)Error
(c)Garbage value
(d)None of the above
```

**Q.22 Find the output of the following program?**

```
#include<stdio.h>
void main()
{
int array[]={10,20,30,40};
printf("%d" -2[array]);
}
(a)-60
(b)-30
```

**Q.23 Find the output of the following program?**

```
#include<stdio.h>
void main()
{
int i=10;
static int x=10;
if(x==i)
printf("Equal")
else
printf("Less than")
}
```

**Q.24 Find the output of the following program?**

```
#include<stdio.h>
void main()
{
char str*+="C-program"
int a=5;
printf(a 10?"Ps\n" "%
}
(a)C-program
(c)Error
```

**Q.25 It is compulsory to write the return type for every function**

**Q.26** The return type of a function cannot be

**Q.27 Every program must have atleast \_\_\_\_\_ function(s)**

**Q.28 The function with the name \_\_\_\_\_ is always written in every program**

**Q.29** Every function must contain minimum of \_\_\_\_\_ arguments passed to it.

**Q.30 In the function definition, the argument list must always be accompanied with the corresponding data type**

(a)True

(b)False

**Q.31 The function name follows the rules of the identifier**

(a)True

(b)False

**Q.32 A void return type for a function indicates that**

(a)The function cannot return any data

(b)The function can return any type of data

(c) The function can return any type of data except for "int"

(d)None of the above

**Q.33 The value returned by a function is returned to the**

(a)main function

(b)Operating System

(c)caller function

(d)called function

**Q.34 A function that does not return any data is called as \_\_\_\_\_ function**

(a)int

(b)void

(c)float

(d)recursive

**Q.35 Argument list is a list of parameters that the \_\_\_\_\_ has to pass to the function**

(a)main function

(b)Operating System

(c)caller function

(d)called function

**Q.36 The parameters passed by the caller function are called as the \_\_\_\_\_ parameters**

(a)actual

(b)formal

(c)informal

(d)reference

**Q.37 The parameters received by the called function are called as the \_\_\_\_\_ parameters**

(a)actual

(b)formal

(c)informal

(d)reference

**Q.38 The number of actual and formal parameters \_\_\_\_\_**

(a)can be different

(b)should not be the same

(c)should be the same

(d)cannot be same

**Q.39 The datatype of actual formal parameters \_\_\_\_\_**

(a)can be different

(b)should not be the same

(c)should be the same

(d)cannot be same

**Q.40 The prototype declaration is required when \_\_\_\_\_**

(a)calling any function

(b)calling a function which is defined before it is called

(c)calling a function which is called before it is defined

(d)None of the above

**Q.41 The prototype of a function can be written\_\_\_\_\_**

(a)only outside a function

(b)only inside a function

(c)both inside and outside a function

(d)only with prefix'#'

**Q.42 The prototype of a function should contain the data type of the parameters to be passed to that function**



**Q.43 The prototype of a function should contain the variable names of the parameters to be passed to that function**



**Q.44** The data types mentioned in the prototype of a function are to be separated by

- (a),(comma)
  - (b).(dot)
  - (c):(colon)
  - (d);(semi-colon)

**Q.45 The true of the actual and formal parameters must be same**



**Q.47 The return datatype of the function and that of the variable accepting the returned value can be different**



**Q.48 A void function cannot return any parameter**



**Q.49 The name of the function is case insensitive**



**Q.50 The prototype declaration can be written without writing the identifiers of the arguments**



**Q.51 To call a function we need to simply write the name of the function followed by the parameters to be passed in the brackets**



**Q.52** The variable used to accept the returned value from the called function must be written on the left of the function call statement separated by an ampersand (&) sign



**Q.53 The prototype declaration of a function can be the same as the header line of the function calling itself again and again**



**Q.54 A recursive function may or may not have a condition such that there is an exit from the function calling itself again**

**Q.55 The actual and formal parameters are the same variables with statement names**

**Q.56** The actual and formal parameters are \_\_\_\_\_

- (a) same variables with different names
  - (b) different variable name with same memory location
  - (c) different memory location with different variable names
  - (d) different memory location with same or different names

**Q.57** An inline function is one that \_\_\_\_\_

- (a) calls itself
  - (b) replaces the function call with the function definition
  - (c) has no return type
  - (d) none of the above

**Q.58** The advantage of an inline function is that the \_\_\_\_\_

- (a) program size becomes smaller
  - (b) execution becomes faster
  - (c) function is written in the same line with the program
  - (d) none of the above

**Q.59** A function can be defined inline by \_\_\_\_\_.

- (a) prefixing the keyword “inline” in the function declaration header
  - (b) suffixing the keyword “inline” in the function declaration header
  - (c) prefixing the keyword “inline” in the function prototype
  - (d) suffixing the keyword “inline” in the function prototype

**Q.60 An inline function cannot have any return type**



**Q.61 An inline function cannot have any return type**



**Q.62 Array is a collection of mixed data types**



**Q.63 We can have a single array containing**



**Q.64 The starting index of an array is always**



**Q.65 The index of the last element of an array of „n“ elements will be**

- (a)  $n+1$       (b)  $n$   
(c)  $n-1$       (d) none of the above

**O.66 The size of an array can be changed during the execution of the program**

**Q.67 The size of an static and cannot be initialized during the execution of the program**

**Q.68 The correct syntax of declaring an array is**

- (a)[array\_size]data\_type array \_name;
  - (b)array\_name data\_type [array\_size];
  - (c)data\_type array\_name [array\_size];
  - (d)data\_type [array\_size] array\_name;

**Q.69 The memory space allocated to the array declared as:**

```
int a[10];
```

will be \_\_\_\_\_ bytes



**Q.70 The memory space allocated to the array declared as:**

```
float a[10];
```

will be \_\_\_\_\_ bytes



**Q.71** To access an element of an array the \_\_\_\_\_ operator is used

- (a),(comma) (b);(semi-colon)  
(c)&(ampersand) (d)[] (square brackets)

**Q.72** The 10<sup>th</sup> element of an array „a“ can be accessed as \_\_\_\_\_

- (a)a[10] (b)a[11]  
(c)a[9] (d)a[8]

**Q.73** The maximum number of dimensions an array can have is \_\_\_\_\_

**Q.74 In a two-dimensions an array can have is \_\_\_\_\_**

- (a)The element with row number 'i' and column number 'j'
- (b) The element with row number 'j' and column number 'i'
- (c) The element with row number (i-1) and column number (j-1)
- (d) The element with element with row number (j-1) and column number(i-1)

**Q.75 An array of characters terminated with a null character is called as \_\_\_\_\_**

- (a)pointer
- (b)string
- (c)structure
- (d)none of the above

**Q.76 The ASCII value of the null character stored at the end of the string is \_\_\_\_\_**

- (a)65
- (b)97
- (c)0
- (d)none of the above

**Q.77 The memory space required to store the string “=ND= ” is \_\_\_\_\_ bytes**

- (a)5
- (b)6
- (c)0
- (d)infinity

**Q.78 Which of the following is a correct method of declaration for a string of 100 characters**

- (a)char a[100];
- (b) char a[101];
- (c)char a[99];
- (d)none of the above

**Q.79 To accept a string from user, which of the following is used**

- (a)getchar()
- (b) putchar()
- (b)gets()
- (d) puts

**Q.80 string accepted from user is automatically terminated with null character (“\0”)**

- (a)True
- (b)False

**Q.81 The header file that has various string functions like strcpy(),strcat(),etc is \_\_\_\_\_**

- (a)string
- (b)float
- (c)int
- (d)void

**Q.82 The strcpy() function will return a \_\_\_\_\_ datatype value**

- (a)string
- (b)float
- (c)int
- (d)void

**Q.83 The strlen() function will return \_\_\_\_\_ for the string with the value “=ND= ”**

- (a)4
- (b)5
- (c)6
- (d)none of the above

**Q.84 The initial of an automatic storage class variable is \_\_\_\_\_**

- (a)zero(0)
- (b)garbage
- (c)1
- (d)none of the above

**Q.85 The automatic storage class variable is stored in \_\_\_\_\_**

- (a)memory
- (b)CPU registers
- (c)nowhere
- (d)compile

**Q.86 The scope of a variable declared as automatic storage class is \_\_\_\_\_**

- (a)local within the function declared
- (b)global
- (c)in multiple programs
- (d)none of the above

**Q.87 The life of a variable as automatic storage class is \_\_\_\_\_**

- (a)until the program completes its execution
- (b)till the control remain in the function where it is declared
- (c)until the computer remains on
- (d)none of the above

**Q.88 The initial value of an integer storage class variable is \_\_\_\_\_**

- (a)zero(0)
- (b)garbage
- (c)1
- (d)none of the above

**Q.89 The register storage class variable is stored in \_\_\_\_\_**

- (a)memory
- (b)CPU registers
- (c)nowhere
- (d)compiler

**Q.90 The scope of a variable declared as register storage class is \_\_\_\_\_**

- (a)local within the function declared
- (b)global
- (c)in multiple programs
- (d)none of the above

**Q.91 The life of a variable declared as register storage class is \_\_\_\_\_**

- (a)until the program completes its execution
- (b)till the control remains in the function where it is declared
- (c)until the computer remain on
- (d)none of the above

**Q.92 The maximum number of register storage class variables can be**

- (a)1
- (b)2
- (c)3
- (d)None of the above

**Q.93 The initial value of an static storage class variables is \_\_\_\_\_**

- (a)zero(0)
- (b)garbage
- (c)1
- (d)none of the above

**Q.94 The static storage class variable is stored in \_\_\_\_\_**

- (a)memory
- (b)CPU registers
- (c)nowhere
- (d)compile

**Q.95 The scope of a variable declared as static storage class is \_\_\_\_\_**

- (a)local within the function declared
- (b)global
- (c)in multiple programs
- (d)none of the above

**Q.96 The life of a variable declared as static storage class is \_\_\_\_\_**

- (a)until the program completes its execution
- (b)till the control remains in the function where it is declared
- (c)until the computer remains on
- (d)none of the above

**Q.97 The initial value of an externally declared variable is \_\_\_\_\_**

- (a)zero(0)
- (b)garbage
- (c)1
- (d)none of the above

**Q.98 The externally declared variable is stored in \_\_\_\_\_**

- (a)memory
- (b)CPU registers
- (c)nowhere
- (d)compile

**Q.99 The scope of a variable declared externally, is \_\_\_\_\_**

- (a)local within the function declared
- (b)global

(c)in multiple programs

(d)none of the above

**Q.100 The life of a variable declared externally,is\_\_\_\_\_**

- (a)until the program completes its execution
- (b)till the control remains in the function where it is declared
- (c)until the computer remains on
- (d)none of the above

**Q.101 Find the output of the following program**

```
#include<stdio.h>
int x;
void f1()
{
    ++x;
}
void main()
{
    int x=10;
    f1();
    x=::x+10;
    printf("%d%d\n" x x)
}
```

- (a): 11 1
- (b)1 1
- (c) 1 1 1
- (d)None of the above

**Q.102 Find the output of the following program**

```
#include<stdio.h>
void f1()
{
extern int n3;
static int n1;
int n2=20;
n1=n1+10;
n2=n1+n2;
n3=n1+n2;
printf("%d%d%d\n" n1 n2 n3)
}
int n3;
void main()
{
register int l;
for(i=1;i<=3;i++)f1();
}
(a)10 10 10
    20 20 20
    30 30 30
(b)10 30 40
    10 30 40
    10 30 40
```

(c)10 30 40

20 40 60

30 50 80

(d)None of the above

**ANSWER KEY**

Que No	Ans						
1	A	31	A	61	B	91	B
2	D	32	A	62	B	92	C
3	A	33	C	63	A	93	A
4	C	34	B	64	A	94	A
5	A	35	C	65	C	95	A
6	B	36	A	66	B	96	A
7	B	37	B	67	A	97	A
8	D	38	C	68	C	98	A
9	C	39	C	69	B	99	C
10	A	40	C	70	D	100	A
11	A	41	C	71	D	101	A
12	D	42	A	72	C	102	C
13	C	43	B	73	D		
14	A	44	A	74	C		
15	A	45	A	75	B		
16	B	46	A	76	C		
17	A	47	B	77	B		
18	B	48	A	78	A		
19	B	49	B	79	C		
20	C	50	A	80	A		
21	A	51	A	81	C		
22	B	52	B	82	C		
23	A	53	A	83	B		
24	A	54	B	84	B		
25	A	55	B	85	A		
26	B	56	D	86	A		
27	A	57	B	87	B		
28	C	58	B	88	B		
29	D	59	A	89	B		
30	A	60	B	90	A		

## Pointer Structure Union

**Q.1 what is the output of the above program code?**

```
#include<stdio.h>
void main()
{
int i=3,*p,**p1;
p=&i;
p1=&p;
printf("%d%d%d" *p **p1 *(*p1))
}
```

- (a)444                      (b)000                      (c)333                      (d)433

**Q.2 which of the following is the correct way of declaring a float pointer:**

- (a)float ptr;                      (b)float \*ptr;  
(c)\*float ptr;                      (d)None of the above

**Q.3 The size of the structure can be determined by**

- (a)size of variable name  
(b)size of(struct tag)  
(a)Only a                      (b)Only b                      (c)Both a and b                      (d)None of the above

**Q.4 An entire structure or union variable can be assigned to another structure or union variable if**

- (a)The two variables have same composition  
(b)the two variable have same type  
(c)Assignment of one structure or union variable to another is not possible  
(d) None of the above

**Q.5 Find the output of the following program**

```
#include<stdio.h>
void main()
{
int i=32;
char *ptr=(char*)&i;
printf("%d" *ptr)
}
```

- (a)1                      (b)32                      (c)compile error                      (d)None of the above

**Q.6 Find the error in the following declaration?**

```
struct author
{
    int age;
    struct inner
    {
        char name[20];
    };
};
```

- (a)Nested structure is not allowed in C

- (b)It is necessary to initialize the member variable of a structure
- (c)inner structure must have a name
- (d)There is no error

**Q.7 Find the output of the following program**

```
#include<stdio.h>
void main()
{
int array[]={10,20,30,40};
printf("%d" 2*array+)
```

- (a)60
- (b)30
- (c)garbage value
- (d)compile error

**Q.8 Find the output of the following program**

```
#include<stdio.h>
void main()
{
double far* p,q;
printf("%d" sizeof(p)+sizeof(q));
```

- (a)12
- (b)8
- (c)4
- (d)compile error

**Q.9 Which of the following is not user defined data type?**

I:  
struct book  
{  
    char name[10];  
    int pages;

II:  
long int x=2.35;  
III:

enum day{Sun,Mon,Tue,Wed};

- (a)I
- (b)II
- (c)III
- (d)Both I and II

**Q.10 Find the output of the following program**

```
#include<stdio.h>
void main()
{
struct employee
{
    char name;
    int age;
    float sal;
};

struct employee e =,"Rajesh"-
```

- ```
printf("%d%f" e age e sal)
}

(a)0, 0.000000                         (b)Garbage value                     (c) error                             (d)None of the above
```

**Q.11 point out the error in the following program**

```
#include<stdio.h>
struct emp
{
    char name[20];
    int age;
};
void main()
{
    emp struct xx;
    int a;
    printf("%d" a)
}
```

- (a)Error:in printf      (b)Error: in emp struct xx;      (c)No error      (d)None of the above

**Q.12 Which of the structure is correct?**

```
1: struct book
{
    char name[10];
    inr pages;
};

2: struct aa
{
    char name[10];
    int pages;
};

3: struct aa
{
    char name[10];
    int pages;
};
```

- (a)1      (b)2      (c)3      (d)all of above

**Q.13 What is the similarity between a structure ,union and enumeration?**

- (a)All of them let you define new values
- (b) All of them let you define new datatype values
- (c) All of them let you define new pointers
- (d) All of them let you define new structures

**Q.14 What will be the output of the program?**

```
#include<stdio.h>
void main()
{
    union var
    {
        int a,b;
    };
    union var v;
```

```

v.a=60; v.b=70;
printf("%d\n" v a)
}
(a)60           (b)70           (c)30           (d)0

```

**Q.15 What will be the output of following program?**

```

#include<stdio.h>
struct course
{
    int courseno;
    char coursename[25];
};
void main()
{
    struct course c*+=,1 "FPL"-,
,2 "Maths"-,
,3 "Physics"--;
printf("%d" c*1+ courseno)
printf("%s\n" *(c+2)) coursename)
}
(a)3 Physics      (b)2 Maths       (c)1FPL        (d)2 Physics

```

**Q.16 Pointer store \_\_\_\_\_**

- (a)value (b)address (c)both value and address (d)None of above  
ANS=(b)

**Q 17 To declare a pointer for an “”int” type variable which if the following is correct statement**

- (a)int \*p; (b)\*int p; (c)float \*p; (d)\*float p;

**Q.18 The name of a pointer has to follow the rules of an identifier**

- (a)True (b)False

**Q 19 With reference to the pointers the ”\*” operator returns the \_\_\_\_\_**

- (a)address (b)value (c)product (d)none of above

**Q 20 With reference to the pointers the ” ” operator returns the \_\_\_\_\_**

- (a)address (b)value (c)product (d)none of above

**Q.21 We can have pointer to another pointer in C programming language**

- (a)True (b)False

**Q.22 Find output of the following program**

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

```
void main()
```

```
{
```

```
    int a,p*;
```

```
a=125;
```

```
p=&a;
```

```

printf("%d\n"    a)
printf("%x\n"    p)
printf("%d\n" *p)
}
(a)125
    Address of variable a
    Address of variable b
(c)125
    125
    125

```

(b)125  
 Address of variable a  
 125  
 (d) Address of variable a  
 125  
 Address of variable a

**Q.23 Find output of the following program**

```

#include<stdio.h>
void main()
{
    int a,p*,**p1;
a=125;
p=&a;
p1=&p;
printf("%d\n" a)
printf("%x\n" p)
printf("%x\n" p1)
printf("%d\n" *p)
printf("%x\n" *p1)
printf("%d\n" **p1)
}

```

(a)125  
 125  
 125  
 125  
 125  
 125

(c)125

(b)125  
 Address of variable a  
 Address of pointer variable p  
 125  
 Address of variable a  
 125  
 (d)125  
 Address of variable a  
 Address of variable a  
 125  
 Address of variable a  
 125

**Q.24 Find the output of the following program**

```

#include<stdio.h>
void main()
{
int a,*a1;
float b,*b1;
a1=&a;
b1=&b;
printf("%x\n%x\n" a1 b1)
a1++;

```

- ```

b1++;
printf("%x\n%x\n" a1 b1)
}

(a)value of variable a value of
variable b (value of
variable a)+1
( value of variable b)+1

(b)Address of variable a Address
of variable b (Address of
variable a)+1 (Address of
variable b)+1

```
- (b) value of variable a  
 value of variable b  
 $(\text{value of variable a})+2$   
 $(\text{value of variable b})+4$
- (d) Address of variable a  
 Address of variable b  
 $(\text{Address of variable a})+2$   
 $(\text{Address of variable b})+4$

**Q.25 find the output of the following program**

```
#include<stdio.h>
void main()
```

```
{
clrscr();
int i,a[2]={10,20};
for(i=0;i<=1;i++)
{
printf("%d\n" a*i)
printf("%d\n" *(a+1))
printf("%d\n" *(i+a))
}
}
```

(a)10

10  
10  
20  
20  
20

(b)10

20  
10  
20  
10  
20

(c)10

10  
10  
10  
10  
10

(d)20

20  
20  
20  
20  
20

**Q.26 Read the statements given below and select the correct statement**

```
int a,*p,**p1;
```

```
p=&a;
```

```
p1=%p;
```

(a)p1 is a pointer to pointer p

(c)both (a) and (b)

(b)p is pointer to variable a

(d)none of the above

**Q.27 When a float pointer is decremented, it decrements by \_\_\_\_\_**



**Q.39** A structure inside another structure can be declared and is called as nested structure  
(a)True    (b)False

**Q.40** Data is more secure in structure as compared to that in union  
(a)True    (b)False

**Q.41**Select the correct answer

int \*p,i[3];  
i[0]=0;i[1]=1;i[2]=2;

P=&i[1];

what is the value of expression \*P++?

- (a)0     (b)1  
(c)2    (d)undefined

**ANSWER KEY**

Que No	Ans								
1	C	11	B	21	A	31	B	41	B
2	B	12	D	22	B	32	A		
3	C	13	B	23	B	33	B		
4	B	14	B	24	D	34	A		
5	B	15	D	25	A	35	B		
6	C	16	B	26	C	36	B		
7	B	17	A	27	C	37	A		
8	A	18	A	28	B	38	A		
9	B	19	B	29	A	39	A		
10	A	20	A	30	A	40	A		

**ANSWERS TO FOLLOWING QUESTIONS ARE IN BOLD**

**1. What is an IDE?**

- a. Internet Debugging Editor
- b. Integrated Development Environment**
- c. Interdependent element

**2. At which stage are #include and #define identified:-**

- a. Precompilation**
- b. Compilation
- c. Linking

**3. Which of these commands would give you access to the printf function:-**

- a. include stdio.h;
- b. #include <stdio.h>**
- c. #include conio.h;

**4. How would you declare a constant of 5 called "MYCONST"?**

- 1. constant MYCONST = 5;

- 2.** int myconst = 5;  
3. #define MYCONST 5
- 5. How would declare two integers called "i" and "j"?**
1. int i, j;
  2. int i + j:
  3. int i int j;
- 6. Which of the following declarations could store the number 5.5?**
1. char num;
  2. int num;
  3. float num;
- 7. What is a variable?**
1. A place to store single items of data that cannot change
  2. A place to store a list of data
  3. **A place to store a single item of data that can be overwritten**
- 8. How would you display an integer variable 'i' starting with the text "Total: "?**
1. printf( 'Total: %i' i );
  2. printf( "Total: %d", i );
  3. printf( "Total: " + i )
- 9. Which of these is NOT a valid name for a C variable:**
1. Hello There
  2. HELLO\_THERE
  3. HelloThere
- 10. What value would be stored in an integer variable "i" as a result of the following calculation:**
- ```
int i, j;
j=3;
i = 4 + 2 * j / ( j - 1 );
```
1. 1
  2. 7
  3. 9
- 11. Which of the following would read a decimal number into a float variable 'f' from the keyboard?**
1. readf ( f );
  2. scanf ( "%f", &f );
  3. scanf ( "&f", f );
- 12. Which of the following will NOT increase an integer variable "i" by 1?**
1. i++;
  2. i+=1;
  3. i=i+i;

**13. Which of the following *for* loops will display a count from 1 to 10 given an integer variable „i“ has already been declared?**

1. `for ( i = 0; i++; i<10) printf( "i is %d", i);`
2. `for ( i = 1; i<10; i++); printf( "i is %d", i);`
3. `for ( i = 1; i<=10; i++) printf( "i is %d", i);`

**14. Which of the following commands would read a single character from the keyboard and place the result in a character variable 'ch' defined as:**

`char ch;`

1. `ch = getch();`
2. `printf( "%c", ch );`
3. `getkeyb ( ch );`

**15. Which of the following would you use to place a comment into your program?**

1. `REM This is a comment`
2. `/* This is a comment */`
3. `{ This is a comment }`

**16. Single line comment will be given by**

1. `//`
2. `/* _____ */`
3. `REMARK`

**17. What number would be shown on the screen after the following lines of C are executed?**

`char ch; int i; ch='G'; i = ch - 'A'; printf( "Number: %d\n", i );`

1. 6
2. 7
3. 8

**18. How would you copy the name "Hello" to a character array (i.e. string) declared as follows:-**

`char str[10];`

1. `str = "Hello";`
2. `printf( str, "Hello" );`
3. `strcpy( str, "Hello" );`

**19. Which of the following switch statements will show the correct days of the week, where 0=Sunday, 1=Monday and 2 = Tuesday (the others are ignored). The initial day value is held in the variable 'day'?**

| (a)                                                                                                                                                                                | (b)                                                                                                                                    | (c)                                                                                                                                                           |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre>switch ( day ) {     case(0): printf("Sun");                break;     case(1): printf("Mon");                break;     default: printf("Tue");               break; }</pre> | <pre>switch ( day ) {     case(0): printf("Sun");     case(1): printf("Mon");     case(2): printf("Tue");               break; }</pre> | <pre>switch ( day ) {     case(0): printf("Sun");                break;     case(1): printf("Mon");                break;     case(2): printf("Tue"); }</pre> |

- 20. Which of the following programs will correctly add up a list of five numbers and show the total?**
- ```
int count, num, total;
```

(a)	(b)	(c)
<pre>total = 0; for ( count=1; count&lt;5;       count++ ) {     printf( "Num %2d: ",             count );     scanf( "%d", num );     total += num; } printf( "Total is: %4d\n",         total );</pre>	<pre>total = 0; for ( count=0; count&lt;5;       count++ ) {     printf( "Num %2d: ",             count );     scanf( "%d", num );     total = num; } printf( "Total is: %4d\n",         total );</pre>	<pre>total = 0; for ( count=1; count&lt;=5;       count++ ) {     printf( "Num %d: ",             count );     scanf( "%d", num );     total += num; } printf( "Total is: %d\n",         total );</pre>

- 21. Which of the following would you use to test if the variable 'i' contains 3, and if it is does display "YES" otherwise display "NO"?**
1. if ( i == 3 ) printf( "YES" ); else printf("NO");
  2. if ( i == 3 ) printf( "NO" ); else printf("YES");
  3. if ( i != 3 ) printf( "YES" ) else printf("NO");

- 22. Which of the following three programs would you consider to be well indented?**

(a)	(b)	(c)
<pre>int i, j = 0;  for (i=0; i&lt;=5; i++) {     printf("i:%d\n", i);     for (i=0; i&lt;=5; i++)     {         printf("j:%d\n", j);     } }</pre>	<pre>int i, j = 0;  for (i=0; i&lt;=5; i++) {     printf("i:%d\n", i);     for (i=0; i&lt;=5; i++)     {         printf("j:%d\n", j);     } }</pre>	<pre>int i, j = 0;  for (i=0; i&lt;=5; i++) {     printf("i:%d\n", i);     for (i=0; i&lt;=5; i++)     {         printf("j:%d\n", j);     } }</pre>

- 23. Which command is used to skip the rest of a loop and carry on from the top of the loop again?**
1. break;
  2. resume;
  3. continue;

- 24.What will be output of the following program:**

```
int i=10;
if(i==12)
    printf(" = am in True")
else
```

```
printf(" = am in false")
```

- a. I am in True
- b. I am in false
- c. Error
- d. None of Above

24. What will be output of the following program:

```
int i=10;
if(i==12)
    printf(" = am in True")
else
    printf(" = am in false")
```

- a. I am in True
- b. I am in false
- c. Error
- d. None of Above

25. What will be output of the following program:

```
int i=10;
if(i=0)
    printf(" = am in True")
else
    printf(" = am in false")
```

- a. I am in True
- b. I am in false
- c. Error
- d. None of Above

26. What will be output of following program

```
int i=4;
printf("%d%d%d" i ++i i++)
```

- a. 4,5,6
- b. 4,6,6
- c. 4,4,5
- d. 6,6,4

27. What will be output of following program

```
int i=4;
printf("%d" i)
printf("%d" ++i)
printf("%d" i++)
```

- a. 4,4,5
- b. 4,5,5
- c. 4,5,6

**28. What will be output of following program**

```
int i=0;  
for(;i<=2;)
```

```
printf("%d",++i);
```

```
getch();  
}
```

- a. Error
- b. 1,2,3
- c. 2,3,4
- d. None of Above

**29. What will be output of following program**

```
int i=4;  
printf("%d\t%d\t%d",i,i--,-i);
```

- a. Error
- b. 2,3,3
- c. 3,2,1
- d. None of Above

**30. What will be output of following program**

```
int i=4;  
printf("%d",i);  
printf("\n%d",i--);  
printf("\n%d",-i);
```

- a. 4,4,2
- b. 2,3,4
- c. 3,2,1
- d. None of Above

**31. What will be output of following program**

```
{  
int i=4,x;  
x=++i + ++i + ++i;  
printf("%d",x);  
}  
a. 20  
b. 21  
c. 18  
d. 22
```

**32. What will be output of following program**

```
int x=5,y; y=x++;
printf("%d%d",x,y);
```

- a. 6,5
- b. 5,6
- c. 6,7
- d. 6,6

## PART B

**5. What are the three main types of computer programming languages?**

- (A) Machine language, assembly language, high level language
- (B) Imperative language, functional language, declarative language
- (C) COBOL, Fortran-77, C++
- (D) A & C

**6. From the point of view of the programmer what are the major advantages of using a high-level language rather than internal machine code or assembler language?**

- (A) Program portability
- (B) Easy development
- (C) Efficiency
- (D) All of above

**9. Compiler translates**

- (A) High Level Language into m/c Level Language
- (B) m/c Level Language into high level Language
- (C) Low level Language into m/c language
- (D) None of above

**11. Any COBOL program has total**

- (A) One division
- (B) Three division
- (C) Two division
- (D) Four division

**12. One of the Cobol Program division is**

- (A) Environment Division
- (B) Coding division
- (C) Specification Division
- (D) Editing Division

**13. Which language is written as string of binary 1s and 0s?**

- (A) High Level Language
- (B) Machine Language
- (C) Assembly Language
- (D) None of the above

**14. Which program or set of instruction that the computer can understand directly without the help of translating program?**

- (A) Machine Language Program
- (B) None of the Above
- (C) High Level Language Program
- (D) Assembly Language Program

**15. Which of the language programmer must have the entire knowledge of the hardware structure of the computer?**

- (A) All the above
- (B) Assembly Language
- (C) Machine Language
- (D) High Level Language

**16. Every computer has a set of operation code called as .**

- (A) Data Set
- (B) Both
- (C) None
- (D) Instruction Set

**17. Which language allows instructions & storage locations to be represented by letters & symbols instead of numbers?**

- (A) Assembly Language
- (B) High Level Language
- (C) Machine Language
- (D) All The Above

**18. Which of the translator program converts assembly language program into equivalent machine language program?**

- (A) Compiler
- (B) Linker
- (C) Assembler
- (D) Interpreter

**19. Which language makes use of mnemonics instead of numeric op-codes & symbolic names for data locations instead of numeric address?**

- (A) Machine Language
- (B) Assembly Language
- (C) None
- (D) High Level Language

- 20. Which of the programming language is said to be machine independent language?**  
 (A) High Level Language (B) Machine Language (C) Assembly Language (D) All the Above
- 21. Which of the translator program converts high level language into its equivalent machine language?**  
 (A) Interpreter (B) Linker (C) Assembler (D) Compiler
- 22. Which program resides permanently on secondary storage?**  
 (A) Interpreter (B) Linker (C) Assembler (D) Compiler
- 23. Which program takes multiple object program files "fits them together to assemble them into the program's final executable form?**  
 (A) Assembler (B) Interpreter (C) Compiler (D) Linker
- 24. The intermediate language is based on ?**  
 (A) Intermediate Definition Language (B) Machine Language  
 (C) High Level Language (D) Assembly Language
- 25. Which of the programming language can be executed on many different types of computers with very less effort?**  
 (A) Intermediate Definition Language (B) Assembly Language  
 (C) Machine Language (D) High Level Language
- 26. Which of the language is sometimes also referred as self-documenting' language?**  
 (A) High Level Language (B) Machine Language (C) Assembly Language (D) None of the above
- 27. Which of the language is said to be one of the oldest high level languages?**  
 (A) BASIC (B) COBOL (C) PASCAL (D) FORTRAN
- 28. Which of the language was designed to solve the scientific & engineering problems?**  
 (A) FORTRAN (B) PASCAL (C) BASIC (D) COBOL
- 29. Who developed the language FORTRAN?**  
 (A) Grace Hopper (B) John Kemeny (C) John Backus (D) None of the above
- 30. When was the language FORTRAN developed?**  
 (A) 1960 (B) 1957 (C) 1980 (D) 1972
- 31. Which of the language became the first standardized language?**  
 (A) COBOL (B) BASIC (C) PASCAL (D) FORTRAN
- 32. Who developed the language COBOL?**  
 (A) Nicklaus Wirth (B) John Backus (C) Grace Hopper (D) John Kemeny
- 33. Which version of FORTRAN was oriented towards structured programming approach?**  
 (A) FORTRAN 77 (B) FORTRAN II (C) FORTRAN N (D) FORTRAN 90
- 34. What is the latest version of FORTRAN?**  
 (A) FORTAN 77 (B) FORTAN 90 (C) FORTAN II (D) FORTAN N
- 35. Which Language was designed for business data processing applications?**  
 (A) COBOL (B) PASCAL (C) BASIC (D) FORTRAN
- 36. What is the latest version of COBOL?**  
 (A) COBOL 74 (B) COBOL 85 (C) COBOL 2002 (D) None of the above
- 37. Which language is said to be a verbose language?**  
 (A) PASCAL (B) FORTRAN (C) COBOL (D) BASIC
- 38. When was the language BASIC developed?**  
 (A) 1958 (B) 1964 (C) 1970 (D) 1985
- 39. Who developed BASIC?**  
 (A) John Kemeny & Thomas Kurtz (B) Glace Hopper  
 (C) John Backus (D) Nicklaus Wirth
- 40. Which language is said to be the first high-level language to be implemented on personal computers when they were introduced?**  
 (A) JAVA (B) C++ (C) C (D) BASIC
- 41. Which of the language can be used for both business & scientific applications?**

- (A) None      (B) PASCAL      (C) BASIC      (D) Both B & C  
**42. The language PASCAL is named after which French mathematician ?**  
 (A) Grace Hopper      (B) Blaise Pascal      (C) John Backus      (D) None of the above  
**43. When was PASCAL developed?**  
 (A) 1971      (B) 1984      (C) 1956      (D) 1949  
**44. Who developed PASCAL?**  
 (A) Blaise Pascal      (B) Grace Hopper      (C) Nicklaus Wirth      (D) John Backus  
**45. Which Language was developed on the concepts of structured programming?**  
 (A) JAVA      (B) BASIC      (C) C      (D) PASCAL  
**46. When did ANSI standardized PASCAL?**  
 (A) 1971      (B) 1960      (C) 1983      (D) None above  
**47. When did ANSI standardized BASIC?**  
 (A) 1964      (B) 1952      (C) 1980      (D) 1978  
**48. When did ANSI standardized COBOL?**  
 (A) 1959      (B) 1968      (C) 1952      (D) 1978  
**49. Which version of FORTRAN was standardized by ANSI in 1966?**  
 (A) FORTRAN IV      (B) FORTRAN 77      (C) FORTRAN 90      (D) FORTRAN II  
**50. When did ANSI standardized FORTRAN?**  
 (A) 1964      (B) 1954      (C) 1975      (D) 1966
- ANSWERS**
5. A      21. D      6. D      9.A      11. D      12. A      13. B      14. A      15. C      16. D      17. A  
 18. C      19. B      20. A      22. 23. D      24. A      25. D      26. A      27. D      28. B      29.C  
 30. B      31. D      32. C      33. A      34.B      35. A      36. C      37. C      38. A      39. A      40. D  
 41. D      42. B      43. A      44. C      45. C      46. C      47. D      48.B      49. A      50. D

- 1. Which of the language is said to be the high level language?**  
 A. C++      B. Java      C. C      D. All the above  
**2. A compiler converts program into which language?**  
 A. High level language      B. Machine Language  
 C. Assembly language      D. C language  
**3. Which of the following translates the source program statements into object codes?**  
 (A) Debugger      (B) Interpreter      (C) Assembler      (D) Compiler  
**4. UNIX is closely associated with Which of the following language?**  
 (A) Java      (B) C      (C) PASCAL      (D) All the above  
**5 Same program in C can be executers This is referred as**  
 (A) Robust      (B) Reusable      (C) Highly portable      (D) All of the above  
**6. The basic combine programming language become the offspring of which language?**  
 (A) C      (B) COBOL      (C) PASCAL      (D) FORTRAN  
**7 n interpreter reads code of the program one line at time**  
 (A) Executable      (B) source      (C) Machine      (D) None of the above  
**9 BCPL is said to be the predecessor to the language**  
 (A) C++      (B) PASCAL      (C) Java      (D) C  
**10. A Compiler is**  
 (A) a combination of computer  
 (B) a program which translates from one hardware high-level to a machine level  
 (C) a program which translates from one high-level language to another  
 (D) None of these  
**11 Computer Software includes**

- (A) Application programs
- (B) Operating system programs
- (C) packaged programs
- (D) All of these

**12. Assembly language**

- (A) used alphabetic codes
- (B) is the easiest language to write machine language
- (C) Place of binary numbers used programs machine language need not be translated into
- (D) None of these

**13. A source program is**

- (A) A program Written in a machine language
- (B) a program to be translated into machine language
- (C) A machine level translation of a program
- (D) None of these

**14 runs on computer hardware and serve as platform for other software's to run on**

- (A) Operating system
- (B) Application software
- (C) system software
- (D) All

**15 is the layer of a computer system between the hardware and the user program**

- (A) Operating environment
- (B) Operating system
- (C) system
- (D) None environment

**16 The primary purpose of an operating system is**

- (A) To make the most efficient use of the computer hardware
- (B) To allow people to use the computer
- (C) To keep systems programmer employed
- (D) To make computers easier to use

**17 system is built directly on the hardware**

- (A) Environment
- (B) System
- (C) Operating
- (D) None

**18. Multiprogramming systems**

- (A) Are easier to develop than single programming systems
- (B) Execute each job faster
- (C) Execute more jobs in the same time period
- (D) Are used only one large mainframe computers

**19 is the first program on a computer when the computer boots up**

- (A) System software
- (B) Operating system
- (C) system operations
- (D) None

**20 shares characteristics with both hardware and software**

- (A) Operating system
- (B) Software
- (C)
- (D) None
- (E) Data

**21 is used in operating system to separate mechanism from policy**

- (A) single level implementation
- (B) Two level implementation
- (C) Multi level implantation
- (D) None

**22. Which of the following Operating System does not implement multitasking truly**

- (A) Windows 98
- (B) Windows NT
- (C) windows XP
- (D) MS DOS

**23. when a computer is first turned on or restarted a special type of absolute loader called is executed**

- (A) Compile and go loader
- (B) Boot loader
- (C) Bootstrap loader
- (D) Relating loader

**24. which of the following operation systems do you choose to implement a client-server network**

- (A) MS DOS
- (B) Windows 98
- (C) Windows 95
- (D) Windows 2000

**25. The operating' System manages .**

- (A) Memory
- (B) Processes
- (C) Disks and I/O devices
- (D) All of the above

**26. The operating system creates \_ from the physical computer**

- (A) Virtual space
- (B) Virtual computers
- (C) Virtual device
- (D) None

**27. Machine language is**

- (A) Readable
- (B) No translation required
- (C) Machine Dependant
- (D) Fast development

- 28. Out of following, what is Opcode in assembly language?**  
 (A) mov      (B) add      (C) cmp      (D) None
- 29. Who converts Assembly language into machine language?**  
 (A) Linker      (B) Assembler      (C) Interpreter      (D) Compiler
- 30. Compiler converts source code into-**  
 (A) Object code      (B) Comment      (C) Test cases      (D) Assembly Language
- 31. IDE stands for -**  
 (A) Integrated Development Environment      (B) Indian Developer Environment  
 (C) Integrated Date Environment      (D) None of the above
- 32. .... Is an example of Interpreted language.**  
 (A) C      (B) BASIC      (C) C++      (D) All above
- 33. .... Is an example of Compiled and Interpreted language.**  
 (A) C      (B) BASIC      (C) C++      (D) Java
- 34. .... Is an algebra based programming language.**  
 (A) FORTRON      (B) MATLAB      (C) COBOL      (D) BASIC
- 35. Kernel is also known as -**  
 (A) Operating System      (B) Heart of OS      (C) High Level Language (D) Low Level Language
- 36. Choose correct form of the format of assembly instruction.**  
 (A) [comment] [label] <opcode> <operand>      (B) [label]<operand><opcode> [;comment]  
 (C)<opcode>(operand)[label] [;comment]      (D) [label]<opcode><operand> [;comment]
- 37. In Java, which component is machine dependent?**  
 (A) Java Source File      (B) Java Virtual Machine (JVM)  
 (C) Java Class / (Byte code) File      (D) All of the above
- 38. Which language is having more readable, easy to debug and fast development features?**  
 (A) Machine Level Language      (B) Assembly Language      (C) High Level Language (D) All of the above
- 39. Which of the characteristic of Java language?**  
 (A) Abstraction      (B) Simplicity      (C) Portability      (D) All above
- 40. LISP is mostly used in -**  
 (A) Application Programming      (B) Web Programming  
 (C) Artificial Intelligence Application Development      (D) Operating System Development
- 41. PASCAL is mostly used in -**  
 (A) Scientific computational Application      (B) Web Application  
 (C) AI Application      (D) GUI Application
- 42. Which language is easy to understand by human being?**  
 (A) Low Level Language      (B) Assembly Language  
 (C) High Level Language      (D) None of the above
- 43. Which language is easy to understand by machine?**  
 (A) Low Level Language      (B) Assembly Language      (C) High Level Language (D) None of the above
- 44. FORTRON stands for -**  
 (A) For Translation      (B) Formula Translator      (C) Formula Translation      (D) None of the above
- 45. COBOL stands for -**  
 (A) Common Business Oriented Language      (B) Common Basic Operation Language  
 (C) Code of Business of Language      (D) None of the above
- 46. BASIC stands for -**  
 (A) Base All Some Translation Code      (B) Business At Some Translation Code  
 (C) Beginners All Purpose Symbolic Instruction Code      (D) None of the above
- 47. PL-1 stands for -**  
 (A) Programming Language 1      (B) Processing Language 1  
 (C) Prompting Language 1      (D) None of the above

### ANSWER KEY

1. D	2. B	3. D	4. B	5. C	6. A	7. B	8. C	9. C	10. B	
11. D	12. A	13. B	14. A	15. B	16. A	17. C	18. C	19. B	20. A	
21. B	22. D	23. C	24. D	25. D	26. B	27. C	28. A	29. B	30. A	
31. A	32. B	33. D	34. A	35. B	36. D	37. B	38. C	39. D	40. C	
41. A	42. C	43. A	44. C	45. A	46. C	47. A	48. C	49. A	50. A	

- 2.** Which data type is the major feature of 'c' language?  
(A) Pointer      (B) Union      (C) Structure      (D) All of the above

**3.** Which language was used to write the UNIX operating system?  
(A) PASCAL      (B) C      (C) JAVA      (D) BASIC

**4.** When did ANSI standardized 'B'?  
(A) 1964      (B) 1983      (C) 1989      (D) 1968

**5.** Who developed C++?  
(A) Bjarne Stroustrup      (B) James Gosling  
(C) Brian Kernighan      (D) None of the above

**6.** Which of the language was primarily used for internet based applications?  
(A) C      (B) JAVA      (C) LISP      (D) COBOL

**7.** When was JAVA's first commercial release made?  
(A) 1952      (B) 1966      (C) 1983      (D) 1995

**8.** JAVA comes in two variants as \_\_\_\_\_?  
(A) JRE & SDK      (B) J2SE      (C) JSP      (D) J2EE

**9.** Who developed the language C#?  
(A) Dennis Ritchie      (B) Brian Kernighan      (C) Anders Hejlsberg      (D) John McCarthy

**10.** Who developed LISP?  
(A) Brian Kernighan      (B) John Backus      (C) James Gosling      (D) John McCarthy

**11.** Which of the language is most widely used language for AI applications?  
(A) PASCAL      (B) LISP      (C) COBOL      (D) C

**12.** ..... is a type of computer language?  
(A) Machine Level Language      (B) High Level Language  
(C) Both A & B      (D) English

**13.** ssembly language closely resembles to      language  
(A) Machine Level Language      (B) High Level Language  
(C) None      (D) Instruction Set

**14.** Executable file Contains -  
(A) Machine Understandable code      (B) Program code  
(C) Text Data      (D) Object code

**15.** Source file contains -

- (A) Machine Understandable code      (B) Program code  
 (C) Text Data      (D) Object code
- 17.** **In Hungarian notation..... Prefix is used for Boolean.**  
 (A) bol      (B) b      (C) bl      (D) None above
- 18.** **In Hungarian notation..... Prefix is used for pointer.**  
 (A) ptr      (B) Pr      (C) P      (D) None above
- 19.** ..... Documentation gives high level view of the software application.  
 (A) Architecture      (B) Comment      (C) User Manual      (D) History
- 20.** **Comments are -**  
 (A) Executable statements in program      (B) Explain program logic  
 (C) Non Executable statements in program      (D) Both B & C
- 21.** **A generalized Syntax is written in -**  
 (A) Symbolic style      (B) Both A & B      (C) Normal text style      (D) None of the above
- 22.** **In generalized syntax the symbol < > indicates -**  
 (A) Greater than and Less Than      (B) Replace this place holder with value  
 (C) Brackets      (D) All of the above
- 23.** **Installation describes?**  
 (A) How to write the program      (B) How to use the program  
 (C) How to install the program      (D) How to read the program
- 24.** **Program documentation is used to?**  
 (A) increase throughput      (B) increase maintainability  
 (C) increase security      (D) None of the above
- 25.** **User manual are used for?**  
 (A) just reading      (B) to know the basic of program  
 (C) to modify program      (D) none of above
- 26.** **Latest software should support?**  
 (A) old version      (B) only new version  
 (C) All versions      (D) none of above
- 27.** **Software designing plays important role in?**  
 (A) developing software      (B) denying software  
 (C) delivering software      (D) All of above
- 28.** **Which of the term refers to the information describing various products and services to the users in both computer and software development. ?**  
 (A) Documentation      (B) Debugging      (C) Testing      (D) None of above
- 29.** ..... refers to the process of collecting, organizing and maintaining a complete record of programs and other documents used during the different phases of software development.  
 (A) Debugging      (B) Documentation      (C) Both A and B      (D) Testing
- 30.** are the forms of documentation that are put within program to help in understanding the logic of the program.  
 (A) System manual      (B) User manual      (C) Comments      (D) Logic Errors
- 31.** does not contain any programming instruction and are just language instructions.  
 (A) User Manual      (B) Comments      (C) System Manual      (D) None
- 32.** arguments the code with the basic description and also help in generating external documentation.  
 (A) Comments      (B) Hungarian Notation      (C) User Manual      (D) None
- 33.** is a form of external documentation and is available in the form of separate documents or unit development folders  
 (A) Hungarian Notation      (B) User Manual      (C) Syntax errors      (D) System Manual
- 34.** .... is a form of external documentation and is required to ensure smooth execution of software ..

- (A) Logical Errors      (B) User Manual      (C) System Manual      (D) Comments  
**35.** .... are used within program to help in understanding the logic of the program and hence are a form of external documentation.  
 (A) Debugging      (B) White Box Testing      (C) Comments      (D) Syntax Errors  
**36.** In..... the name of the variable indicates its type of intended use.  
 (A) Debugging      (B) Hungarian Notation      (C) System Manual      (D) Logic Errors  
**37.** ..... is said to be a popular naming convention in complete programming  
 (A) User Manual      (B) Hungarian Notation      (C) Black Box Testing      (D) Testing  
**38.** Following is the DOS command to print the file' add.c ' without installing a printer driver on your PC.  
 (A) type add.c> prn      (B) edit add.c > prn      (C) print add.c      (D) print add.c > prn  
**39.** Command for creating directory in MS DOS is  
 (A) md      (B) gd      (C) cd      (D) fd  
**40.** The software tool that. is used for linking modules together is called \_\_\_\_\_.  
 (A) Editor      (B) Linker      (C) Compiler      (D) Debugger  
**41.** A component of a computer that locates a given program or application from the offline storage, loads it into the main memory and facilitates its execution is called \_\_\_\_\_.  
 (A) Interpreter      (B) Compiler      (C) Linker      (D) Loader  
**42.** \_\_\_\_\_ loads a given program from a disk  
 (A) Linker      (B) Compiler      (C) Interpreter      (D) Loaders

**ANSWERS**

1. A      2.A      3. B      4.C      5.A      6.B      7.D      8.A      9.C      10. D  
 11. B      12. C      13. D      14. A      15. B      16. C      17. B      18. C      19. A      20. D  
 21. A      22. B      23. C      24. B      25. B      26. C      27. A      28. A      29. B      30. C  
 31. B      32. A      33. D      34. B      35. C      36. B      37. B      38. A      39. A      40. B  
 41. D      42. D

- 1. Flowchart is used to \_**  
 (A) Visual representation of application Logic      (B) Represent Application Modules  
 (C) Give short description of application      (D) Explain user interface of application
- 2. Step by step description of program is known as-**  
 (A) Pseudo code      (B) Flowchart  
 (C) Algorithm      (D) Test case
- 3. To show Start and End of program**  
 (A) Kite box is used.      (B) Circle is used.  
 (C) Round corner rectangle is used      (D) Pentagon is used.
- 4. Which statement is used to jump the control of program?**  
 (A) Switch      (B) Loop  
 (C) Conditional statement      (D) Goto
- 5. In programming language, programs are implementation of \_**  
 (A) Flowchart      (B) Algorithm  
 (C) None of the above      (D) Pseudo code
- 6. By using ..... , Algorithms (Program logic flow) can be shown in pictorial way.**  
 (A) Program      (B) Flowchart  
 (C) Test Case      (D) Pseudo code
- 7. Processing Symbol is used to indicate      in the program.**  
 (A) Process (i.e. Arithmetic process)      (B) Input and Output  
 (C) Decisions      (D) Start and End



**data type is expected for the variable.**

- (A) Name      (B) Termination      (C) Decision      (D) Declaration

**24. The following pseudo code is an example of a(n) \_\_\_\_\_ structure:**

*Get number*

*Get another number*

*If first number is bigger than second then*

*print first number*

*Else*

*print second number*

- (A) Sequence      (B) Decision      (C) Loop      (D) Nested

**25. The following pseudo code is an example of**

*Get number*

*Get another number*

*Add number*

*Print result*

- A) Sequence      B) Decision      C) Loop      D) Nested

**26. The following pseudocode is an example of**

**Do step a**

**Do step b**

**if condition c is true then**

**Do step d**

**else**

**Do step e**

**end if**

**while condition f is true**

**Do step g**

**end while**

- (A) Nesting      (B) Stacking      (C) Posttest      (D) Pretest

**27. The following pseudocode is an example of**

**if condition a is true then**

**Do step e**

**else**

**Do step b**

**Do step c**

**Do step d**

**end if**

- (B) Nesting      (B) Stacking      (C) Posttest      (D) Pretest

**28. In a case structure the term-----means “if none of the other cases were true ”**

- (A) Else      (B) Then      (C) Default      (D) Loop

**29. Fill in the blank in the following pseudo code:**

**If some condition is true then Do one process \_\_\_\_\_ do the 0 process**

- (A) Then      (B) While  
(C) Do      (D) Else

**30. What is another name for a loop structure?**

- (A) Execution      (B) Selection  
(C) Iteration      (D) Case

**31. A case structure can be replaced one or more \_\_\_\_\_ structures.**

- (A) If-then-else      (B) Do-while  
(C) Do-until      (D) While

**32. Which name is best suited to a module that calculates overtime pay?**

- (A) CalcoO
- (B) CoO
- (C) Calculate overtimeO
- (D) CalculateovertimeO

**33. The----- can be a useful tool when a program must be modified months or years after the original writing.**

- (A) Flowchart
- (B) Hierarchy chart
- (C) Pseudo code
- (D) Variable declaration

**34. In a \_\_ program, the user sees a screen and can typically make Selections using a mouse or other pointing device.**

- (A) Reusable
- (B) Modular
- (C) GUI
- (D) Command-line

**35. Which step occurs first?**

- (A) Understanding user's needs
- (B) Clarifying requirements
- (C) Coding program
- (D) Developing program logic

**36. Variable declarations are made in the \_\_ section of a program,**

- (A) Main loop
- (B) End-of-job routine
- (C) Housekeeping
- (D) File opening

**37. Declaring a variable involves selecting a name and a \_\_**

- (A) Size
- (B) Length
- (C) Style
- (D) Type

**38. Some use a variable-naming convention called \_\_ notation, in which a variable's data type or other information is stored as part of the name. For example, a numeric field might always start with the prefix num.**

- (A) Prefix
- (B) American
- (C) Polish
- (D) Hungarian

**39. A group of variables is often called a \_\_ .**

- (A) Linked group
- (B) Data structure
- (C) Data object
- (D) Module

**40. When a variable is \_\_ it is both declared and initialized.**

- (A) Set
- (B) Instantiated
- (C) Defined
- (D) Documented

**41. The time factor when determining the efficiency of algorithm is measured by**

- (A) counting microseconds
- (B) Counting the number of key operations
- (C) Counting the number of statements
- (D) Counting the kilobytes of algorithm

**42. The space factor when determining the efficiency of algorithm is measured by**

- (A) Counting The Maximum Memory Needed By The Algorithm
- (B) Counting The Minimum Memory Needed By The Algorithm
- (C) Counting The Average Memory Needed By The (A)lgorithm
- (D) Counting the maximum disk space needed by the algorithm

**43. Which of the following case does not exist in complexity theory**

- (A) Best case
- (B) Worst case
- (C) Average case
- (D) Null case

**44. The Worst case occur in linear search algorithm when**

- (A) Item is somewhere in the middle of the array
- (B) Item is not in the array at all
- (C) Item is the last element in the array
- (D) Item is the last element in the array or is not there at all

**45. The Average case occur in linear search algorithm**

- (A) When Item is somewhere in the middle of the array
- (B) When Item is not in the array at all
- (C) When Item is the last element in the array
- (D) When Item is the last element in the array or is not there at all

**46. -----is used to write the algorithms.**

- (A) Computer Language 'C'
- (B) Computer Language 'C++'

(C) Any Programming Language                          (D) English Language

**47. There is no symbol for expressing \_\_\_\_\_ while drawing a flowchart or writing an algorithm.**

- (A) Assertion    (B) Comparison  
 (C) Negation    (D) No Action

**48. A program design tool in which standard graphical symbols are used to represent the logical flow of data is called as a**

- (A) Flowchart    (B) Pseudocode    (C) Algorithm    (D) Structured Chart

**49. Which of the following is an iterative control structure?**

- (A) Decision Making                                    (B) Sequential    (C) Jump    (D) loop

**50. Which of the following structures are used in computer**

**programs**

- (A) Sequential    (B) Decision    (C) Iterative    (D) All of above

#### ANSWERS

1.A	2.C	3.C	4.D	5.B	6.B	7.A	8.A	9.B	10.C
11.A	12.C	13.B	14.C	15.D	16.C	17.D	18.B	19.C	20.A
21.A	22.C	23.A	24.B	25.A	26.A	27.A	28.C	29.D	30.C
31.A	32.D	33.C	34.C	35.A	36.D	37.D	38.D	39.B	40.C
41.B	42.A	43.D	44.D	45.A	46.D	47.D	48.A	49.D	50.D

**1. Which of the following scenario is correct?**

- (A) flowchart->algorithm- > programming language  
 (B) flowchart->programming language->algorithm  
 (C) algorithm->flow chart->programming language  
 (D) algorithm->programming language->flow chart

**2. Instructions in algorithms should be \_\_\_\_\_**

- (A) Precise    (B) Unambiguous  
 (C) Precise & Unambiguous                            (D) None of above

**3. As compared to a flowchart, it is easier to modify the \_\_\_\_\_ of a program logic when program modifications are necessary.**

- (A) Macro flowchart                                    (B) Micro flowchart  
 (C) Terminal     (D) Pseudo code.

**4. Algorithm halts in \_\_\_\_\_**

- (A) Finite time                                        (B) Infinite time  
 (C) Logarithmic time                                (D) Exponential time

**5. The \_\_\_\_\_ flow chart symbol(s) represents one way flow of control.**

- (A) Processing                                        (B) Decision  
 (C) Terminal    (D) All above

**6. What is an infinite loop?**

- (A) It is an endless loop                              (B) It means multiple loops  
 (C) It is a nested loop                                (D) It is an unclosed loop

**7. The normal flow of flowchart is from**

- (A) Left to Right                                      (B) Right to Left  
 (C) A & D     (D) Top to Bottom

**8. Which tool shows textual design solution?**

- (A) Flowchart    (B) Structure chart  
 (C) Pseudocode                                        (D) Algorithm

**9. Finiteness property of an Algorithm is**

- (A) The number of steps in the algorithm should be finite.  
 (B) The algorithm should terminate after a finite no. of times.  
 (C) For all possible combinations of input data, the algorithm terminates after a finite no. of steps.  
 (D) None of above
- 10. Pseudocode consist of ..... and ommits .....**  
 (A) structural conventions of programming languages;  
     subroutines, variable declarations or language- specific syntax  
 (B) subroutines; structural conventions of programming languages  
 (C) variable declarations; language- specific syntax  
 (D) subroutines; Functions
- 11. Method which uses a list of well-defined instructions to complete a task, starting from a given initial state to end state, is called**  
 (A) Program    (B) Algorithm  
 (C) High level Language  
 (D) Flowchart
- 12. Flow chart help for**  
 (A) Better communication                         (B) Efficient coding  
 (C) Program Testing                                (D) A & B
- 13. Basic symbols of flow chart are**  
 (A) Start and End                                    (B) Processing, Decision  
 (C) Input - Output                                  (D) All of above
- 14. Diamond shape in flow chart denotes**  
 (A) Start    (B) Decision  
 (C) End    (D) Input - Output task
- 15. The chart that contains only function flow and no code is called as a**  
 (A) Structure chart                                (B) Function chart  
 (C) Flowchart                                      (D) Psudochart
- 16. Amongst the flowchart symbols, which of the following is an Auxiliary symbol?**  
 (A) Sequence                                        (B) Connector  
 (C) Decision                                        (D) repetition
- 17. Which of the following shape is used for representing a Conditional Statement in a Flow chart?**  
 (A) Parallelogram                                (B) Rhombus  
 (C) Trapezoid                                      (D) Rectangle
- 18. Amongst the following symbols, which of the following is not a symbol used in a flowchart?**  
 (A) Star    (B) Terminal Box  
 (C) Input-Output Box                            (D) Diamond
- 19. Algorithm and Flowchart help us to**  
 (A) Know the memory capacity                    (B) Identify the base of a number system  
 (C) Direct the output to a printer              (D) Specify the problem completely and clearly
- 20. In a flowchart, a Data File is represented by a .... shape.**  
 (A) Diamond                                        (B) Parallelogram  
 (C) Rectangle                                      (D) Cylinder
- 21. A good algorithm should not \_\_\_\_.**  
 (A) Execute for a given set of initial conditions                                (B) Produce the correct output  
 (C) Terminate after finite number of steps                                      (D) Result into ambiguous state
- 22. \_\_\_\_\_ is used to write the algorithms.**  
 (A) Computer Language 'c'                        (B) Computer Language 'C++'  
 (C) Any Programming Language                    (D) English Language

- 23. Which of the followings is a program planning tool?**  
 (A) Compiler                                 (B) Flow Charts  
 (C) Psuedo Code                             (D) Both B and C
- 24. Which amongst the following flowchart symbols is a 'two way' branching symbol?**  
 (A) Parellogram                             (B) Connectoer  
 (C) Diamond                                 (D) Rectangle
- 25. Paralelogram is used to represent in a flowchart.**  
 (A) Decision                                 (B) Processing  
 (C) Termination                             (D) Input and Output
- 26. Connector in a flowchart represents**  
 (A) Arithmetic operation                     (B) Data movement operation  
 (C) Comparison operation                     (D) None of the above
- 27. Detailed flowchart is also called as**  
 (A) Macro flowchart                           (B) Micro flowchart  
 (C) Mini flowchart                            (D) None of L le above
- 28. Macro flowchart is also called as**  
 (A) Less detail i10wchart                   (B) More detail flowchart  
 (C) Simple flowchart                          (D) None of the above
- 29. Sentinel value is used to**  
 (A) Start a loop                              (B) Terminate a loop  
 (C) Input value                               (D) Output value
- 30. Goto statement is used for**  
 (A) Conditional jump only                   (B) Unconditional jump only  
 (C) both conditional and unconditional jumps                     (D) None of the above
- 31. There is no symbol for expressing while drawing a flowchart or writing an algorithm.**  
 (A) Assertion                                 (B) Comparison  
 (C) Negation                                 (D) No Action
- 32. A program design tool in which standard graphical symbols are used to represent the logical flow of data is called as a**  
 (A) Flowchart                                 (B) Pseudocode  
 (C) Algorithm                                 (D) Structured Chart
- 33. Any program can be written using**  
 (A) Selection logic                           (B) Sequence and selection logic  
 (C) Iterative logic                           (D) Sequence, selection and Iterative logic
- 34. Any program can be written using structures.**  
 (A) Sequence logic, Merge logic, Insertion logic  
 (B) Sequence logic, Selection logic, Iteration logic  
 (C) Sequence logic, Branch logic, Iteration logic  
 (D) None of the above
- 35. 'DO ... WHILE' and 'REPEAT .... UNTIL' structure are called**  
 (A) Sequential logic structures             (B) Decision logic structures  
 (C) Iterative logic structures                 (D) None of the above
- 36. 'IF ... THEN ... ELSE' or 'CASE 'structure are called**  
 (A) Selection logic structures             (B) Sequence logic structures  
 (C) Iteration logic structures                 (D) Program logic structures
- 37. Loops in a program are written using**  
 (A) Selection logic                           (B) Iteration logic  
 (C) Sequence logic                           (D) None of the above
- 38. Which of the following logic is used to produce loops in programme logic?**

- (A) sequence logic                         (B) selection logic  
 (C) iteration logic                         (D) none of them
- 39. Flowlines are used for**  
 (A) connecting from one page to another page     (B) input-output  
 (C) Decision logic                             (D) Indicate flow of program
- 40. Which of the following is not used as a logic structure while writing 'C' programs?**  
 (A) sequence logic                             (B) process logic  
 (C) selection logic                             (D) iteration logic
- 41. The default flow of control, particularly in imperative programming is**  
 (A) Parallel                                     (B) Sequential  
 (C) Random                                     (D) None above
- 42. In flowcharts, ellipse is used for denoting**  
 (A) Start only                                  (B) Stop only  
 (C) Both Start and End                         (D) None of these
- 43. The symbol is used to indicate the beginning, ending, and pauses in the program logic flow.**  
 (A) Flowlines                                     (B) Processing  
 (C) Input/Output                                 (D) Terminal.
- 44. To write the correct and effective program we must first**  
 (A) Draw a flowchart                             (B) Plan its logic  
 (C) Write the pseudocode                         (D) All of the above
- 45. Pseudocode is also called as the**  
 (A) Program Design Language (PDL)             (B) Microflowchart  
 (C) imitation                                     (D) Decision.
- 46. Pseudocode emphasize on the aspect of a program.**  
 (A) Development                                     (B) Coding  
 (C) design   (D) debugging.
- 47. The similarity between structure charts and flow charts is**  
 (A) both of them use top-down approach         (B) both of them use bottom-up approach  
 (C) both of them provide pictorial view.         (D) none of them hide specific language syntax
- 48. The logic is used to produce loops in program logic when one or more instruction may be executed several times depending on some conditions.**  
 (A) Iteration logic                                 (B) Selection logic  
 (C) Sequence logic                                 (D) Decision logic
- 49. Which logic is used to select the proper path out of two or more alternative paths in the program logic?**  
 (A) Looping logic                                     (B) Sequence logic  
 (C) Iteration logic                                 (D) Selection logic
- 50. The \_\_\_\_\_ is a program-planning tool that allows the programmers to plan program logic by writing program instructions in an ordinary language**  
 (A) Flowchart   (B) Pseudocode  
 (C) Program   (D) Looping

**Answers**

1. A   2. C   3. D   4. A   5. A   6. A   7. D   8. C   9. C   10. A  
 11. B   12. D   13. D   14. B   15. B   16. B   17. B   18. A   19. D   20. D  
 21. D   22. D   23. D   24. C   25. D   26. D   27. B   28. A   29. B   30. B  
 31. D   32. A   33. D   34. B   35. C   36. A   37. B   38. C   39. D   40. B  
 41. B   42. C   43. D   44. D   45. A   46. C   47. C   48. A   49. D   50. B

1. Selection logic is also called as the  
(A) Decision logic    (B) Iteration logic    (C) Sequence logic    (D) Looping logic
2. What do you mean by an iterative operations?  
(A) It is a control structure that repeats the execution of a block of instructions  
(B) It is a control structure that asks a true/false question and then selects the next instruction based on the answer  
(C) In it Instructions are executed in order  
(D) All of the above
3. Which of the following is used for making the next iteration of the loop to be started?  
(A) break    (B) case    (C) continue    (D) All the Above
4. In which of the following scenario, sequence logic will not be used?  
(A) Accepting an input from the user.    (B) Comparing two sets of data  
(C) Giving an output to the user.    (D) Adding two numbers
7. Which of the following statement is false?  
(A) Flowchart provides graphical representation of program logic  
(B) Drawing a flowchart before writing the program is better  
(C) Pseudocode gives graphical representation of program logic  
(D) Writing pseudocode before writing the program is better
8. Which of the following statements is correct?  
(A) Flowchart is a pictorial representation of an algorithm  
(B) Pseudocode is an analysis tool used for planning program logic  
(C) Both A and B are false  
(D) Both A and B are true
9. A structured chart is  
(A) A statement of information processing requirements  
(B) A hierarchical partitioning of the program  
(C) A document of what has to be accomplished  
(D) All of the above
10. Connector in flowchart is represented by  
(A) Rectangle    (C) Ellipse    (B) Diamond    (D) Circle
11. A rectangle in flowchart denotes  
(A) Start of Program    (B) Input or output function  
(C) Arithmetic and data movement instruction    (D) End of program
12. In a flowchart, flow lines are used to indicate  
(A) Beginning of program    (B) Data movement  
(C) Flow of operations    (D) All of the above
13. Which of the following symbol is not used while drawing flowchart?  
(A) Terminal    (B) Input/Output    (C) Processing    (D) Control
14. Infinite loops can be avoided by using  
(A) Sentinel    (B) Counter    (C) Algorithm    (D) Both A & B
15. Structure charts are read in direction.  
(A) left-right,top-down    (B) top-down,left-right  
(C) down~up,left-right    (D) top-down,right-left
16. An algorithm is represented as  
(A) Programs    (B) Flow charts    (C) Pseudo Codes    (D) All of above
17. A diamond is used in flowcharts to represent \_\_\_\_\_.  
(A) Arithmetic & data movement instructions    (B) Input  
(C) Output    (D) Decision

- 18.** Functional flow of a program is shown by:  
 (A) Flowchart      (B) Pseudo code      (C) Structure chart      (D) Program Map
- 19.** The is a program design tool that visually represents the solution logic.  
 (A) Flowchart      (B) Program map      (C) Pseudo code      (D) Structure chart
- 20.** The term algorithm refers to  
 (A) step by step description of how to arrive at the solution of problem.  
 (B) it is a kind of flow chart.  
 (C) it is a set of instructions in specified sequence.  
 (D) All of the above.
- 21.** Pseudo code is used to  
 (A) Run a program      (B) Compile a program  
 (C) Plan program logic using natural language      (D) Debug a program
- 22.** Algorithm can be represented in following ways except  
 (A) as a program      (B) as a flowchart      (C) as a process      (D) as a pseudo code
- 23.** Rectangle can be used for representing  
 (A) decision      (B) processing      (C) input-output      (D) none of these
- 24.** Which one of the following is the disadvantage of a flowchart?  
 (A) Efficient coding      (B) Systematic Debugging  
 (C) Better Communication      (D) None of these
- 25.** Iteration logic is used to execute instructions  
 (A) Depending upon some condition to choose one of the path  
 (B) One after another  
 (C) Several times depending upon some condition  
 (D) None of the above

**Answers :-**

1. A    2. A    3.C    4.B    5.A    6.A    7.C    8.D    9.B    10.D    11.C    12.C    13.D  
 14.D    15.B    16.D    17.D    18.C    19.A    20.A    21.C    22.C    23.B    24.D    25.C

- 1.** Which of the variable occupies 2 bytes of the memory?  
 (A) Float      (B) Double      (C) Short Integer      (D) None of the above
- 2.** The preprocessor can be denoted using which of the symbols?  
 (A) #      (B)-      (C)\$      (D) &
- 3.** The escape character \n is used for----purpose  
 (A) Tab      (B) New line      (C) Header flies      (D) None of the above
- 4.** The operator '+' has which priority?  
 (A) First      (B) Third      (C) Second      (D) Fourth
- 5.** The type of constants can be enclosed between single quotes  
 (A) Real      (B) Character      (C) Integer      (D) Float
- 6.** The type of constants have to be enclosed between double quotes.  
 (A) Float      (B) Integer      (C) String      (D) Character
- 7.** In C the maximum length of the ..... is said to be 8.  
 (A) Character      (B) String      (C) Integer      (D) Identifiers
- 8.** Which of the following variable has the maximum length of 4 bytes?  
 (A) Float      (B) Double      (C) Integer      (D) Character
- 9.** Which of the following variable the maximum length of 8 bytes?  
 (A) String      (B) Constant      (C) Integer      (D) Double
- 10.** 8 Bytes will be occupied by which the following statements?  
 (A) float a, s;      (B) char p, q;      (C) int x, y;      (D) All of above

**11. The constants in c can express in both fractional & exponential forms.**

- (A) String      (B) Character      (C) Real      (D) Integer

**12. The statement char ch='z' would store in ch**

- (A) ASCII value of z      (B) The character z      (C) along with single inverted comma  
(D) All the Above

**13 The maximum value of constant is 32767**

- (A) Integer      (B) Double      (C) Character      (D) String

**14. Integer Constant in C must have**

- (A) At least one digit      (B) Digits separated by comma  
(C) At least one decimal point      (D) A comma along with digits

**15. Which of the following is not a character constant?**

- (A) 'thank you'      (B) 'enter values of P,N ,R'  
(C) '23.56E-03'      (D) All the Above

**16. If a is an integer variable, a=5/2 will return a value**

- (A) 2.5      (B) 0      (C) 3      (D) 2

**17. If z is a float variable, z=4/2 will return a value**

- (A) 1.5      (B) 2.0      (C) 0      (D) None of the above

**18. What is the value of !0?**

- (A) 1      (B) 0      (C) -1      (D)-5

**19. Address of the variable can be displayed by \_\_\_\_\_ operator.**

- (A) #      (B) \*      (C) &      (D)@

**20. What would be the remainder of 8%10?**

- (A) 8      (B) 0      (C) 10      (D) None above

**21. Addition of two numbers can be performed using .**

- (A) Binary Operator      (B) Arithmetic Operator  
(C) Unary Operator      (D) Relational Operator

**22. What is the result of 16>>2?**

- (A) 16      (B) 2      (C) 4      (D) 8

**23. What is the result of 5 &&2?**

- (A) 1      (B) 0      (C) 2      (D) 5

**24. 48 to 57 is said to be the ascii range for .**

- (A) a to z      (B) A to Z      (C) 65 to 70      (D) 0 to 9

**25. What is the ascii range for a to z letters?**

- (A) 97 to 122      (B) Both A & C      (C) 0 to 9      (D) 90 to 120

**26. Which function is appropriate for accepting a string?**

- (A) gets ()      (B) puts()      (C) getch()      (D) scanf ()

**27. array always ends with a null (\0) character**

- (A) Integer      (B) String      (C) Character      (D) Float

**28. Array elements are stored in**

- (A) Scattered memory location      (B) Sequential memory location

- (C) Distributed location      (D) Both A & C

**29. =f u declare an array without initializing the value to it then it will be set to**

- (A) A null value      (B) Zero      (C) Garbage value      (D) All the Above

**30. ..... are passed as arguments to a function by reference & value**

- (A) Array      (B) Constants      (C) Variables      (D) Both A & C

**31. What is the correct way to declare a pointer?**

- (A) int ptr\*      (B) \*int ptr      (C) int \*ptr      (D) int\_ptr z

**32. n array is a collection of**

- (A) Same data type      (B) Both A & C      (C) Different data type      (D) None of the above

**33. If the elements in the array must be**

- (A) Initialized      (B) of same type      (C) Defined      (D) Verified

**34. A C variable cannot start with**

- (A) an alphabet      (B) a number      (C) a special symbol other than underscore  
(D) Both C and B

**35. Which of these are reasons for using pointers?**

- (A) To manipulate parts of an array  
(B) To refer to keyword such as 'for' and 'if'  
(C) To return more than one value from a function  
(D) To refer to particular programs more conveniently

**36. Which of the following is a Scalar Data type**

- (A) Float      (B) Union      (C) Array      (D) Pointer

**37. Which of the following are tokens in C?**

- (A) Keywords      (B) Variables      (C) Constants      (D) All of the above

**38. Which symbol is used as a statement terminator in C?**

- (A) !      (B) #      (C) -      (D) ;

**39. Which escape character can be used to begin a new line in C?**

- (A) \a      (B) \ b (C) \m (D) \n

**40. Which escape character can be used to beep from speaker in C?**

- (A) \a (B) \b (C) \m\ (D) \n

**41. Character constants should be enclosed between \_**

- (A) Single quotes      (B) Double quotes      (C) Both a and b      (D) None of these

**42. String constants should be enclosed between \_**

- (A) Single quotes      (B) Double quotes      (C) Both a and b      (D) None of these

**44. The maximum length of a variable in C is \_ characters.**

- (A) 8      (B) 16      (C) 32      (D) 64

**45. What will be the maximum size float variable?**

- (A) 2 byte      (B) 4 byte      (C) 8 byte      (D) 16 byte

**46. What will be the maximum size double variable?**

- (A) 2 byte      (B) 4 byte      (C) 8 byte      (D) 16 byte

**47. A declaration float a,b; occupies memory**

- (A) 1 byte      (B) 4 byte      (C) 8 byte      (D) 16 byte

**48. The size of a String variable is**

- (A) 1 byte      (B) 4 byte      (C) 8 byte      (D) None of these

**49. Which of the following is an example of compounded assignment statement?**

- (A) a = 5      (B) a += 5      (C) a = b = c      (D) a = 5

**50. The operator && is an example for \_ operator.**

- (A) Assignment      (B) Increment      (C) Logical      (D) Rational

#### ANSWERS

- |       |       |       |       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1. C  | 2. A  | 3. B  | 4. C  | 5. B  | 6. C  | 7. D  | 8. A  | 9. D  | 10. A |       |
| 11. C | 12. B | 13. A | 14. A | 15. D | 16. D | 17. B |       | 18. A | 19. C | 20. A |
| 21. B | 22. C | 23. A | 24. D | 25. A | 26. A | 27. C |       | 28. B | 29. C | 30. D |
| 31. C | 32. A | 33. B | 34. D | 35. C | 36. C | 37. D | 38. D | 39. D | 40. B |       |
| 41. A | 42. B | 43. D | 44. A | 45. B | 46. C | 47. C |       | 48. D | 49. B | 50. C |

**1. The operator & is used for**

- (A) Bitwise AND                                     (B) Bitwise OR  
 (C) Logical AND                                     (D) Logical OR
- 2. The operator I can be applied to**  
 (A) integer values                                  (B) float values  
 (C) double values                                  (D) All of these
- 3. The equality operator is represented by**  
 (A) :=    (B) .EQ.  
 (C) =   (D) ==
- 4. Operators have precedence. It is used to know which operator**  
 (A) is most important                              (B) is used first  
 (C) is faster                                       (D) operates on large numbers
- 5. The bitwise AND operator is used for**  
 (A) Masking                                        (B) Comparison                                     (C) Division                                     (D) Shifting bits
- 6. The bitwise OR operator is used to**  
 (A) set the desired bits to 1                    (B) set the desired bits to 0  
 (C) divide numbers                                (D) multiply numbers
- 7. Which of the following operator has the highest precedence?**  
 (A) \*    (B) ==   (C) ->   (D) +
- 8. The associativity of! operator is**  
 (A) Right to Left                                (B) Left to Right  
 (C) for Arithmetic and (b) for Relational    (D) for Relational and (b) for Arithmetic
- 9. Which operator has the lowest priority?**  
 (A) \*    (B) !   (C) ++   (D) +
- 10. Integer Division results in**  
 (A) Rounding the fractional part                (B) Truncating the fractional part  
 (C) Floating value                                (D) An Error is generated
- 11. The type cast operator is**  
 (A) (type)                                        (B) cast ()                                     (C) II   (D) " "
- 12. Explicit type conversion is known as**  
 (A) Casting                                        (B) Conversion                                (C) Disjunction                                   (D) Separation
- 13. The operator + in a+=4 means**  
 (A) a = a + 4                                    (B) a + 4 = a                               (C) a = 4   (D) a = 4 + 4
- 14. p++ executes faster than p+ 1 because**  
 (A) p uses registers                              (B) p++ is a single instruction  
 (C) ++ is faster than +                           (D) None of these
- 15. Header files in C contain**  
 (A) Compiler commands                            (B) Library functions  
 (C) Header information ofC programs           (D) Operators for files
- 16. Which pair of functions below are used for single character IO.**  
 (A) getchar ()and putchar ()                    (B) gets () and puts ()  
 (C) scanf() and printf()                        (D) fgets () and fputs 0
- 17. The output of printf ("%u", -1) is**  
 (A) -1    (B) minimum int value                           (C) maximum int value                   (D) Error message
- 18. An Ampersand (&) before the name of a variable denotes .....**  
 (A) Actual value                                (B) Address                                   (C) Variable value                           (D) Data type
- 19. Symbolic constants can be defined using**  
 (A) #define                                      (B) constS                                   (C) symbols                                   (D) None of these
- 20. Null character is represented by**  
 (A) \N    (B) \0                                       (C) \0   (D) \e
- 21. A statement differs from expression by terminating with a**

- (A) ;                    (B) :                    (C) NULL            (D) .
- 22. Which operator in C is called a ternary operator**  
 (A) ++                    (B) 0                    (C) If .... then        (D) ? :
- 23. The conversion characters '%s' for data input means that the data item is**  
 (A) An unsigned decimal integer                    (B) A hexadecimal integer  
 (C) A short integear                                (D) A string followed by white space
- 24. An expression contains relational, assignment and arithmetic operators. If Parenthesis are not present, the order will be**  
 (A) Assignment, relational, Arithmetic            (B) Relational, arithmetic, assignment  
 (C) Assignment, arithmetic, relational            (D) Arithmetic, relational, assignment
- 25. Which of the following is a key word is used for a storage class**  
 (A)printf                    (B)auto                    (C)external                    (D)scanf
- 26. In the C language 'a' represents**  
 (A) A Character            (B) An integer            (C) A digit                    (D) A word
- 27. The number of the relational operators in the C language is**  
 (A) Four                    (B) Six                    (C) Three                    (D) Done
- 28. In C, a Union is**  
 (A) memory store            (B) memory screen        (C) memory location                    (D) None
- 29. A multidimensional array can be expressed in terms of**  
 (A) Array of pointers rather than as pointers to a group of continuous array  
 (B) Array without the group of continuous array  
 (C) Data type array  
 (D) None of these
- 30. C allows arrays of greater than two dimensions, who will determined this**  
 (A) Parameter                    (B) Compiler                    (C)                            (D) None of these  
 Programmer
- 31. A pointer to a pointer in a form of**  
 (A) Multiple indirection                    (B) A chain of pointers  
 (C) Both A & B                                        (D) None of these
- 32. Pointers are of**  
 (A) Integer data type                    (B) Unsigned integer data type  
 (C) Character data type                    (D) None of these
- 33. Maximum number of elements in the array declaration int a[5] [8] is**  
 (A) 28                    (B) 32                    (C) 35                    (D) 40
- 34. If the size of the array is less than the number of initialises then,**  
 (A) Extra values are being ignored                    (B) Generates an error message  
 (C) Size of array is increased                            (D) Size is neglected when values are given
- 35. Array subscripts in C always start at**  
 (A) -1                    (B) 0                            (C) 1                            (D) Value provided by the user
- 36. A Structure**  
 (A) Cannot be read as a single entity                    (B) Can be read as a single entity  
 (C) Can be displayed as a single entity  
 (D) has member variables that cannot be read individually
- 37. Identify the invalid pointer arithmetic**  
 (A) Addition of float value to a pointer  
 (B) Comparison of pointers that do not point to the element of the same array  
 (C) Subtracting an integer from a pointer  
 (D) Assigning the value 0 to a pointer variable
- 38. An identifier cannot start with**  
 (A) #                    (B) \_                            (C) Uppercase alphabet                    (D) Lowercase alphabet

- 39.** Symbolic constants are defined as -  
 (A) # define sl s2    (B) #define sl s2;    (C) #define sl = s2    (D) #define sl = s2;
- 40.** An escape sequence commences with -  
 (A) \    (B) /    (C) #    (D)?
- 41.** Identify the wrong declaration  
 (A) int n = (7);    (B) char c2 = 'A' + 25, c1 = 'z';    (C) int a =10,b = 20,c;    (D) int x = 10, y = x\*20, year;  
 (e) None of above
- 42.** Where does execution of every C program starts?  
 (A) main ()    (B) begin ()    (C) start ()    (D) init ()
- 43.** Which operator is not used in C.  
 (A) \*\*    (B) ~    (C) %    (D) 1\
- 44.** The operator % can be applied only to  
 (A) Integral values    (B) Float and double value    (C) Char value    (D) All of these
- 45.** Identify the relational operator  
 (A) !    (B) >    (C) "    (D)&&
- 46.** Which operator has highest priority?  
 (A) ++    (B) +    (C) %    (D)/
- 47.** In C how is logical AND represents?  
 (A) ||    (B) AND    (C) &&    (D)@@
- 48.** If the value of a = 10 and b = -1, the value of x after executing the following expression is  
 $x = (a != 10) \&\& (b=1)$   
 (A) 0    (B) 1    (C) -1    (D) 10
- 49.** How many main o function can be define in a C program?  
 (A) 1    (B) 2    (C) 3    (D) Any number of times
- 50.** int z, x=5, y=10,a=4, b=2;  
 $z = x++ - --y*b/a;$   
 What will be value of z in above sample code?  
 (A) 5    (B) 10    (C) 11    (D) 1

**ANSWERS**

1. A    2. D    3. D    4. B    5. A    6.A    7. C    8. A    9. D    10. B    11. A    12. A    13. A  
 14. B    15. B    16. A    17. C    18. B    19. B    20. B    21. A    22. D    23. D    24. D    25. B  
 26.A    27. B    28. A    29. A    30. B    31. C    32. D    33. D    34. B    35. B    36.A    37. A  
 38. A    39. A    40. A    41. E    42. A    43. A    44. A    45. B    46.A    47. C    48. A    49. A  
 50. D

- 4.** The tab is represented by which escape sequence?  
 (A) \t    (C) \n    (B) %d    (D) None above
- 5.** Which of the variables can have many declarations but only one definition?  
 (A) Local variable    (B) Global variable    (C) Static variable    (D) All the above
- 6.** Which function gets execute as we execute a 'C' program?  
 (A) Printf ()    (B) Main ()    (C) MAIN ()    (D) main ()
- 7.** The variables can be initialized by  
 (A) Decrement operator(--)    (B) Both A & C  
 (C) Equal to(=)    (D) Less than equal to(≤)
- 8.** An----- integer variable values greater than or equal to zero  
 (A) Unsigned    (B) Long    (C) Signed    (D) All the above
- 9.** ..... are said to be user defined names.  
 (A) Constants    (B) Identifiers    (C) Keywords    (D) Header files

- 10.** In C every----- has a type, a name, & a value  
 (A) Keywords      (B) Function      (C) Header files      (D) Variable
- 11.** Which of the data type has the range -128 to 127?  
 (A) Integer      (B) Double      (C) Character      (D) Float
- 12.** The C program cannot start with a.....  
 (A) Number      (B) Hyphen( - )      (C) Spaces      (D) All the Above
- 13.** In one statement of C how many variables can be declared?  
 (A) Any no. of variables      (B) One variable  
 (C) Ten variables      (D) Two variables
- 14.** The value of the variable can be kept variant by using which keyword?  
 (A) Constant      (B) Volatile      (C) Private      (D) Public
- 15.** Which of the following is the incorrect keyword name?  
 (A) Char      (B) Printf      (C) else      (D) Both A & B
- 16.** void \*ptr;  
**myStruct myArray[10];**  
**ptr = my Array;**  
 Which of the correct way to to increment the variable ptr?  
 (A) Ptr = ptr + sizeof(myStruct);      (B) ++(int\*)ptr;  
 (C) Ptr = ptr + sizeof(myArray);      (D) Increment(ptr); ptr= ptr + siazef (ptr)
- 17.** "My salary was increased by 15 %!" Which of following statement will produce exact statement?  
 (A) printf (" \ " My salary was increased by 15%!\ \"\n");  
 (B) printf ("My salary was increased by 15%! \n");  
 (C) printf ("My salary was increased by 15%' ! \ n");  
 (D) printf(""\My salary was increased by 15 \ % \ ! \"\n");
- 18.** What is difference between a declaration and a definition of variable?  
 (A) Both can occur multiple times but declaration can occur only once  
 (B) A declaration can occur once, but definition can occur many times  
 (C) There is no difference between them  
 (D) A definition occurs once, but declaration can occur many times
- 19.** int testarray[3] [2] = {1, 2, 3, 4, 5,6,7 ,8, 9,10, 11, 12};  
 What is value of testarray[2][1][0]?  
 (A) 3      (B) 1      (C) 9      (D) 7
- 20.** int a=10,b;  
 b=a++ + ++a ;  
 printf ("%d, %d, %d, %d",b,a++ ,a,++a) ;  
 What is output of above code?  
 (A) 12,10,11,13    (B) 22,10,11,13  
 (C) 12,11,11,11    (D) 22,13,13,13
- 21.** int x[] = {1,4,8,5,1,4};  
 int \*ptr,y;  
 ptr = x + 4;  
 y = ptr -x;  
 What does y in sample code above equal?  
 (A) -3      (B) 0      (C) 4      (D) 4 + sizeof (int)
- 22.** 11^5 What does operation produce?  
 (A) 1      (B) 14      (C) 6      (D) 8
- 23.** #define MAX\_NUM 15  
 Referring to the sample above what is MAX\_NUM?  
 (A) MAX\_NUM is an integer variable      (B) MAX\_NUM is an integer constant

## Answers :

4.A      5.B      6.D      7.C      8.A      9.B      10.D      11.C      12.D      13.A      14.B  
15.D     16.A     17.D     18.B     19.C     20.D     21.C     22.B     23.D     24.B     25.A  
26.C     27.B     28.A

- 1.** What action is exactly performed when the prototype of the function is mentioned?  
(A) Defining it      (B) Call to the function    (c) Declaring it      (D) None of the above
  - 2.** In which of the case the default statement "all the case statement are false "is executed?  
A. For               B. Switch           C. Do while           D. If else
  - 3.** Each case statement in which switch () is separated by.....  
A. Exit               B. Continue           C. Break               D. Goto
  - 4.** The keyword 'else' can be used with.....  
A. Do while()       B. Switch() statement    C. For()               D. If statement
  - 5.** We can select one task switch between the function in a program by using .....statement  
A. While() statement      B. Switch() statement  
C. Do while () statement    D. If statement
  - 6.** The Positive value is Display by which function?  
A. abs()               B. s               C. pow()               D. None of the above
  - 7.** Recursion is the process in which a function calls.....  
A. itself               B. Another function    C. main() function      D. None of the above
  - 8.** If there are too many recursive calls it may result to.....  
A. Memory overflow    B. Stack overflow      C. Queue Overflow     D. All the above
  - 9.** In C which of the function can't be call recursively?  
A. main() function      B. private Function    C. Public function      D. None of the above
  - 10.** 'break' statement is used to exit from:  
A. an if statement      B. a for loop      C. a program              D. the main() function

- 11.** The control statement that allows us to make a decision from number of choice is called  
 A .structure      B. switch statement      C. if loop      D. for loop
- 12.** Which header file is essential for using strcmp() function?  
 A. text.h      B. strcmp.h      C. strings.h      D. string.h
- 13.** malloc() function used in dynamic allocation is available in which header file?  
 A. stdlib.h      B. conio.h      C. stdio.h      D. mem.h
- 14.** File manipulation function in C are available in which header file?  
 A. files.h      B. streams.h      C. stdio.h      D. stdlib.h
- 15.** C support how many basic looping constructs  
 A. 2      B. 3      C. 4      D. 5
- 16.** What should be the expression return value for a do-while to terminate  
 A. 1      B. 0      C. -1      D. NULL
- 17.** Which among the following is a unconditional control structure  
 A. Do-while      B. if-else      C. goto      D. for
- 18.** continue statement is used for  
 A. to go to next iteration in the loop      B. come out of loop  
 C. exit and return to main function      D. Restart interations beginning from loop
- 19.** Which of following header file is required for strcpy() function?  
 A. strings.h      B. strcpy.h      C. files.h      D. string.h
- 20.** A compound statement is a group of statements included between a pair of  
 A. Double quotes      B. Parenthesis      C pair of /'s      D. Curly braces
- 21.** A link is  
 A. A computer      B. A C interpreter  
 C. An active debugger      D. An analysing tool
- 22.** The continue command cannot be used with  
 A. switch      B. while      C. do      D. for
- 23.** When the main function is called, it is called with the arguments  
 A. argc      B. argv      C. Both A & B      D. None of these
- 24.** Parameters are used-  
 A. To return values from the called function  
 B. To send values to the called function  
 C. A & B both  
 D. To specify return type of function
- 25.** Recursive call result when  
 A. A function calls itself  
 B. A function calls another function, which in turn call the function  
 C. A & B both  
 D. A function call another function
- 26.** The main function calls in a C program  
 A. Allows recursive calls      B. does not allows recursive calls  
 C. Is built in function      D. Is optional
- 27.** With every use of memory allocation function, which function should be used to release allocated memory which is no longer needed?  
 A. unalloc()      B. free()      C. dealloc()      D. release()
- 28.** char\*myFunction(char\*ptr)  
 {  
 ptr +=3;  
 returnptr;  
 }

```

int main()
{
char *x, *y;
x =":ello"
y = myFunction(x);
printf("y = %s" y)
}

```

**What will be output of program?**

- A. y= Hello
- B)y= ello
- C.y= llo
- D)y= lo

29. void myFunc(int x)

```

{ if(x>4)
myFunct(--x);
printf("%d " x)
}
int main()
{
myFunc(5);
return 0;
}

```

**What will be code print?**

- A. 0,0,1,2,3,4
- B.4,3,2,1,0,0
- C.1,2,3,4,5,5
- D.0,1,2,3,4,5

30. Which function is correct choice for moving binary data that are of arbitrary size and position in memory?

- A. memcpy()
- B. Strncpy()
- C. memset()
- D. memmove()

31. Which one of following provides conceptual support for function call?

- A. The system stack
- B.The data segment
- C. The processors registers
- D. The text segment

32. int i,j:

```

int ctr = 0;
int myArray[2][3];
for(i=0;i<3;i++)
    for(j=0;j<2;j++)
    {
        myArray[j][i]=ctr;
        ctr++;
    }

```

**What is value of myArray[1][2]?**

- A. 2
- B. 3
- C. 1
- D. 5

33. int x=3;
if(x==2)
 x =0;
if(x==3)
 x++;
else
 x+=2;

**What will be the value of x?**

- A. 1
- B. 3
- C. 4
- D. 5

```
34. x=3,counter=0;
    while((x-1))
    {
        ++counter;
        x--;
    }
```

**What will be value of counter?**

- A. 0    B. 1    C. 2    D. 5

```
35. void (*signal(int sig,
    void(*handler)(int)))(int);
```

**Which one of the following definitions of sighandler\_t allows the above declaration to be rewritten as below:**

```
sighandel_t signal(int sig, sighandler_t handler);
```

- A. typedef void\*sighandler\_t(int);
- B. typedef sighandler\_t  
void(\*)(int);
- C. #define sighandler\_t  
void(\*)(int);
- D. Typedef  
void(\*sighandler\_t)(int);

```
36. struct customer *ptr =
    malloc(sizeof(struct customer));
```

**Give then sample allocation for the pointer found above, which of the following statement is used to reallocptr to be an array of 10 elements?**

- A. ptr+=malloc(9\*sizeof(struct customer));
- B. realloc(ptr, 10 \*sizeof(struct customer));
- C. ptr= realloc(ptr, 10\*sizeof(struct customer));
- D. relloc(ptr, sizeof(struct customer));

```
37. short testarray[4][3] = {{1},{2,3},{4,5,6}};
```

```
printf("%d\n" sizeof(testarray))
```

**What will be output assuming short need 3 bytes?**

- A. 6
- B. 7
- C. It will not compile since not enough initializations
- D. 24

```
38. char buf*+ = "hello World!"
```

```
char*buf= "hellow World!"
```

**in term of code generation do above initializations differ?**

- A. The first definition certainly allow the contents to be safely modified at runtime; the second does not
- B. They do not differ they are functionally equivalent
- C. The first definition is not legal because it does not indicate the size of array to be allocated, the second one is legal.
- D. The first definition allocates enough space for a NUL\_ character, nor does it append one; he second definition does.

**39. In a c expression, how logical AND operator represented?**

- A. &&
- B. @@
- C. ||
- D. AND

**40. :ow to print()'s format specifier %e and %f differ in their treatment of floating number?**

- A. %e display a double in engineering notation if the number is very small or very large. Otherwise it behaves like %f and displays numbers in decimal notation

- B. %e display a argument of type double with trailing zeros and %f never displays trailing zeros
- C. %f and %e both expect a corresponding argument of type double and format it identically. %e is left over from K && C; standard C prefers %f for new code
- D. %e always display and argument type double in engineering notation %f always displays ad argument of type double in decimal point

**41. Which one of the following will read a character from the keyboard and will store it in the variable C?**

- A. c = getch();    B. C = getchar();    C. c = getchar(stdin);    D. getchar(&c);

**42. #include<stdio.h>**

```
int I;
void increment(int i)
{
    i++;
}
int main()
{
    for(i=0;i<10;increment(i))
        printf("i=%d\n" i)
return 0;
}
```

**what is output of above program?**

- A. i=9;
- B. it will not compile
- C. i=10
- D. it will loop indefinitely

**43. int i=4;**

```
switch(i)
{
    default: ;
    case 3:
        i+=5;
        if(i==8)
        {
            i++;
        if(i==9)break;
        i*=2;
        }
        i-=4;
        break;
    case 8:
        i+=5;
        break;
}
```

**printf("i=%d\n" i)**

**what will be output of the sample code above be?**

- A. i=5;    B.i=9;    C. i=10;    D.i=18;

**44. Which one of the following operators is right associator ?**

- A. ->
- B. []

C. =

D. ,

**45. What does the auto specifier do?**

- A. It automatically increment the variable when used
- B. It indicate that a variable memory will automatically be preserved
- C. It automatically initialised a variable to 0
- D. It indicate that a variables memory space is allocated upon entry in to a block

**46. How do you include a system header file called sys header .h in C source file?**

- A. #incl<sysheader.h>
- B. #includefile<sysheader>
- C. #includesysheader.h
- D. #include<sysheader.h>

**47. Which one of following printf(format specifier indicates to print double value in decimal notation, left align in a 30 character forrmat field 4 digit precision?**

- A. %30f.4e
- B. %4.30f
- C. %30.4f
- D. %4.30e

**48.**  
int x=0;  
for(;;)  
{  
    if(x++=4) break;  
    continue;  
}  
printf("x=%d\n" x)

**what is the output of above code?**

- A. x=5
- B. x=0
- C. x=4
- D. x=1

**49. According to the standard C specification what are the respective minimum size of following three data types :short,int,long?**

- A. 1,2,2
- B. 1,2,4
- C. 2,4,8
- D. 2 ,2,4

**50. What is output of following code?**

```
#include<stdio.h>
void main()
{
    char letter='Z'
    printf("\n %c" letter)
}
```

- A. 90
- B. Z
- C. Error

**D. Garbage Value****Answers:**

1. C	2. B	3. C	4. D	5. B	6. A	7. A	8. B	9. A	10. B
11. B	12. D	13. A	14. C	15. B	16. B	17. C	18. A	19. D	20. D
21. D	22. A	23. C	24. B	25. A	26. A	27. B	28. D	29. D	30. D
31. A	32. D	33. C	34. C	35. D	36. C	37. D	38. B	39. A	40. D
41. B	42. D	43. A	44. C	45. B	46. D	47. C	48. A	49. B	50. B

**1. What will be output of following program?**

```
#include <stdio.h>
#define a 10
void main()
{
    printf("%d    a)
    foo();
    printf("%d .. "
,a);
}
void foo()
{
#define a
#define a 50
}
```

(A) 10.10                    (B) 10.15                    (C) Error                    (D) 0

**2. Array is passed as an argument to a function is interpreted as**

- (A) Address of array                    (B) Number of elements in array  
     (C) Value of the first element in array            (D) Address of the first element of

**3.**

```
main()
{
    char thought [20] [30] = {"Don't walk in front of me .. ", "1 may not follow" };
    printf("%c%c", * (th ought [0]+9), *(thou ght+0)+5));
}
```

**What is output of program?**

- (A) Don't walk in front of me                    (B) kk                            (C) 1 may not follow            (D) K

**4.**

```
#include <stdio.h>
void main()
{
    int i=3, *j, **k;
    j = &i;
    k=&j;
    printf("%d%d %d", *j, **k, *(*k));
}
```

**What is output of above code?**

- (A) 000                            (B) 333                            (C) 444                            (D) 433

**5. Which of the following is the correct way of declaring a float pointer?**

- (A) float ptr;      (B) \*float ptr;      (C) float \*ptr;      (D) None

**6. The reason for using a pointer is ....**

- (A) Accessing arrays or strings      (B) Dynamic memory allocation  
 (C) Implementing Linked lists, trees, graphs and many other data structures  
 (D) All the above

**7. The size of structure can be determined by**

- a. Size of variable name  
 b. Size of (structure tag)  
 (A) Only a      (B) Only b      (C) Both a and b      (D) None

**8. main()**

```
{
    Struct
```

```
    {
        int i;
    }xyz;
(*xyz)->i=10;
printf("%d",xyz.i);
```

What will be the output?

- (A) Program will not compile      (B) No answer      (C) 10      (D) Address of i

**9. Pushdown list means:**

- (A) Stack      (B) Queue      (C) Linked List      (D) All the Above

**10. What output following program produce?**

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

```
main()
{
char str*+="S\005 B"
printf(" \n %d",sizeof(str));
}
```

- (A) 7      (B)      (C) 5      (D) Error

**11. fputs function is used to**

- |                              |                            |
|------------------------------|----------------------------|
| 1. write character to a file | 2. takes 2 parameters      |
| 3. returns a character       | 4. requires a file pointer |
- (A) All are true      (B) All are false  
 (C) Only 1 and 2 are true      (D) Only 1 and 3 are true

**12. #include<stdio.h>**

```
Void main()
```

```
{
    int a;
    print("%d" a^a)
}
```

- (A) 1      (B)0      (C)Unexpected      (D)Runtime error

**13. Time taken for addition of element in queue is**

- (A) O(1)      (B) O(logn)      (C) O(n)      (D) None of these

**14. To delete a dynamically allocated array names 'a' , the correct statement is**

- (A) delete a[10];      (B) delete []      (C) delete a;      (D) delete [0]a;

**15.What is output of following code?**

```
#include<stdio.h>
void swap(int&,int&);
void main()
{
    int a=10,b=20; swap(a++,b++ );
}
void swap(int&x,int&y)
{
    x+=2; y+=3;
}
```

(A) Error                    (B) 10,20  
 (C) 14,24                    (D) 11,21

**16.What will be value of 'a' after following code is executed?**

```
#define square(x) x*x
A = square(2+3);

```

(A) 25                    (B) 13                    (C) 11                    (D) 10

**17.The five items: A, B, C, D and E are pushed in a stack, one after another starting from A. The stack is popped four times and each element is inserted into queue. Then two elements are deleted from the queue and pushed back to stack. Now one item is popped from the stack. The popped item is**

- (A) A                    (B) B  
 (C) C                    (D) D

**18.What is output of following code?**

```
#include<stdio.h>
Void main()
{
    int a=0,b=0;
    a=(b=75)+9;
    printf(" \ n%d%d",a,b);
}
```

(A) 75,9                    (B) 84,75                    (C) 75,84                    (D) None

**19. When applied to a variable, what does the unary & operator means?**

- (A) The variable value                    (B) The variable format  
 (C) The variable address                    (D) The variable's right value

**20. FILE**

```
*f=fopen(fname, "r"");readData(f);if(? ???){puts("End of file reached");}
(A) F=EOF()
(B) eof(f);
(C) feof(f)
(D) f=NULL
```

**21. Global variables that are declared static are----- .**

Which one of the following correctly completed the sentence above?

- (A) Deprecated by standard C                    (B) Allocated to Heap  
 (C) Internal to the current translation unit                    (D) Visible to all translation units

**22. According to standard C, what type of an unsuffixed floating point literal, such as 123.45?**

- (A) float                    (B) double                    (C) unspecified                    (D) long double

**23.** Which one of the following valid for opening a read-only ASCII file?

- (A) fileOpen(fname,"r"); (B) fileOpen(fname,"ra");  
 (C) fopen(fname,"r"); (D) fileOpen(fname,"read");

**24.** f = fopen(fname,"r"); Referring to the code, what is proper definition of variable f from following?

- (A) FILE f; (B) File \*f; (C) int f; (D) struct FILE f;

**25.** short int x; // x is 16 bits

What is the maximum number that can be printed using printf("%d\n",x)

- (A) 127 (B) 128 (C) 65,536 (D) 32,767

**26.** char \*dwarves[]={"Sleppu","Dopey","Doc","happy","Grumpy","sneezy","Bashful",}; how many element will dwarves will contain? Assuming c compiler employed strictly with the requirements of standard C.

- (A) 4 (B) 5 (C) 6 (D) 7

**27** char \*buffer = 0123456789"

char \*ptr = buffer;

ptr +=5;

printf("%s \n",ptr);

printf("%s\n", buffer);

What will be printed when above code is, executed?

- (A) 0123456789,56789 (B) 5123456789,5123456789  
 (C) 56789,0123456789 (D) 56789,56789

**28.** int y[4]={6,7,8,9}; int

\*ptr= z +2;

printf("%d \ n,ptr[l]);

//ptr+ l=ptr[l]

- (A) 6 (B) 7 (C) 8 (D) 9

**29.** Penny = one

Nickel = five

Dime = ten

Quarter = twentyfive

How is enum is used to define the values of the American coins listed above?

(A) enum

(B) enum

coin( (penny, 1),(nickel,5),

coin(penny= 1,nickel=5,

(dime, 10),(quarter,25);

dime=10,quarte=25);

(D) enum

coin(penny=1,nickel=5,

coin{penny,nickel, dime=10,quarter=25}

dime,quarter}(1 ,5, 1 0,25);

**30.** char txt[20] = "Hello World \ 0";

How many bytes are allocated for above definition?

- A) 11 B) 12 C) 21 D) 20

**31.** int i=4;

int x=6;

double z;

```

z=x/l;
printf("z = %.2f\n",z);
What will above code print?
(A) Z=1.00      (B) Z=1.50
(C) Z=0.00      (D) Z=2.00

```

**32. Which of following variable name is not valid?**

- (A) go\_cart      (B) 4season      (C) run4      (D) \_what

**33. long factorial(long x)**

{

????

```

return x*factorial(x-1);
}

```

What should replace ????to make function return correct result?

- (A) If(x==0) return 0;      (B) If(x>=2) return 2 ;  
 (C) If(x<=1) return 1 ;      (D) If(x==0) return 1;

**34. How variable is accessed from other file?**

- (A) The global variable is referenced via the extern specifier  
 (B) The global variable is referenced via the pointer specifier.  
 (C) The global variable is referenced via the global specifier.  
 (D) The global variable is referenced via auto specifier.

**35. What number is equivalent to 4e3?**

- (A) 40      (B) 0.004      (C) 400      (D) 4000

**36. How does a variable definition differ from declaration?**

- (A) Variables may be defined many times but declared only once  
 (B) Definition allocates storage for a variable, but declaration only informs the compiler the type.  
 (C) Variable definition must be preceded by variable declaration  
 (D) There is no difference in C between variable declaration and definition.

```

37. int x[] ={1, 2, 3, 4, 5}
int u;
int *ptr = x;
?????????
for (u=0,u<5;u++)
{
    printf("%d-", x*u++)
}
printf("\n")

```

Which one of following will replace the ?????in the code above to cause string 1-2-3-10-5- to be printed?

- (A) \*ptr + 3 = 10      (B) \*ptr[3] =10  
 (C) \*(ptr+3) = 10      (D) (\*ptr)[3] = 10

**38. #include <stdio.h>**

```

Void func()
int x=0;
static int y=0;

```

```

x++,y++;
printf("%d%d \ n" ,x,y);
}
void main()
{
    Func();
    Func();
    return 0;
}

```

What will be output of the above code?

- (A) 1-1            (B) 1-1  
     2-1        1-2  
   (C) 1-1            (D) 1-0  
   1-2        1-0;

**39. Except 1 all choices are O.K. c = getchar();**

What is the proper declaration for the variable in the code above?

- (A) Unsigned int c;     (B) Unsigned char c;     (C) int c;     (D) char c;

**40. When did the first ANSI come out?**

- (A) 1949     (B) 1975     (C) 1958     (D) 1966

**41. Which of following is not standard issuing body?**

- (A) X3     (B) ISO     (C) BSI     (D) ANSI

**42. List in chronological order, when these languages officially recognized as a standard.**

1.     ANSIC
  2.     ANSI COMMON LISP
  3.     ANSI COBOL
  4.     ANSI ADA
- (A) 1,2,3,4     (D) 1,3,2,4  
   (C) 4,3,1,2     (D) 1,3,42

**43. What are standards for?**

1.     To provide uniformity for everyone.
2.     To allow monopoly of the product in the industry
3.     To define a level of quality others have to meet
4.     To restrict unauthorized changes in a design and its development

(A)

2 and 4

- (B) 1 and 3  
   (C) 1,2 and 4 only  
   (D) None of above

**44. What implementation of C++ makes C++ programming language powerful?**

- (A) Easy implementation  
   (B) Reusing of code  
   (C) Easy memory management  
   (D) All the above

**45. What are the main differences between 3rd and 4th generation languages?**

- I. Both follow procedural code
  - II. Third generation language are mostly compiled languages.
  - III. Fourth generation languages are in-line with minimum work and skill concept
  - IV. Third generation languages are user friendly and have intelligent default option.
- (A) ii and iii only  
   (B) i and iv only

- (C) i and iii only  
 (D) None

**46. Which of following features would make next generation PL popular?**

- I. They are highly portable and are offered over wide range of systems
  - II. They are suitable for development of programs of arbitrary size and complexity.
  - III. They are reasonably stable during changes in hardware and system software.
  - IV. They have both procedural and nonprocedural approach.
- (A) ii and iii only      (B) I, iii and iv only  
 (C) iii and iv only      (D) all

**47. Which of following language has potential to become the next programming language?**

- I. Java      II. Html
  - III. COBOL97      IV. ADA95
- (A) I and N only    (B) I AND II only  
 (C) I and III only (D) NONE

**48. #define max 10 +2**

```
void main()
{
int i;
i = max*max;
clrscr();
printf("%d",i);
getch();
}
What will be the output?
(A) 32      (B) 60      (C) 12      (D) 19
```

**49.What will be the output?**

```
Void main()
{
char *str1='powla'
char *str2='er'
clrscr();
printf("%s \ b \ b%s",str1,str2);
}
(A) powlaer    (B) powler      (C) power      (D) None
```

**50. What will be output?**

```
Void main()
{
int a=270;
char *p;
p=(char *)&a;
clrscr();
printf("%d", *p);
getch();
}
(A) 200      (C) 14      (B) 16      (D) 15
```

**Answer :**

1. C	2. D	3.D	4.B	5. C	6. D	7.C	8.A	9.A	10.C	11.D	12. B	13.B
14.C	15.A	16.B	17.D	18.B	19. C	20.A	21. B	22. A	23.C	24.B	25. D	26.D
27.C	28.D	29. B	30. D	31. A	32. B	33.C	34.A	35.D	36.A	37.C	38.B	
39.D	40. C	41. A	42. C	43.B	44.D	45.A	46.D	47.C	48.A	49.C	50.C	

**1. what is the output of following?**

```
void main()
{
    int a=5;
}
clrscr();
printf("%d" a)
getch();
}
(A) 7           B) 5           c) 8           D) 6
```

**2. what will be the output?**

```
void main()
{
    int a=5;
    {
        int a=7; a++;
        printf("%d" a)
    }
    clrscr();
    printf("%d" a)
    getch();
}
(A) 5           because the scope of variable int a=7 is close after a++ and printf stateme.
(B) 7
(C) 8
(D) None
```

**3. What is output?**

```
void change(int const*p)
{
    *((int *)p) = 20;
}
void main()
{
    int const x=10;
    change(&x);
    clrscr();
    printf("%d",x);
    getch(); }
```

(A) 20 (B) 10 (C) 15 (D) 40

**4. What is output?**

```
void main()
{
    int a=1;
    static int count;
    clrscr();
    count++;
    while(a)
    {
        count++;
        a&=a-1;
    }
    printf("%d",count);
    getch();
}
```

- (A) 20    (B) 1    (C) 16    (D) None

**5. What will be the output?**

```
void main()
{
    int array[]={1,2,3,4,5,6};
    void xxx(int[5]);
    xxx(arr);
    getch();
}
void xxx(int ch[5])
{
    clrscr();
    printf("%d",1 [ch]);
}
```

- (A) 20    (B)1    (C) 16    (D) None

**6. find(int x, int y)**

```
{
    return((x<y):0:(xy));
```

**call using find(x,find(x,y));**

The purpose of the code is to find

- (A) Maximum of x and y
- (B) Minimum of x,y
- (C) Positive difference between x and y
- (D) Sum of x and y

**7. Integer needs 2 bytes, the maximum value it can hold as unsigned is?**

- |                    |                    |
|--------------------|--------------------|
| (A) (2 power 16)-1 | (B) (2 power 15)-1 |
| (C) (2 power 16)   | (D) (2 power 15)   |

**8. Expression  $3*(y-8)/9$  and  $(y-8/9)*3$  yields same value and y is of integer then y**

- |                                     |                                |
|-------------------------------------|--------------------------------|
| Must yield same value               | (B) Must yield different value |
| (C) May or may not yield same value | (D) None                       |

(A)

9. `printf("%f",9/5)`  
will print  
(A) 1.8 (B) 1.0 (C) 2.0 (D) None (Error)

10. What will Output of Following Program

```
if(a=7)
    printf("a is 7")
else
    printf("a is not 7")
```

(A) a is 7 (B) is not 7  
(C) Nothing (D) Garbag

11. What will Output of Following Program

```
if(a>b)
    if(b>c)
        s1;
    else
        s2;
```

s2 will be executed if  
(A) b>c  
(B) a<=b  
(C) b<=c and a<=b  
(D) a>b and b<=c

12. What will Output of Following Program

```
void main()
{
    inc();inc();inc();
}
inc()
{
    static int x;
    printf("%d",++x);
}
```

(A) 012 (B) 3  
(C) 123 (D) 111

13. Preprocessing is done

- (A) either before or at beginning of compilation process  
(B) after compilation before execution  
(C) after loading  
(D) None

14. `printf("%d",sizeof(""));` will print

- (A) 1 (B) Error  
(C) 0 (D) Garbage

15. What will Output of Following Program

```
main()
{
    int a=5,b=2;
```

```

    printf("%d", a++ + b);
}
(A) results in syntax   (B) 8
(C) 7                 (D) None

```

**16. The process by which one bit pattern is converted in to another by bit wise operation**

- (A) Masking
- (B) Pruning
- (C) Bitting .
- (D) Chopping

**17. Value of automatic variable that is declared but not initialized will be**

- (A) 0
- (B) 1
- (C) Unpredictable
- (D) None

**18. int v=3, \*pv=&v; printf("%d%d",v,\*pv);  
output will**

- be
- (B) 3 3
  - (A) Error
  - (D) 3 address of v
  - (C) None

**19. declaration**

```

enum
cities(Bethlehem,J ericho,N azareth =1,jeruslem)
assign value 1 to
(A) Bethlehem      (B) nazareth
(C) Bethlehem and nazareth
(D) Jerich() and nazareth

```

**21. Consider scanf and sscanf function which is true**

- (A) no standard function called sscanf
- (B) sscanf input character are taken from string
- (C) sscanf is equivalent to scanf
- (D) None of above

**22. int x[3][4] = { {1,2,3},{4,5,6},{7,8,9}}**

**zero value will be present at**

- (A) x[2][2] = x[2][2] = x[2][3] =0
- (B) None
- (C) Value in last row is zeor
- (D) Value in fourth column is zero

**23.**

```

main()
{
printf("%u", main());
}

```

- (A) prints starting address of main()
- (B) prints garbage
- (C) infinite loop
- (D) Execution error

**24. int a, \*b=&a, \*\*c=&b;**

```

-----
-----
a=4; **c=5;
(A) Does not change value of a
(B) Assigns 5 to a
(C) Assigns value orb to a
(D) Assigns address of c to a

```

**25. What is o/p**

i=5;

```
i=(++i)/(i++);
printf("%d",i)
(A) 2    (C) 5
(B) 6    (D) 1
```

**26. What is o/p**

```
void main()
{
int const *p=5;
printf("%d",++(*p));
}
```

- (A) Compile time error    (B) Run time error  
 (C) Address                (D) 5

**28. main()**

```
{
float me=1.1;
double you=1.1;
if(me==you)
printf("I LOVE YOU");
else
printf("= : TE YOU")
(A) I LOVE YOU
(B) I HATE YOU
(C) Compile error
(D) Run time error
```

**29.**

```
main()
{
char *p;
printf("%d %d" ,sizeof(*p),sizeof(p));
}
```

What will be the output?

- (A) 11    (B) 1 2  
 (C) 22    (1) Cannot tell

**30.**

```
main()
{
static int var=5;
printf("%d",var-- );
if(var)
main();
}
```

What will be the output?

- (A) 1
 (B) Will print 54321
 (C) Compile error
 (D) 5555555

```

31.    main()
{
    int i=3;
    switch(i)
    {
        default: printf("zero");
        case1:   printf("one")
                  break;
        case 2:  printf("two");
                  break;
        case 3:  printf("three");
                  break;
    }
}

```

What will be the output?

- (A) Zero      (B) One
- (C) Three     (D) Two

**32.**

```

main()
{
    int c=-2;
    printf("c=%d",c)
}

```

What will be the output?

- (A) C = 2      (B) C=-2
- (C) Garbage value
- (D) Compile error

**33.**

```

#define int char
main()
{
    int i=65;
    printf("sizeof(i)=%d",sizeof(i));
}

```

- |                 |                   |
|-----------------|-------------------|
| (A) Sizeof(i)=1 | (B) Sizeof(i)=2   |
| (C) Sizeof(i)=3 | (D) Compile error |

**34.**

```

main()
{
    int i=10; i!=i>14 ;
    printf("%d",i);
}

```

- (A) True      (B) False
- (C) 1          (D) 0

**35.**

```

#define sqaure(x) x*x
main()

```

```
{
    int i=64/square( 4);
    printf("%d",i);
}
```

What will be output of program?

- (A) 16     (B) 64
- (C) 4     (D) 32

**36.**

```
#include <stdio.h>
#define a 10
main()
{
    #define a 50
    printf("%d" ,a)
}
```

What will be output of program?

- (A) 50     (B) 10     (C) 40     (D) 60

**37.**

```
main()
{
    int i=10
    printf("%d%d%d" ,a,++a,a++);
}
```

What will be output of program?

- (A) 121211     (B) 121010
- (C) 111112     (D) 101012

**38.**

```
main()
{
    int i=0;
    for(;i<2;)
        printf("%d " ,i++);
}
```

What will be output of program?

- (A) 0 1 2     (B) 0 1 2
- (C) 1 2 3     (D) Compile error

**39.**

```
main()
{
    int x;
    for(x=1;x<=5;x++)
        printf("%d" ,x);
}
```

What will be output of program?

- (A) 1 2 3 4 5     (B) 1
- (C) 5     (D) 6

**40.**

```
main()
{
    int array[]={10,20,30,40};
    printf("%d",sizeof(5.2));
}
```

What will be output of program?

- (A) 2    (B) 4    (C) 8    (D) 10

**41.**

```
main()
{
    int array[]={10,20,30,40};
    printf("%d",-2[array]);
}
```

What will be output of program?

- (A)-60                         (B)-30                         (C) Garbage value             (D) Compile error

**42.**

```
main()
{
    int array[3]={5};
    int i;
    for(i=0;i<2;i++)
        printf("%d",array[i]);
}
```

What will be output of program?

- (A) 5 Garbage value
- 
- (B) 500
- 
- (C) 5 null null
- 
- (D) Compile error

**43.**

```
main()
{
    int a=5;
    int b=10;
    {
        int a=2;
        a++;
        b++;
    }
    printf("%d%d",a,b);
}
```

What will be output of program?

- (A) 510                                 (B) 611
- 
- (C) 5 11                                 (D) 6 10

**44.      main()**

```
{
    int x=2,y=3;
    if(x+y<=5)
        printf('True');
    else
```

```

        printf("False");
    }
What will be output of program?
(A) True      (B) False
(C) Compilation Error
(D) Run time error

```

**45. main()**

```

{
const int i=5;
i++;
printf("%d",i);
}
What will be output of program?
(A) 5      (B) 6
(C) Compile Error
(D) Run time Error

```

- |       |       |       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1. B  | 2. A  | 3. A  | 4. C  | 5. B  | 6. C  | 7. A  | 8. B  | 9. D  | 10. D |
| 11. D | 12. C | 13. A | 14. A | 15. C | 16. A | 17. C | 18. B | 19. D | 20. A |
| 21. B | 22. D | 23. C | 24. B | 25. A | 26. A | 27. A | 28. B | 29. B | 30. A |
| 31. C | 32. D | 33. A | 34. D | 35. B | 36. A | 37. A | 38. B | 39. D | 40. C |
| 41. B | 42. B | 43. C | 44. A | 45. C |       |       |       |       |       |

- 1. Syntax error is -**
  - (A) Compile Time Error
  - (B) Logical Error
  - (C) Run Time Error
  - (D) All above
- 2. In Black Box testing**
  - (A) Tester doesn't look into the internal behavior and functionality of system
  - (B) Testing is done to decide whether or not to accept the product
  - (C) Tester accesses the internal data structure and algorithms
  - (D) Integration of external or third party system is tested.
- 3. Comments are added to -----to understand program logic.**
  - (A) Source code
  - (B) System Manual
  - (C) User Manual
  - (D) None of the above
- 4. Hungarian notation is used to**
  - (A) Define name of variable according its data type and intended use
  - (B) Create System Manual
  - (C) Create User manual
  - (D) All of the above
- 5. In white box testing -**
  - (A) Tester doesn't look into the internal behavior and functionality of system.
  - (B) Testing is done to decide whether or not to accept the product.
  - (C) Tester accesses the internal data structure and algorithms
  - (D) Integration of external or third party system is tested
- 6. ..... is done to eliminate errors of the application.**
  - (A) Compilation
  - (B) Debugging
  - (C) Documentation
  - (D) All above

7. .... is a process of validating the correctness of program.  
(A) Compilation      (B) Debugging  
(C) Testing      (D) Documentation
8. Documentation is done to increase  
(A) Readability of program      (B) Development time of program  
(C) Cost of program      (D) All of the above
9. Testing can be -  
(A) Manual only      (B) Manual and Automated  
(C) Automated only      (D) None of the above
10. Data structure and code is access by a tester in -  
(A) Black Box Testing      (B) Acceptance Testing  
(C) White Box Testing      (D) Stress Testing
11. .... is steps by step execution of program.  
(A) Testing      (B) Compilation  
(C) Debugging      (D) All above
12. \_\_\_ is a failure if a program doesn't work correctly.  
(A) Programming      (B) Testing  
(C) Both a and b      (C) None of these
13. Which of the following is not a characteristic for Testability?  
(A) Operability      (B) Observability  
(C) Simplicity      (D) Robustness
14. Cyclomatic Complexity method comes under\_ testing method.  
(A) White box      (B) Black box  
(C) Green box      (D) Yellow box
15. Testing is a \_\_\_ if a program does not work correctly.  
(A) Failure      (B) Success  
(C) Complete      (D) Partial
16. A \_\_\_ is a sequence of statements from one place in the program to another.  
(A) Route      (B) Path  
(C) Sub path      (D) Gateway
17. Loop Testing comes under which testing method?  
(A) White Box      (B)'Black Box  
(C) Green Box      (D) Yellow Box
18. Which of these can be successfully tested using Loop Testing methodology?  
(A) Simple Loops      (B) Nested Loops  
(C) Concatenated Loops      (D) All of the above
19. Graph based testing comes under which testing methods?  
(A) White Box      (B) Black Box  
(C) Green Box      (D) Yellow Box
20. Which testing methods are used by end-users who actually test software before they use it.  
(A) Alpha and Beta Testing      (B) White Box Testing  
(C) Black Box Testing      (D) Trial and Error Testing'
21. To test a function, the programmer has to write a \_\_\_, which calls the function and passes it test data.  
(A) Stub      (B) Driver  
(C) Proxy      (D) None of the above
22. White box testing is primarily:  
(A) Data driven      (B) Logic driven  
(C) Bottom up driven      (D) Defect driven

- 23. A regression test:**  
(A) Will always be automated  
(B) Will help ensure unchanged areas of the software have not been affected  
(C) Will help ensure changed areas of the software have not been affected  
(D) Can only be run during user acceptance testing
- 24. Verification is-**  
(A) Checking that we are building the right system.  
(B) Checking that we are building the system right  
(C) Performed by an independent test team  
(D) Making sure that it is what the user really wants
- 25. The purpose of requirement phase is -**  
(A) To freeze requirements (B) To understand user needs  
(C) To define the scope of testing (D) All of the above
- 26. ... is the process of locating and correcting program errors**  
(A) Testing (B) Executing  
(C) Debugging (D) None
- 27. ... is said to be important step in program development**  
(A) Testing (B) Debugging  
(C) Both (D) None
- 28. These errors occur when the values of programming language are not followed.**  
(A) Syntax Errors (B) Run-time errors  
(C) Compile Time Errors (D) Logical errors
- 29. A program cannot be compiled and executed until these errors have been corrected**  
(A) Run-time errors (B) Logical error  
(C) None (D) Syntax errors
- 30. "" Errors typically involve \_ incorrect punctuations, undefined term or misuse of terms.**  
(A) Compile-time Errors (B) Syntax Errors  
(C) Logical Errors (D) Run-time Errors
- 31. ... are the error in planning the program logic**  
(A) Syntax errors (B) Compile time errors  
(C) Logical errors (D) None
- 32. ... Errors produce incorrect output**  
(A) Compile time errors (B) Logical errors  
(C) Syntax error (D) All the above
- 33. A program with ... errors can be compiled and executed but will produce wrong output .**  
(A) Run time errors (B) Syntax errors  
(C) Logical errors (D) Compile time errors
- 34. . . are the errors which are detected and identified by the compiler and the execution of the program cannot be completed until all errors are rectified.**  
(A) Syntax errors (B) Logical errors  
(C) Bugs (D) Compile time errors
- 35. "" Errors are the errors that the compiler missed during compile time.**  
(A) Run time errors (B) Virus  
(C) Compile time errors (D) None
- 36. These errors occur when a logical mistake has been taken place resulting in problem such as in finite loops**  
(A) Bugs (B) Compile time errors  
(C) Syntax errors (D) Run time errors

37. .... Is the process of finding and reducing the no of bugs, or errors, in a computer program thus making it behave as expected  
(A) Debugging (B) Testing  
(C) Compiling (D) Executing
38. ... is tool is used for debugging  
(A) Assembler (B) Interpreter  
(c) Compiler (D) Debugger
39. ... is the software tool which enables the programmer to monitor the execution of program, stop it, start it, etc  
(A) Debugger (C) Compiler  
(B) Loader (D) Linker
40. A person who carries out the debugging process is referred to as  
(A) Programmer (B) Debugger  
(C) Developer (D) None
41. ... is the software tool which helps the programmer in following the step-by-step execution of a program by allowing display of intermediate calculations and result whenever necessary.  
(A) Compiler (C) Debugger  
(B) Assembler (D) None
42. ... is the process undertaken to access the quality of computer software.  
(A) Debugging (C) compiling  
(B) None (D) Testing
43. Various approaches like reviews, walkthroughs or inspections in a software testing are considered as ...  
(A) Unit Testing (B) Static Testing  
(C) Dynamic testing (D) All
44. The actual execution of a program with a given set of test cases in a given development stage is referred as ...  
(A) Dynamic Testing (B) Static Testing  
(C) White box testing (D) Unit Testing
45. ..... Is the process which involves ensuring that the final product matches the customer requirements?  
(A) Testing (C) Validation  
(B) Debugging (D) Verification
46. ..... is the process of ensuring that the product has been built matches all the specification.  
(A) Verification (B) Validation  
(C) Debugging (D) Testing
47. In which of the testing internal behavior is not needed?  
(A) White box testing (B) Black box testing  
(C) System testing (D) Integration
48. Which of the testing involves feeding the input and observing the output from the test object?  
(A) White box testing (B) Black box testing  
(C) System testing (D) Unit testing
49. ..... testing is used when the tester has access to the internal data structures, codes and algorithms:  
(A) Integration box testing (B) Unit testing  
(C) Black box testing (D) White box Testing

**50. In which of the testing each unit of the software is tested to verify that the detailed design for the unit has been correctly implemented?**

- |                      |                        |
|----------------------|------------------------|
| (A) Unit box testing | (B) Black box testing  |
| (C) System testing   | (D) Regression testing |

**ANSWERS**

1. A	2. A	3. A	4. A	5. C	6. B	7. C	8. A	9. B	10. C
11. C	12. B	13. D	14. A	15. A	16. C	17. A	18. D	19. B	20. A
21. B	22. B	23. B	24. B	25. D	26. C	27. C	28. A	29. D	30. B
31. C	32. B	33. C	34. D	35. A	36. D	37. A	38. D	39. A	40. B
41. C	42. D	43. B	44. A	45. C	46. B	47. B	48. B	49. D	50. A

**1. Memory unit is one part of**

- |                   |                             |
|-------------------|-----------------------------|
| (A) Input device  | (B) Control unit            |
| (C) Output device | (D) Central Processing Unit |

**2. The basic operations performed by a computer are**

- |                          |                       |
|--------------------------|-----------------------|
| (A) Arithmetic operation | (B) Logical operation |
| (C) Storage and relative | (D) All the above     |

**3. The earliest calculating devices are**

- |                       |                   |
|-----------------------|-------------------|
| (A) Abacus            | (B) Clock         |
| (C) Difference Engine | (D) None of these |

**4. The man who built the first Mechanical Calculator was**

- |                           |                    |
|---------------------------|--------------------|
| (A) Joseph Marie Jacquard | (B) John Mauchly   |
| (C) Blaise Pascal         | (D) Harward Ailken |

**5. Punched cards were first introduced by**

- |              |                      |
|--------------|----------------------|
| (A) Powers   | (B) Pascal           |
| (C) Jacquard | (D) Herman Hollerith |

**6. Computers built before the First Generation of computers were**

- |                |                        |
|----------------|------------------------|
| (A) Mechanical | (B) Electro-mechanical |
| (C) Electrical | (D) None of these      |

**7. The unit KIPS is used to measure the speed of**

- |               |                |
|---------------|----------------|
| (A) Processor | (B) Disk drive |
| (C) Printer   | (D) Tape drive |

**8. What is the name of the latest Server Operating System developed by Microsoft?**

- |                |                  |
|----------------|------------------|
| (A) Windows NT | (B) Windows 2000 |
| (C) Windows XP | (D) Windows 2003 |

**9. What is the name of the software that allows us to browse through web pages called?**

- |                |                 |
|----------------|-----------------|
| (A) Browser    | (B) Mail Client |
| (C) FTP Client | (D) Messenger   |

**10. Macromedia is a name of a company related with**

- |                 |              |
|-----------------|--------------|
| (A) Hardware    | (B) Software |
| (C) Peripherals | (D) Services |

**11. What is the address given to a computer connected 'to a network called?**

- |                    |                |
|--------------------|----------------|
| (A) System Address | (B) SYSID      |
| (C) Process ID     | (D) IP Address |

**12. Direct X is a**

- |                      |   |
|----------------------|---|
| (A) Computer Part    | (B) Software that drives Graphic hardware |
| (C) A User Interface | (D) None of these                         |

**13. When you purchase a product over a Mobile Phone, the transaction is called**

- (A) Web Commerce                         (B) e-Commerce  
(C) m-Commerce                             (D) Mobile Purchases
- 14. Which of the following device can store large amounts of data?**  
(A) Floppy Disk    (B) Hard Disk  
(C) CDROM        (D) Zip Disk
- 15. Data (information) is stored in computers as**  
(A) Files           (B) Directories  
(C) Floppies        (D) Matter
- 16. Which technology is used in a CDROM Drive?**  
(A) Mechanical                                 (B) Electromechanical  
(C) Optical                                      (D) Fiber Optical
- 17. MTBF means**  
(A) Mean Time Before Failure  
(B) Master Time Buffer Feature  
(C) Most Treated Buffer Time  
(D) Master Test Board Feature
- 18. Floppy Disk Drives were first introduced by which of the following computer manufacturers?**  
(A) IBM              (B) Sony  
(C) Panasonic      (D) Compaq
- 19. Which of the following companies is a leader in manufacture of Hard Disk Drives?**  
(A) Samsung          (B) IBM  
(C) Fujitsu           (D) Segate
- 20. Usually, in MSDOS, the primary hard disk drives has the drive letter**  
(A) A :              (B) B :  
(C) C:               (D) D :
- 21. Which of the memories below is often used in a typical computer operation?**  
(A) RAM              (B) ROM              (C) FDD              (D) HDD
- 22. Time taken to move from one cylinder of a hdd to another is called**  
(A) Transfer rate      (B) Average seek time  
(C) Latency            (D) Roundtrip time
- 23. Which of the following RAM times have to be refreshed often in order to retain its contents?**  
(A) SIMM              (B) DIMM  
(C) SDMM              (D) DSMM
- 24. Which of the following is not a logic gate?**  
(A) AND              (B) OR  
(C) NOT              (D) NAT
- 25. The Analytical Engine developed during First Generation of computers used \_ as a memory unit**  
(A) RAM              (B) Floppies  
(C) Cards            (D) Counter Wheels
- 26. Which storage device is mounted on 'reels'?**  
(A) Floppy Disk                                 (B) Hard Disk  
(C) Magnetic Tapes                             (D) CDROM
- 27. Which of the following statements is/are true?**  
(A) Cache Memories are bigger than RAM      (B) Cache Memories are smaller than RAM  
ROM is faster than RAM                             (D) Information in ROM can be written by users      (C)
- 28. In a computer \_\_ is capable to store single binary bit.**  
(A) Capacitor        (B) Flip flop

- (C) Register (D) Inductor  
**29.** A set of flip flops integrated together is called \_\_  
(A) Counter (B) Adder  
(C) Register (D) None of the above
- 30.** Which of the following are the best units of data on an external storage device?  
(A) Bits (C) Hertz  
(B) Bytes (D) Clock cycles
- 31.** Separate ReadWrite heads are required in which of these memory access schemes.  
(A) Random Access (B) Sequential Access  
(C) Direct Access (D) None of these
- 32.** A register organized to allow to move left or right operations is called as \_\_  
(A) Counter (B) Loader  
(C) Adder (D) Shift register
- 33.** Which of the following are the cheapest memory devices in terms of Cost / Bit?  
(A) Semiconductor Memories (B) magnetic Disks  
(C) Magnetic Tapes (D) Compact Disks
- 34.** Which of the following have the fastest access time?  
(A) Semiconductor Memories (B) Magnetic Disk's  
(C) Magnetic Tapes (D) Compact Disks
- 35.** \_\_ is a semi conductor memory.  
(A) Dynamic (B) Static  
(C) Bubble (D) Both A & B
- 36.** Which of the following is a read only memory storage device.  
(A) Floppy disk (B) Hard disk  
(C) CDROM . (D) None of these
- 37.** DMA stands for \_\_  
(A) Direct Memory Access (B) Distinct Memory Access  
(C) Direct Module Access (D) Direct Memory Allocation
- 38.** transforms one interface into another interface  
(A) Program (C) Data  
(B) Software (D) None
- 39.** interface consists of things --- like program counter, registers, interrupts and terminals  
(A) Hardware (B) Software  
(C) Data (D) None
- 40.** Swapping  
(A) Works best with many small partitions  
(B) Allows many programs to use memory simultaneously to use the memory  
(C) Allows each program in turn  
(D) Does not work with overlaying
- 41.** Poor response times are usually caused by  
(A) Process busy (B) High I/O rates  
(C) High paging rates (D) Any of the above
- 42.** Which of the following program is not a utility?  
(A) Debugger (C) Spooler  
(B) Editor (D) All above
- 43.** A co-processor .  
(A) Is relatively easy to support in software (B) Causes all processors to function equally  
(C) Works with any application (D) Is quite common in modern computers
- 44.** Page stealing

- (A) Is a sign of an efficient system      (B) Is taking page frames from other working sets  
 (C) Should be the tuning goal
- 45. Which generation of computer is still under development?**  
 (A) Fourth      (B) Fifth  
 (C) Sixth      (D) Seventh
- 46. Artificial intelligence is associated with which generation?**  
 (A) First Generation      (B) Second Generation  
 (C) Seventh Generation      (D) Fifth Generation
- 47. Which of the operations are not performed by the computer?**  
 (A) Controlling      (B) Inputting  
 (C) Processing      (D) Understanding
- 48. Fifth generation computer is also known as----**  
 (A) Knowledge information processing system      (B) Very Large Scale Integration(VLSI)  
 (C) Both A & B      (D) None above
- 49. The brain of any computer system is ....**  
 (A) Control unit      (B) Central Processing unit  
 (C) Arithmetic Logic unit      (D) Storage unit
- 50. Which one of the following will declare a pointer to an integer at address Ox200 in memory?**  
 (A) int \*x = \*Ox200;  
 (B) int \*x(&Ox200)  
 (C) int \*x = &Ox200;  
 (D) \*x = Ox200;

**Answer:**

1. D    2. D    3. A    4. C    5. D    6. B    7. A    8. B    9. A    10. B  
 11. D    12. B    13. C    14. B    15. A    16. C    17. A    18. A    19. D    20. C  
 21. A    22. B    23. D    24. D    25. D    26. C    27. B    28. B    29. C    30. B  
 31. D    32. D    33. D    34. A    35. D    36. C    37. A    38. B    39. A    40. C  
 41. D    42. C    43. A    44. B    45. B    46. D    47. D    48. A    49. B    50. D

- 1. ..... testing exposes defects in the interfaces and interactions between integrated components.**  
 (A) White box testing      (B) System testing  
 (C) Regression testing      (D) Integration testing
- 2. In which of the testing, it tests a completely integrated system to verify that it meets its requirements?**  
 (A) Unit testing      (B) Sanity box testing  
 (C) System Integration testing      (D) System testing
- 3. .....testing verifies that a system is integrated to an external third party system defined in the system requirements**  
 (A) Black box testing      (B) White box testing  
 (C) System integration testing      (D) Unit testing
- 4. " ..... can begin in the early stages of software development.**  
 (A) Testing      (B) Debugging      (C) Documentation      (D) Compiling
- 5. .....'" can begin only after program is coded.**  
 (A) Testing      (B) Documentation      (C) Compiling      (D) Debugging
- 6. .....is said to be complete, when requirement verification have been performed.**  
 (A) Debugging      (B) Compiling      (C) Documentation      (D) Testing
- 7. ..... is said to be complete when all errors are known and have been fixed.**

- (A) Debugging      (B) Testing      (C) Execution      (D) None
- 8. Which of the following is the type error?**  
 (A) Syntax      (B) Logical      (C) Run time      (D) All the above
- 9. Key used to start debugging in C Language program.**  
 (A) F4      (B) F5  
 (C) F6      (D) F7
- 10. What is break point in debugging?**  
 (A) Signal to suspend execution of program at that point  
 (B) Signal to suspend execution of program  
 (C) Resume program execution  
 (D) Cancel program execution
- 11. What are watches in debugging?**  
 (A) Monitor stack values      (B) Monitor function values  
 (C) Monitor Procedure execution      (D) Monitor values of variables
- 12. What is immediate window in debugging?**  
 (A) Monitor Stack Values      (B) Monitor function values  
 (C) Monitor procedure execution      (D) Monitor value of variables
- 13. F8 key in C language program is used for ..... while debugging.**  
 (A) Overriding function execution      (B) Bypass function execution  
 (C) Override variable      (D) Bypass variable
- 14. Bug has appeared in a program on 1025 line onward what will be used for fast debugging?**  
 (A) User break point      (B) Use watches      (C) Use variables      (D) Use help
- 15. How will you bypass going into function while debugging in C language program?**  
 (A) Use Ctrl + F9      (B) Use F8      (C) Use Ctrl + F8      (D) Use F7
- 16. A program can be debugged only if it has no .....**  
 (A) Syntax Errors      (B) Logical Errors      (C) A & B both      (D) Divide by zero error
- 17. Debugging can only be done for ..... other than removal of bug**  
 (A) Understanding Logic of program      (B) Time pass  
 (C) A and B both      (D) None of above
- 18. For debugging, a program should be ..... free.**  
 (A) Syntax Error      (B) Logical Error      (C) Warning      (D) All of the above
- 19. Debugging helps in understanding.....**  
 (A) Logic of program      (B) Flow of program      (C) A and B both      (D) None
- 20. Debugging can be done only in .....**  
 (A) Executable file      (B) Source code file      (C) Object file      (D) Header file
- 21. ..... file cannot be debugged in C language.**  
 (A) Header file      (B) Libraries      (C) Executable File      (D) All of above
- 22. In C Language ..... key is used to run program after debugging.**  
 (A) Ctrl + F1      (B) Ctrl + F2      (C) Ctrl + FB      (D) Ctrl + F9
- 23. Toggle break point in debugger is used for .....**  
 (A) Assign or remove break point      (B) Only assign break point  
 (C) Only remove break point      (D) None
- 24. Program testing is done for detection of .....**  
 (A) Variable in program      (B) Data type in program  
 (C) Bug's in program      (D) Structures in program
- 25. For program testing to be done program must be Free.**  
 (A) Syntax error      (B) Logical      (C) A and B both      (D) Warning
- 26. Program testing has to be done ..... debugging.**  
 (A) Before      (B) After      (C) Before and After      (D) Never

**27. Program is cycle of software engineering.**

- (A) Fourth      (B) First  
 (C) Second     (D) Third

**28. API stands for .**

- (A) Application program interconnection      (B) Application program interface  
 (C) Application program inter call        (D) None

**29. API contains.**

- (A) Routines      (B) Data structures      (C) Objects      (D) All the above

**30. Logical error is also called as**

- (A) Semantic error      (B) compilation error      (D) syntax error      (D) Warning error

**31. Logic error produces.**

- (A) Desired output      (B) Undesired output      (C) Other behaviour      (D) Both A and B

**32. Removal of Logic errors requires .....the program.**

- (A) Scanning      (B) Printing      (C) Debugging      (D) Deleting

**33. API/libraries can be.**

- (A) Language dependent      (B) Language      (C) Both A and B      (D) None independent

**34. Header files in C language are .....**

- (A) Procedures      (B) Functions      (C) Routines      (D) Libraries

**35. API Information is ..... by the company who generates the API.**

- (A) Protected      (B) Unprotected      (C) Related      (D) None

**36. Libraries can be created for.....**

- (A) Generalized functions      (B) Generalized procedures      (C) Variables      (D) Both A and B

**37. Program testing should be done by.....**

- (A) Programmer      (B) Developer      (C) Other people      (D) Both A and B

**38. Before releasing of program..... "" is released.**

- (A) Beta version      (B) Sample      (C) Directly program      (D) None

**39. Serious error/ bug in released program are incorporated in .....**

- (A) Next version      (B) Next edition      (C) Next Patches      (D) Left as it is

**40. All upgrades in previous release of program are incorporated in**

- (A) Next version      (B) Next Edition (C) Next Patches      (D) Left as it is

**41. When a key is pressed on the keyboard, which standard is used for converting the keystroke into the corresponding bits**

- (A) ANSI      (B) ASCII      (C) EBCDIC      (D) ISO

**42. Which device is used as the standard pointing device in a Graphical User Environment**

- (A) Keyboard      (B) Joystick      (C) Mouse      (D) Track ball

**43. Which number system is usually followed in a typical 32-bit computer?**

- (A) 2      (B) 10      (C) 16      (D) 32

**44. Which of the following is not an output device?**

- (A) Scanner      (B) Printer      (C) Flat Screen      (D) Touch Screen

**45. Which of the following devices have a limitation that we can only read information from it but cannot erase or modify it**

- (A) Floppy Disk      an Hard Disk      (C) Tape Drive      (D) CDROM

**46. Which technology is used in Compact disks?**

- (A) Mechanical      (B) Electrical      (C) Electro Magnetic      (D) Laser

**47. Which of the following storage devices can store maximum amount of data?**

- (A) Floppy Disk      (B) Hard Disk      (C) Compact Disk      (D) Magneto Optic Disk

**48. The programs which are as permanent as hardware and stored in ROM is known as**

- (A) Hardware      (B) Software      (C) Firmware      D) ROMware

**49. Primary memory stores**

(A) Data alone      (B) Programs alone      (C) Results alone      (D) All of these

**50. Which device can understand difference between data & programs?**

(A) Input device      (B) Output device      (C) Memory      (D) Microprocessor

#### ANSWERS

1. D    2. D    3.C    4.C    5.D    6. D    7. A    8. D    9.D    10. A    11.D    12.D    13.B  
 14.A    15. B    16. A    17. A    18. A    19.C    20. B    21. D    22. D    23.A    24.C    25. A  
 26. C    27. A    28.B    29.D    30. A    31. D    32. C    33.C    34.D    35. A    36. D    37. C    38.A  
 39.C    40. A    41. B    42. C    43.D    44.D    45. D    46. D    47. B    48.C    49.D    50. D

**1. Pick the one that is used for logical operations or comparisons such as less than equal to or greater than**

(A) Arithmetic and Logic unit      (B) Control unit      (C) Both A & B      (D) None

**2. Analog computers work on the supply of -----**

(A) Continuous electrical pulses      (B) Electrical pulses but not continuous  
 (C) Magnetic strength      (D) None of theses

**3. Digital devices are -----**

(A) Clock with a dial and two hands      (B) Digital clock  
 (C) Automobile speed meter      (D) All of them

**4. The computer that process both analog & digital is called -----**

(A) Analog Computer      (B) Digital Computer      (C) Hybrid Computer      (D) Mainframe

**5. UNIVAC stands for -----**

(A) Universal Automatic Computer      (B) Unique Automatic Computer  
 (C) Universal Array Computer      (D) Unvalued Automatic Computer

**6. CD-ROM stand for-----**

(A) Compactable Read Only      (B) Compactable Disk Read Only Memory  
 (C) Compact Data Read Only memory      (D) Compact Disk read Only Memory

**7. ALU is -----**

(A) Arithmetic logic Unit      (B) Array Logic Unit  
 (C) Application Logic Unit      (D) None of these

**8. VGA stands for -----**

(A) Video Graphics Array      (B) Video Graphics Adapter  
 (C) Visual Graphics Array      (D) Volatile Graphics Array

**9. IBM 1401 is -----**

(A) First Generation      (B) Second Generation  
 (C) Third Generation      (D) Fourth Generation

**10. MSI stands for -----**

(A) Medium System Intelligent Circuit      (B) Medium System Integrated Circuits  
 (C) Medium Scale Intelligent Circuit      (D) Medium Scale Integrated Circuits

**11. The capacity of 3.5 inch floppy disk is -----**

(A) 1.40 MB      (C) 1.44MB  
 (B) 1.44 GB      (D) 1.40 GB

**12. The first computer introduced in Nepal was-----**

(A) IBM 1400      (B) IBM 1401  
 (C) IBM 1402      (D) IBM1402

**13. WAN stand for**

(A) Wide Area Network      (B) Wap Area Network  
 (C) Wide Area Net      (D) Wireless Area Network

- 14. MICR stands for -----**  
(A) Magnetic Ink Code Reader      (B) Magnetic Ink Character Reader  
(C) Magnetic Ink Cases Reader      (D) None of these
- 15. EBCDIC stands for -----**  
(A) Extended Bit Code Decimal Interchange Code  
(B) Extended Binary Case Decimal Interchange Code  
(C) Extended Bit Case Decimal Interchange Code  
(D) Extended Binary Coded' Decimal Interchange code
- 16. BCD is -----**  
(A) Bit Coded Decimal      (B) Binary Coded Digit  
(C) Binary Coded Decimal      (D) Bit Coded Digit
- 17. ASCII stands for -----**  
(A) American Stable Code For International Interchange  
(B) American Standard Code For Interchange Information  
(C) American Stable Case For Information Interchange  
(D) American Standard Code For Institutional Interchange
- 18. Which of the following is the first generation computer?**  
(A) IBM-1401      (B) EDSAC  
(C) CDC-1604      (D) ICL 2900
- 19. Chief component of first generation computer was -----**  
(A) Vacuum Tubes & Valves      (B) Integrated Circuits  
(C) Transistors      (D) None of these
- 20. FORTRAN is -----**  
(A) File Translation      (B) Format Translation  
(C) Formula Translation      (D) Floppy Translation
- 21. EEPROM stands for -----**  
(A) Electrically Erasable Programmable Read Only Memory  
(B) Easily Erasable Programmable read Only Memory  
(C) Electronic Erasable Programmable Read Only Memory  
(D) None of the Above
- 22. Second Generation computers were developed during -**  
(A) 1949 to 1955      (B) 1956 to 1965  
(C) 1970 to 1990      (D) 1965 to 1970
- 23. The computer size was large in ----- Generation**  
(A) Fourth      (B) Second  
(C) Third      (D) First
- 24. What is a job of Scheduler?**  
(A) Share processor time in all running process  
(B) Share output devices in all running process  
(C) Share input device in all running process  
(D) All of the above
- 25. Basic Concepts used in virtual memory**  
(A) Online Secondary Storage      (B) Demand Paging  
(C) Swapping      (D) All of above
- 26. Which of the division specifies the program & its author?**  
(A) Data Division      (B) Environment Division  
(C) None      (D) Identification Division
- 27. Which of the division specifies the computer peripherals use to compile & execute the program?**  
(A) Environment Division      (B) Data Division

- (C) Identification Division    (D) Procedure Division  
**28. Which of the division specifies the structure & format of the input & output data files?**  
(A) Environment Division    (B) Procedure Division  
(C) Data Division    (D) Identification Division
- 29. Which of the division specifies the sequence of operations to be performed by the program?**  
(A) Identification Division    (B) Data Division  
(C) Procedure Division    (D) Environment Division
1. A    2. A    3. B    4. C    5. A    6. D    7. A    8. A    9. B    10. D    11. C    12. B  
13. A   14. B   15. D   16. C   17. D   18. B   19. A   20. C   21. A   22. B   23. C  
24. A   25. D   26. D   27. A   28. C   29. C

- 1. Comments are written using the**  
(A) General English Statements  
(B) Assembly Language Statements  
(C) Higher Level Language Statements  
(D) Block Of Code
- 2. A system manual contains**  
(A) Input requirements, forms, type of output required, flowcharts, control procedure  
(B) Information about OS  
(C) Manual of computer systems  
(D) Multimedia information
- 3. \_\_\_ do not contain any program logic and are ignored by language processor.**  
(A) Protocols    (C) Comments  
(B) Loops        (D) None above
- 4. Documentation is carried out in ..... phase.**  
(A) Maintenance    (B) Testing  
(C) System requirement    (D) Implementation
- 5. Comments are**  
(A) Executable statements  
(B) Non executable statements  
(C) Assignment statements  
(D) Input/output statements
- 6. Documentation is any communicable material ( such as text, video, audio, etc., or combinations thereof) which \_\_**  
(A) Explain some attributes of an object, system or procedure  
(B) Are in books or computer readable file formats  
(C) Describe the structure and components, or on the other hand, operation, of System.  
(D) All of above
- 7. Consider the following statements:**  
(a) Indentation makes programs more readable and simpler to understand  
(b) Indentation is compulsory while writing a program  
Which of the following option is correct?  
(A) Only (a) is true    (B) Only (b) is true  
(C) Both (a) and (b) are true                                      (D) Both (a) and (b) are false
- 8. Documentation standards use**  
(A) Hungarian notations  
(B) Comments  
(C) Function description

9. **What does user manual provide?**  
(A) Help for developer  
(B) Help for end user  
(C) Help for tester  
(D) Help for analyst
10. **Which of the following is generally used for documentation?**  
(A) Comments    (B) Variables  
(C) Data types    (D) Functions
11. **----- also specifies the information about the security measures for using the software.**  
(A) Program Messages  
(B) User manual  
(C) System manual  
(D) Comments
12. **User manual are used for**  
(A) Modifying the program  
(B) Maintaining a program  
(C) To know the operational details of program  
(D) None of above
13. **The instructions in machine language must be in streams of \_**  
(A) Decimal digits  
(B) ASCII code  
(C) Os & Is  
(D) UNICODE
14. **Today's computers belong to generation.**  
(A) Third    (B) Fifth  
(C) Fourth    (D) Second
15. **Which of the following are characteristics of a good programming language?**  
(A) Safety    (B) Simplicity  
(C) Performance    (D) All above
16. **The command is used to store a program within the computer.**  
(A) Store command  
(B) Hold command  
(C) Save command  
(D) Load Command
17. **Characteristics of good programming are**  
(A) Simplicity, natural, efficient, compactness.  
(B) Hard to understand, lengthy & incompact.  
(C) Unstructured, inefficient & complex.  
(D) Complex, English - like, non- modular
18. **A system call is a method by which a program makes a request to the**  
(A) I/O management  
(B) Memory management  
(C) Interrupt processing  
(D) Operating system
19. **The most important aspect of program coding is**  
(A) Readability  
(B) Usability  
(C) Productivity

- (D) All above
- 20. Which of the following is not a characteristic of a good programming language?**
- (A) Simplicity    (B) Natural  
(C) Locality    (D) Complexity
- 21. Which of the following is not related to machine language?**
- (A) Opcode  
(B) Data movement operations  
(C) Instruction set  
(D) None
- 22. Which of the following is not case sensitive language?**
- (A) C    (B) JAVA  
(C) C++    (D) None of these
- 23. In which of the following language the 'opcode' is used?**
- (A) Assembly language  
(B) Machine language  
(C) High-level language  
(D) None of these
- 24. In which of the following language a program can be written using symbolic names?**
- (A) Assembly language    (B) High-level language  
(C) Machine language    (D) All the above
- 25. The Language made of streams of 0 , s & 1 ' s is called as a**
- (A) Symbolic language  
(B) High level Language  
(C) Machine Language  
(D) Algorithm
- 26. Each line of program consists of four columns known as fields**
- (A) Machine language    (B) Assembly language  
(C) Scripting Language    (D) Pascal
- 27. Which of the following is a high-level language?**
- (A) BASIC    (B) PASCAL  
(C) FORTRAN    (D) All of the above
- 28. PASCAL is a**
- (A) Low level language  
(B) Machine level language  
(C) High Level language  
(D) Object oriented language
- 29. What is the correct file extension for a C++ program?**
- (A) C++    (B) C+    (C) CPP    (D) .CCP
- 30. Fortran is**
- (A) General purpose  
(B) Procedural  
(C) Imperative programming  
(D) All of above
- 31. Line editor and the types of editor**
- (A) Function editor  
(B) Module editor  
(C) Screen editor  
(D) None these

- 32.** The language that the computer can understand and execute is called  
(A) Low-level machine language  
(B) High-level language  
(C) Assembly Language  
(D) None of the above
- 33.** Which of the following factors should be considered while selecting a programming language for application development?  
(A) Nature of application  
(B) Ease of learning the language  
(C) Familiarity with the language  
(D) All A, B, C
- 34.** Interpreter is used to convert  
(A) Low level to high level,  
(B) High level to Machine level  
(C) Assembly to low level  
(D) None of these
- 35.** Which of the following languages is effective for mathematical calculations  
(A) FORTRAN (B) C  
(C) PASCAL (D) All of the above
- 36.** Instructions are encoded as number is a feature of  
(A) Assembly language  
(B) High level language  
(C) Machine language  
(D) C language
- 37.** Which of the following statement(s) is/are correct?  
(A) Linker is a program that takes one or more object generated by a Computer and assembles them into a single executable program  
(B) Linker is a program that takes one or more source program files and assembles them into a single executable program  
(C) Linker is a program that translates a high-level language program into its equivalent object code  
(D) None of the above
- 38.** Which of the Following is not a Translator program?  
(A) Assembler (B) Compiler  
(C) Interpreter (D) Linker
- 39.** A Linker  
(A) Combines different modules of the program  
(B) Allows user to write a program  
(C) Finds out errors  
(D) Is used to debug '3 program
- 40.** 'C' can be used on platform(s),  
(A) MS-DOS operating system  
(B) Linux operating system  
(C) Windows operating system  
(D) All the above
- 41.** Which of the following is an assembly language instruction?  
(A) 1.00E+15 (B) ADD AX 14  
(C) X = X + Y (D)(SET! X Y)
- 42.** Consider the following statements:  
(i) Compilers and Interpreters are used to find errors.  
(ii) Compilers are faster when compared to the interpreters

**Which of the following statement is correct?**

- (A) Both the statements are correct
- (B) Only first statement is correct
- (C) Only Second statement is correct
- (D) Both the statements are wrong

**43. Which of the following language is easy to debug?**

- (A) Assembly language
- (B) Machine language
- (C) All high-level languages
- (D) All the above

**44. Which of the following saves the generated object code?**

- (A) Interpreter    (B) Linker
- (C) Compiler        (D) Loader

**45. Advantage(s) of interpreters over compliers are**

- (A) They are less complex programs than compliers
- (B) They need less memory space for execution than compliers
- (C) Syntax error in a program statement is detected during processing of that statement
- (D) All of them

**46. FORTRAN stands for-**

- (A) Foreign translator
- (B) Formula Transmission
- (C) Formula Translator
- (D) Formula Transaction

**47. Which one of the following is an example of machine language?**

- (A) ADD r1, r2    (B) 10010111
- (C) y - y+2      (D) printf( "Welcome \ ");

**48. Assembly languages are High Level languages**

- (A) The statement is correct
- (B) The statement is wrong
- (C) The statement is partially correct
- (D) None of above

**49. Which files are linked by a Linker?**

- (A) Source Files
- (B) Object Files
- (C) Executable Files
- (D) Text Files

**50. Which of the following is a business oriented language?**

- (A) FORTRAN    (B) PASCAL
- (C) C            (D) COBOL

#### ANSWERS

- |      |       |       |       |       |       |       |       |       |       |
|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1. A | 6. D  | 11. B | 16. C | 21. D | 26. B | 31. C | 36. C | 41. B | 46. C |
| 2. A | 7. A  | 12. C | 17. A | 22. D | 27. D | 32. A | 37. A | 42. A | 47. B |
| 3. C | 8. D  | 13. C | 18. D | 23. B | 28. C | 33. D | 38. D | 43. C | 48. B |
| 4. D | 9. B  | 14. B | 19. D | 24. A | 29. C | 34. B | 39. A | 44. C | 49. B |
| 5. B | 10. A | 15. D | 20. C | 25. C | 30. D | 35. D | 40. D | 45. D | 50. D |

- 1. Which statement(s) is used to terminate the current loop immediately and transfer control to the statement immediately following that loop?**  
(A) Exit  
(B) break  
(C) Both exit & break  
(D) None of above
- 2. Which of the following statement is true?**
  1. Every program is an algorithm.
  2. Every algorithm is a program.

(A) Both      (B) Only 1  
(C) Only 2      (D) Neither 1 nor 2
- 3. Which of the following is an iterative control structure?**  
(A) Decision Making  
(B) Sequential  
(C) Jump  
(D) Loop
- 4. Which of the following structures are used in computer programs**  
(A) Sequential    (B) Decision  
(C) Iterative    (D) All of above
- 5. Instructions in algorithms should be**  
(A) Precise  
(B) Unambiguous  
(C) Precise & Unambiguous  
(D) None of above
- 6. Which of the following statement does not belong to structured programming?**  
(A) while      (B) do\_while  
(C) for      (D) goto
- 7. As compared to a flowchart, it is easier to modify the of program logic when program modifications are necessary.**  
(A) Macro flowchart  
(B) Micro flowchart  
(C) Terminal  
(D) Pseudo code.
- 8. Algorithm halts in**  
(A) Finite time  
(B) Infinite time  
(C) Logarithmic time  
(D) Exponential time
- 9. In which discipline(s), an algorithm is used ?**  
(A) Mathematics  
(B) Computing  
(C) Linguistics  
(D) All of above
- 10. The \_\_ flow chart symbol(s) represents one way flow of control.**  
(A) Processing    (B) Decision  
(C) Terminal      (D) All above
- 11. What is an infinite loop?**

- (A) It is an endless loop  
(B) It means multiple loops  
(C) It is a nested loop  
(D) It is an unclosed loop
- 12. The normal flow of flowchart is from**
- (A) Left to Right  
(B) Right to Left  
(C) A & D  
(D) Top to Bottom
- 13. Which tool shows textual design solution**
- (A) Flowchart  
(B) Structure chart  
(C) Pseudo code  
(D) Algorithm
- 14. Finiteness property of an Algorithm is**
- (A) The number of steps in the algorithm should be finite.  
(B) The algorithm should terminate after a finite no. of times.  
(C) For all possible combinations of input data, the algorithm terminates after a finite no. of steps  
(D) None of above
- 15. Pseudo code consists of and omits .**
- (A) Structural conventions of programming languages; subroutines, variable declarations or language-specific syntax  
(B) Subroutines; structural conventions of programming languages  
(C) Variable declarations; language-specific syntax  
(D) Subroutines; Functions
- 16. Terminal symbol in a flow chart indicates**
- (A) Decision    (C) Process  
(B) End        (D) None above
- 17. ----- Statement is used to indicate the end of a 'DO ... WHILE' construct in the pseudo code**
- (A) END DO    (B) DOEND  
(C) END        (D) CLOSE
- 18. A good algorithm is not**
- (A) Simple and powerful  
(B) Clear for implementation  
(C) Dependent on a particular machine  
(D) Effective
- 19. English statements that follow a loosely defined syntax & are used to convey the design of an algorithm is called**
- (A) Program    (B) Flowchart  
(C) Pseudo code    (D) None of the above.
- 20. A flowchart is used in \_\_ of the software development.**
- (A) Implementation phase  
(B) Testing phase  
(C) Analysis phase  
(D) Design phase
- 21. Indentation is used to format**
- (A) Program source code

- (B) Object code  
(C) Executable code  
(D) All of the above
- 22.** Consider the following statements regarding algorithms:  
(a) Each instruction of an algorithm should be executed in a finite time  
(b) One or more instructions of an algorithm should not be repeated infinitely  
(c) Any program is an algorithm  
  
(A) A, B, C and D are true  
(B) Only A, Band D are true  
(C) Only B, C and D are true  
(D) Only A, Band C are true
- 23.** Another name for pseudo code is  
(A) Imitation code  
(B) Flowchart  
(C) Program  
(D) Algorithm
- 24.** How many basic symbols are available to draw a flowchart?  
(A) 4    (B) 6  
(C) 8    (D) 7
- 25.** Which of the following is not the way to represent an algorithm?  
(A) As an executable code  
(B) As a program  
(C) As a flowchart  
(D) As a pseudo code
- 26.** Consider the following statements and determine which of the following is correct?  
(a) Indentation makes programs more readable and simpler to understand  
(b) Indentation is compulsory while writing a program  
  
(A) Only (a) is true  
(B) Only (b) is true  
(C) Both (a) & (b) are true  
(D) Both (a) & (b) are false
- 27.** The valid symbol(s) in flowchart is/are \_\_\_\_\_  
(A) Connector  
(B) Terminal Symbol  
(C) Processing Symbol  
(D) All of above
- 28.** The algorithm cannot be represented as  
(A) A flowchart  
(B) a program  
(C) a process  
(D) a pseudo code
- 29.** A decision symbol can be used for  
(A) A two way branch decision  
(B) A three way branch decision  
(C) Multiple way branch decision  
(D) All of the above
- 30.** What is a Hungarian notation?

- (A) Notation for writing Loops  
(B) Notation for Manipulating Pointers  
(C) Notation for writing Variable Names  
(D) Flowchart Notation
- 31.** **Each step in an algorithm should be performed in a time.**  
(A) Finite      (B) Infinite  
(C) short      (D) Long.
- 32.** **Which of the following statement is not appropriate?**  
(A) Indentation improves the performance of the program.  
(B) Indentation is needed to make the program more readable.  
(C) Indentation helps the program to distinguish control statements.  
(D) Indentation makes the program easy to debug.
- 33.** **The structured programming languages are also known as languages.**  
(A) Object oriented  
(B) Procedure oriented  
(C) Modular programming  
(D) All the above
- 34.** **A Symbol is used in a flowchart to represent arithmetic and data movement instructions.**  
(A) Flow lines      (B) Processing  
(C) Input/output      (D) Terminal.
- 35.** **The with arrowheads are used to indicate the flow of an operation, that is, the exact Sequence in which the instructions are to be executed.**  
(A) Flow lines      (B) Processing  
(C) Decision      (D) Terminal.
- 36.** **The use of an algorithm is not intended for**  
(A) Modularizing the programs  
(B) Documentation  
(C) Writing variable names  
(D) Debugging
- 37.** **Which of the following is a low level language?**  
(A) C  
(B) LISP  
(C) Machine Level Language  
(D) JAVA
- 38.** **Which of the following language is best suited for system-level programming?**  
(A) BASIC      (B) C  
(C) LISP      (D) JAVA
- 39.** **Which of the following sequence is correct?**  
(A) Source code - compiler - object code - linker - executable code  
(B) Source code -linker - object code compiler - executable code  
(C) Object code - compiler - source code - linker - executable code  
(D) Object code -linker - source code - compiler - executable code
- 40.** **The computer software has been classified into two categories. They are \_**  
(A) Hardware & Software  
(B) Input & Output  
(C) System Software & Application software  
(D) Linker & Loader
- 41.** **A 'C' program is portable means it**

- (A) Can run on any machine  
(B) Can write on any machine  
(C) Can read from as well as write to any machine  
(D) All of the above

42. **Which programming language is machine independent?**  
(A) Machine level language  
(B) Assembly level language  
(C) High level language  
(D) Both A and B

43. **A program that aids in effective execution of user programs is called**  
(A) Application program  
(B) System program  
(C) Both System and Application program  
(D) Neither System nor Application program

44. **instruct the assembler to perform certain actions during the assembly of programs**  
(A) Assembler directives  
(B) Compiler directives  
(C) Declarative statements  
(D) Imperative statements

45. **is an example of a High Level language.**  
(A) C ++      (B) Assembly language.  
(C) Java      (D) Both A & C

46. **Low level languages are** —  
(A) Machine level language  
(B) Assembly level language  
(C) High level language  
(D) Both A and B

47. **Language is understood by a computer without using translation as** (A)  
Assembly language  
(B) Symbolic language  
(C) Machine language  
(D) Higher level language

48. **Application software can be for** —  
(A) Operating system  
(B) Translator  
(C) General-purpose application & Application specific solutions  
(D) All of the above

49. **Which of the following is the easiest language to learn and use to write programs?** (A)  
High level language  
(B) Machine level language  
(C) Assembly level language  
(D) Middle level language

50. **Which of the following language is predecessor to C Programming Language?** (A)  
A  
(B) B  
(C) C++  
(D) BCPL

## ANSWERS

1. B    2. B    3. D    4. D    5. C    6. D    7. D    8. A    9. B    10. A

11. A    12. D    13. C    14. C    15. A    16. B    17. A    18. C    19. C    20. D  
 21. A    22. D    23. A    24. B    25. A    26. A    27. D    28. C    29. D    30. C  
 31. A    32. A    33. C    34. B    35. A    36. C    37. C    38. B    39. A    40. C  
 41. A    42. C    43. B    44. A    45. D    46. D    47. C    48. C    49. A    50. B

**1. How will you write comment in a „C” Program?**

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

**2. Which of the following is FALSE in C?**

- A. Keyword can be used as variable names
- B. Variable names can contain a digit
- C. Variable names do not contain a blank space
- D. Capital letters can be used in variable names

**3. In „C” arithmetic instruction cannot contain**

- A. Variables
- B. Constants
- C. Variable names on right side of =
- D. Constants on left side of =

**4. An expression contains relational operators, assignment operators and arithmetic operators. In the absence of parentheses, they will be evaluated in which of the following order**

- A. Assignment, Relational, Arithmetic
- B. Assignment, Relational, Assignment
- C. Relational, Arithmetic, Assignment
- D. Assignment, Arithmetic, Relational

**5. In b=6.6/a+2\*n; which operation will be performed first?**

- A. 6.6/a
- B. a+2
- C. 2\*n
- D. Depends upon compiler

**6. Which among the following is not a structured data type in C?**

- A. Union
- B. Pointer
- C. String
- D. Structure

**7. Which of the following operator is used to write expression in „C”?**

- A. {}
- B. ()
- C. []
- D. None of above

**8. Values of data items of types int, float, char are displayed by writing \_\_\_\_\_ in printf statement in C**

- A. %d, %f, %s

- B. %f, %d, %c
- C. %d, %d, %c
- D. %d, %f, %c

**9. The general form of printf statement is**

- A. printf ("format string " list of variables )
- B. print ("list of variables" format sting)
- C. printf ("format string list of variables ")
- D. print (" format sting " list of variables )

**10. The statement in „C“ is terminated by**

- A. {
- B. :
- C. ,
- D. None above

**11. The general form of for statement in C is**

- A. for (initialize counter, increment, test counter)
- B. for (increment counter; initialize counter; test counter)
- C. for (test counter; increment counter; initialize counter)
- D. for ( initialize counter; test counter; increment counter)

**12. Difference between „while“ and „do-while“**

- (A) \ 'while \ ' loop executes one or more times and \ 'do-while \ ' executes zero or more times (B) Both \ 'while \ ' loop and \ 'do- while \ ' executes one or more times
- (C) Both \ 'while \ ' loop and \ 'do-while \ ' executes zero or more times
- (D) \ 'while \ ' loop executes zero or more times and \ 'do-while \ ' executes one or more times

**13. To avoid the repetition of same code we are using.**

- (A) Array (C) Function
- (B) Function (D) Structure

**14. Number of functions that might be called in a 'C' program is \_\_**

- (A) 5 (B) 6
- (C) Any number of functions
- (D) 1

**15. void main()**

```
{\n    int a=12,b=12;\n    if(a=b)\n        printf("a and b are equal");\n}
```

What will be the output of the sample code shown above?

- (A) 12
- (B) Run time error
- (C) Compile time error
- (D) a and b are equal

- 16. Every recursive version has an equivalent (but possibly more or less complex) iterative version, and vice versa: validate this statement.**
- (A) It is true sometimes  
(B) TRUE  
(C) FALSE  
(D) None of above
- 17. Which element of the array does the expression num\*4+ references where „num“ is a name of array?**
- (A) Forth  
(B) Third  
(C) Fifth  
(D) First
- 18. In a ' C ' expression, how is a logical' AND ' represented?**
- (A) &            (B) ||  
(C) AND        (D) &&
- 19. How do you include a system header file called stdio.h in a ' C ' source file?**
- (A) #include<stdio.h>  
(B) #incl \ "stdio.h \ "  
(C) #includefile<stdio>  
(D) #include stdio.h
- 20. Which one of the following variable name is NOT a valid name?**
- (A)go\_cart    (B)go4it  
(C)4season    (D)run4
- 21. Which of the following shows the correct priority of arithmetic operators in ' C ' ? (Priority for leftmost operator is highest and priority for the rightmost operator is lowest. Operators with equal priority are separated with the word' or'**
- (A)\*\*, \* or t, + or -  
(B)\*\*, \*, t, +, -  
(C)\*\*, t, \*, +, -  
(D)t or \*, + or -
- 22. Which of the following statement transfers the control to the beginning of the loop?**
- (A) exit        (B)break  
(C)continue    (D)None above
- 23. A ' do-while' 100.;;> is useful when the statements within the loop must be executed:**
- (A) Only once (B) At least once (C) More than once (D) None of the above
- 24. Assuming an unsigned integer is represented using 16 bits, the maximum value that an integer constant can have is**
- (A)256        (B)32768  
(C)65536      (D)128
- 25. The break statement is used to exit from?**

- (A)an \ 'if\ ' statement  
(B) \ 'for \ ' statement  
(C)Both from the \ 'if\ ' and \ 'for \ ' statement    (D )The main function

**26. The two way selection is implemented using statement.**

- (A)case  
(B)else---if  
(C)switch  
(D)if---else

**27. The getch() function in' C' is\_**

- (A) User defined function  
(B) Library function  
(C) Both above  
(D) None above

**28. A pointer is a**

- (A) Derived data type  
(B) User defined data type  
(C) Abstract data type  
(D) All of the above

**29. Which of the following is correct way of declaring a float pointer**

- (A)float ptr    (B)float \*ptr  
(C)\*float ptr    (D)None above

**30. In code shown below, which is the line that contains an error?**

```
int fun(int x, y)
{
    int z;
    return z;
```

- (A) 1    (C)3  
(B)    (D)4

**31. Which of the following statements are true for the following Program?**

```
#include<stdio.h>
void main()
{
    int x=10, y=100%9;
    for(i=1;i<=10;i++)
        if(x!=y);
    printf( \ "x=%d y=%d \ ",x,y);
}
```

(A) The printf function is called 10 times  
(B) The program will produce the output x=10 y=1  
(C) The ; after if(x!=y); would produce an error  
(D) The program will not produce any output

**32. The printf statement is used to \_**

- (A) Print the message on the console
- (B) Read the data from keyboard
- (C) To store the value in the memory
- (D) None of the above

**33. Which of the following is not infinite loop?**

- (A) int i=1; while(i) int i=1; while(i){i++;}
- (B) fore; ;;
- (C) int True=0, false; while(True) { False=1;}
- (D) int y,x=0;

**34. Array can be initialized, provided they are**

- (A) Automatic
- (B) External
- (C) static
- (D) both B & C

**35. Which of the following ' C ' . statement is syntactically correct?**

- (A) for( );
- (B) for();
- (C) for( , );
- (D) for( ; ; )

**36. Which one of the following is not a valid character specification for C language?**

- (A) ASCII
- (C) Digit
- (B) Control
- (D) for( ; ; )

**37. The string manipulation function appends a string to the end of another string**

- (A) stradd
- (B) strcat
- (C) strcmp
- (D) strcpy

**38. In what sequence the initialization, testing and execution of the body is done in a ' do-while' loop**

- (A) Hninitialization, execution of the body, testing
- (B) Execution of the body, initialization, testing
- (C) Hninitialization, testing, execution of the body
- (D) N one of the above

**39. Which of the following is not a integer constant in ' C ' ?**

- (A) \ 'C\ r
- (B) -
- (C) 45
- (D) l.2

**40. A ' return' statement is used \_ .**

- (A) To return the value from a function
- (B) To exit from the program
- (C) To terminate the iterative loop
- (D) All of the above

**41. The strcat() function is used \_ .**

- (A) To copy string
- (B) To compare string
- (C) To reverse the string
- (D) None of these.

**42. An array is the data type.**

- (A) Primary
- (B) Derived

(C)User defined      (D)empty

**43. An end of a string is denoted by the \_\_\_ character.**

- (A)Enter key      (B)\\0  
 (C)\\ \\      (D)/0

**44. The syntax of the array declaration is:**

- (A)datatype nameofarray [size];  
 (B)nameofarray [size];  
 (C)datatype nameofarray ;  
 (D)all of above

**45. A ' continue' statement is used**

- (A)To terminate a loop  
 (B)To continue a loop  
 (C)To continue a program  
 (D)None of these

**46. If 'a' is a variable defined in a 'C' program then &a denotes the \_\_\_**

- (A)Content of a  
 (B)Address of a  
 (C)Both A and B  
 (D)None of these

**47. Which of the following loops executes at least once, though it is not satisfying the condition?**

- (A)while loop    (B)do--while loop  
 (C)for loop    (D)if-else

**48. If int x = 2945, what is the value of the expression (x%100)%10?**

- (A)5    (B)9  
 (C)294    (D)0.5

**49. If int x = 2945, what is the value of the expression x!10?**

- (A)5    (B)294  
 (C)294.5 (D)2

**50. Hint x = 2945, what is the value of the expression x%10?**

- (A)294    (B)5  
 (C)294.5    (D)0.5

#### ANSWER

1. C    2. A    3. D    4. B    5. A    6. B    7. B    8. D    9. A    10. D  
 11. D    12. D    13. C    14. C    15. D    16. B    17. C    18. D    19. A    20. C  
 21. D    22. C    23. B    24. C    25. C    26. D    27. B    28. A    29. B    30. A  
 31. B    32. A    33. C    34. D    35. D    36. B    37. B    38. A    39. D    40. A  
 41. D    42. B    43. B    44. A    45. B    46. B    47. B    48. B    49. B    50. B

**1. A sentinel is called as a \_\_\_\_\_**

- A. Variable      B.Counter      C. True value      D. Flag value

**2. Which digit(s) is /are used in a binary number system?**

- A. 0 and 2
  - B. 1 and -1
  - C. 0 and 1
  - D. 0 & 1 & 2
3. ROM is the \_\_\_\_\_
- A. Volatile memory
  - B. Non-volatile memory
  - C. Virtual memory
  - D. None of above
4. RAM stands for \_\_\_\_\_
- A. Read only memory
  - B. Random access memory
  - C. Recently Acquired memory
  - D. Read Ahead memory
5. Which of the following is not type of the printer?
- A. Dot matrix printer
  - B. Laser printer
  - C. Drum printer
  - D. Scanner
6. Which of the following is not part of the computer?
- A. Monitor
  - B. Hard disk
  - C. RAM
  - D. Printer
7. The operation included in the instruction set of a computer are \_\_\_\_\_
- A. Logical
  - B. Arithmetic
  - C. Input-Output
  - D. All of above
8. The main memory is also called as the \_\_\_\_\_
- A. Primary memory
  - B. Cache memory
  - C. Secondary memory
  - D. Auxiliary memory
9. What will be the hexadecimal equivalent of the binary number 1111
- A. D
  - B. F
  - C. C
  - D. E
10. The difference between main memory and secondary storage is that the main memory is \_\_\_\_\_ and the secondary storage is \_\_\_\_\_
- A. Temporary, permanent

- B. Permanent, temporary
  - C. Slow, fast
  - D. None of above
11. Secondary storage is also known as \_\_\_\_\_
- A. Primary Memory
  - B. Ancillary Memory
  - C. An Auxiliary Memory
  - D. Read only Memory
12. What does IBM stands for?
- A. Indian Business Machine
  - B. International Business Machine
  - C. Indian Business Model
  - D. International Business Model
13. CD-ROM is a \_\_\_\_\_
- A. Semiconductor memory
  - B. Optical memory
  - C. Magnetic memory
  - D. None of above
14. A logical system uses \_\_\_\_\_ number system.
- A. Binary
  - B. Decimal
  - C. Octal
  - D. Hexadecimal
15. Base of hexadecimal number system is.
- A. 2
  - B. 8
  - C. 10
  - D. 16
16. How many nibbles a byte contains?
- A. 2
  - B. 8
  - C. 4
  - D. 6
17. The 2's complement of 1000 is
- A. 111
  - B. 10
  - C. 1000
  - D. 1
18. The condition is tested at the \_\_\_\_\_ of loop in a 'while ' statement
- A. Start
  - B. End
  - C. Middle

- D. Any Where
19. The arithmetic operations are carried out using \_\_\_\_\_.  
A. Output Device  
B. ALU  
C. Memory Device  
D. Timing and Control Unit
20. Base of octal number system is.  
A. 2  
B. 8  
C. 10  
D. 16
21. Which is the smallest unit of memory?  
A. Byte  
B. Nibble  
C. Bit  
D. Word
22. What is the octal equivalent of the decimal number 33?  
A. 38  
B. 39  
C. 40  
D. 41
23. Laptop are also known as \_\_\_\_\_ Computers  
A. Mainframe  
B. Super  
C. Notebook  
D. Personal
24. After counting 0, 1, 10, 11, the next binary number is \_\_\_\_\_  
A. 11  
B. 100  
C. 101  
D. 111
25. The binary number system uses base of  
A. 2  
B. 8  
C. 10  
D. 16
26. How many bits a byte contains?  
A. 2  
B. 8  
C. 4  
D. 6
27. Which of the following is not a program planning tool?

- A. Flowchart
  - B. Structure chart
  - C. Pseudo codes
  - D. Loop
28. Common sentinel values use a 'Null' character for indicating
- A. The end of a null –terminated string.
  - B. The last string.
  - C. The previous of last string.
  - D. None of above
29. Actual execution of instructions in a computer takes place in
- A. ALU
  - B. Control unit
  - C. Storage unit
  - D. None of above
30. The use of mathematical logic for computer programming is also called \_\_\_\_\_
- A. Physical programming
  - B. Logical programming
  - C. View programming
  - D. Computer programming
31. Information retrieval is faster from
- A. Floppy disk
  - B. Magnetic tape
  - C. Hard disk
  - D. None of above
32. The basic operation performed by a computer are
- A. Arithmetic operation
  - B. Logical operation
  - C. Input and Output
  - D. All of above
33. Pseudo code instructions are phrases written in a \_\_\_\_\_
- A. Machine language
  - B. Assembly language
  - C. High level language
  - D. Natural language
34. Which device is used commonly as the standard pointing device in a Graphical User Environment
- A. Keyboard
  - B. Mouse
  - C. Joystick
  - D. Track ball
35. Which of the following is an input device?
- A. Monitor
  - B. Mouse

- C. Printer  
D. Editor
36. Which of the following is an output device?  
A. Monitor  
B. Keyboard  
C. Touch-screen  
D. Mouse
37. Which technology is used in reading a Compact disk?  
A. Mechanical  
B. Electrical  
C. Electro Magnetic  
D. Optical
38. Which of the following have the fastest access time?  
A. Semiconductor Memories  
B. Magnetic Disks  
C. Magnetic Tapes  
D. Compact Disks
39. Which of the following is the smallest & fastest computer?  
A. Super computer  
B. Quantum computer  
C. Micro computer  
D. Mini computer
40. Primary memory stores \_\_\_\_\_  
A. Input Data only  
B. Instructions only  
C. Output Data only  
D. All of above
41. Which of the following device has a limitation that we can only read information from it but cannot erase or modify it  
A. Floppy Disk  
B. Hard Disk  
C. Tape Drive  
D. CDROM
42. Which device can understand the difference between data and instructions?  
A. Input device  
B. Output device  
C. Memory  
D. Microprocessor
43. From a \_\_\_\_\_ we can only read the information. We cannot erase or modify the information  
A. Floppy Disk  
B. Hard Disk  
C. Tape Drive

D. CDROM

44. What is the other name for LAN card?

- A. Network Interface Card
- B. Network Connector
- C. Modem
- D. Internet Card

45. Which of the following storage device can store maximum amount of data?

- A. Floppy Disk
- B. Hard Disk
- C. Compact Disk
- D. DVD

46. Which of the following is the larger manufacturer of Hard Disk Drives?

- A. IBM
- B. Seagate
- C. Microsoft
- D. 3M

47. Which number system is usually followed in a typical 32-bit computer?

- A. Binary
- B. Decimal
- C. Hexadecimal
- D. Octal

48. Which of the following cables can transmit data at high speeds?

- A. Coaxial cable
- B. Fiber Optic Cable
- C. Twisted pair Cable
- D. UTP Cable

49. The program stored in ROM is known as \_\_\_\_\_

- A. Hardware
- B. Software
- C. Firmware
- D. ROMware

50. The octal number system includes \_\_\_\_\_.

- A. Only the digits 0 to 7
- B. Only the digits 0 to 8
- C. Only the digits 0 to 9
- D. Only the digits 0 and 1

### Answers

1.D	2.C	3.B	4.B	5.D	6.D	7.D	8.A	9.B	10.A	11.C	
12.B	13.B	14.A	15.D	16.A	17.C	18.B	19.B	20.B	21.C	22.D	23.C
24.B	25.A	26.B	27.D	28.A	29.A	30.B	31.C	32.D	33.D	34.B	

35.B 36.A 37.D 38.A 39.B 40.D 41.D 42.D 43.D 44.A 45.B  
46.B 47.B 48.B 49.C 50.A

1. One byte equals \_\_\_\_\_ bits
  - A. 7
  - B. 8
  - C. 10
  - D. 12
2. The information in the form of data is stored in \_\_\_\_\_
  - A. Memory data register
  - B. Memory address register
  - C. Memory access register
  - D. Memory arithmetic register
3. Who among the following is a Personal Computer manufacturer?
  - A. CISCO
  - B. IB
  - C. Kodak
  - D. APC
4. Which of the following is considered as auxiliary storage device?
  - A. Disk B.
  - RAM C.
  - ROM D.
  - Cache
5. Which of the following type of memory is used during execution of program instruction?
  - A. RAM
  - B. CDROM
  - C. FDD
  - D. HDD
6. EPROM can be used for \_\_\_\_\_
  - A. Erasing the contents of ROM
  - B. Reconstructing contents of ROM
  - C. Erasing & reconstructing contents of ROM
  - D. Duplicating ROM
7. The valid symbol(s) in flowchart is/are \_\_\_\_\_
  - A. Connector
  - B. Terminal symbol
  - C. Processing symbol
  - D. All of above
8. A file has to be created in the hard disk. Which one of the following parts of an operating system is useful in this context?
  - A. Processor management
  - B. Memory management
  - C. Secondary memory management

- D. Process scheduler
9. Printout of contents of main memory and registers are taken in \_\_\_\_\_  
A. Debugging  
B. Memory dump  
C. Hand simulation of program code  
D. Putting print statement in program code
10. When a key is pressed on the keyboard, which standard is used for converting the keystroke into the corresponding bits  
A. ANSI  
B. ASCII  
C. EBCDIC  
D. ISO
11. Central processing unit consist of  
A. Input and output unit  
B. Control unit and arithmetic logic unit  
C. Storage unit  
D. None of above
12. CPU is acronym for  
A. Computer program unit  
B. Central processing unit  
C. Central programing unit  
D. None of above
13. The memory location address are limited to a range of values from \_\_\_\_\_  
A. 00000 to 9ffff(16)  
B. 00001 to 9ffff(16)  
C. 00010 to 9ffff(16)  
D. 10000 to 9ffff(16)
14. Which out of the following is not a type of operation performed by a computer  
A. Arithmetic  
B. Logical  
C. Emotional  
D. Mathematical
15. Result of logical operation is \_\_\_\_\_  
A. Boolean  
B. Integer  
C. Character  
D. String
16. CISCO manufactures  
A. Software  
B. Processors  
C. Cables  
D. Network equipments

17. The earlier calculating device is
- Abacus
  - Clock
  - Difference engine
  - None of these
18. The first mechanical computer designed by Charles Babbage was called
- Abacus
  - Processor
  - Calculator
  - Analytical engine
19. What will be the subtraction of following binary numbers  $(1111) - (1100)$
- 100
  - 11
  - 101
  - 10
20. In computer technology, information means
- Raw data
  - Useful data
  - Alphanumeric data
  - Program
21. What is the first phase of Program development Life Cycle?
- Design
  - Testing
  - Coding
  - Analysis
22. The processor execute the instruction from the \_\_\_\_\_
- RAM
  - Pen drive
  - CDROM
  - HARD DRIVE
23. What will be the addition of the binary numbers  $(1111)+(1100)$
- 11011
  - 10011
  - 110110
  - 10111
24. What will be the BCD equivalent of the decimal number 12?
- 0001 0010
  - 0010 0001
  - 0010 1000
  - 1000 0100
25. The place where programs & data are stored temporarily during processing is
- Main memory

- B. Secondary memory
  - C. Hard disk
  - D. CD-ROM
26. Which of the following is a storage device?
- A. Store room
  - B. Printer
  - C. CPU
  - D. Pen Drive
27. Find the odd man
- A. CDROM
  - B. ROM
  - C. EPROM
  - D. PROM
28. Which of the following unit is used with computer system?
- A. Gifabyte
  - B. Kilobyte
  - C. Megabyte
  - D. All of above
29. What will be the binary equivalent of hexadecimal number 8?
- A. 10
  - B. 1000
  - C. 1110
  - D. 110
30. What of the following unit is not used to count the speed of a printer
- A. Character Per Second
  - B. Dot Per Inches
  - C. Page Per Minute
  - D. All of above
31. Which device is used as the standard input device in a textual user interface?
- A. Keyboard
  - B. Mouse
  - C. Joystick
  - D. Track ball
32. What will be the decimal equivalent of the binary number 10000
- A. 32
  - B. 16
  - C. 8
  - D. 24
33. ALU is called the \_\_\_\_\_ of a computer.
- A. Heart
  - B. Master Dispatcher
  - C. Primary Memory

- D. All
34. \_\_\_\_\_ gives a computer its unique address across the network.
- A. System Address
  - B. SYSID
  - C. Process ID
  - D. IP Address
35. Which of the following consortium looks for the standard representation of data in the Internet?
- A. ISOC
  - B. W3C
  - C. IEEE
  - D. IETE
36. Which of the following is not just an output device?
- A. Plotter
  - B. Printer
  - C. Flat Screen
  - D. Touch Screen
37. Which hardware was used by first generation computer?
- A. Vacuum tubes
  - B. Transistor
  - C. VLSI
  - D. IC's
38. Which of the following statement(s) is/are correct?
- I. An algorithm consists of series of steps to be performed to solve a problem.
  - II. To a given problem there may be more than one algorithm.
- A. I is correct
  - B. II is correct
  - C. I & II are correct
  - D. I & II are wrong
39. Base of decimal number system is \_\_\_\_\_.
- A. 2
  - B. 8
  - C. 10
  - D. 16
40. Base of binary number system is \_\_\_\_\_.
- A. 2
  - B. 8
  - C. 10
  - D. 16
41. What is the binary equivalent of decimal number 27?
- A. 11101
  - B. 10111
  - C. 11011

- D. 11110
42. RAM(random access memory)is \_\_\_\_\_memory
- A. Not volatile
  - B. volatile
  - C. write only
  - D. All of above
43. Modern computers use \_\_\_\_\_
- A. LSI /VLSI chip
  - B. Vacuum tubes
  - C. SSI chips
  - D. MSI chips
44. Primary storage is \_\_\_\_\_ as compared to secondary storage.
- A. Allow and inexpensive
  - B. Fast and inexpensive
  - C. Fast but expensive
  - D. Slow and expensive
45. Which of the following statements are related to the machine language?
- A. Difficult to learn
  - B. First generation language
  - C. Machine-dependent
  - D. All of above
46. Assembly languages consist of \_\_\_\_\_instructions
- A. Mnemonics
  - B. Opcodes
  - C. Operands
  - D. Fields
47. Pascal is a structured programming language, meaning that the flow of control is structured into standard statements except Statement
- A. if then else
  - B. for do
  - C. repeat Until
  - D. go to
48. \_\_\_\_\_ governs the sequencing of control through program
- A. Control structure
  - B. Control program
  - C. Control time
  - D. All of above
49. What is the name of the software that allows us to view web pages?
- A. Browser
  - B. Mail Client
  - C. FTP Client
  - D. Messenger

50. Which of the following is application software

- A. Tally
- B. AutoCAD
- C. MS-Office
- D. All of above

### Answers

1.B	2.A	3.D	4.A	5.A	6.C	7.D	8.C	9.B	10.B	11.B	12.B
13.A	14.C	15.A	16.D	17.A	18.D	19.B	20.B	21.D	22.A	23.A	24.A
25.A	26.D	27.A	28.D	29.B	30.B	31.A	32.B	33.A	34.D	35.B	36.D
37.A	38.C	39.C	40.A	41.C	42.B	43.A	44.C	45.D	46.A	47.D	48.A
49.A	50.D										

1. The keyword „void“ in function declaration indicates \_\_\_\_\_
  - A. The function will return ‚int‘ type of value
  - B. The function will return a default value
  - C. A function not returning any value
  - D. The function will return ‚void‘ type of value
2. The \_\_\_\_\_ format specification is used to write a long integer variable.
  - A. %d
  - B. %dd
  - C. %ld
  - D. %if
3. Constants in „C“ refer to \_\_\_\_\_
  - A. A fixed value that do not change during the execution of the program.
  - B. A fixed value that can change during execution of the program
  - C. A fixed value that can change after compilation of the program
  - D. A fixed value that can change after linking the program
4. Which of the following function is used to send the output to the console?
  - A. Scanf
  - B. Getch
  - C. Printf
  - D. Clrscr
5. =n „C“ a semicolon is used \_\_\_\_\_
  - A. To terminate a statement
  - B. To break a loop
  - C. To give a comment
  - D. None
6. If a=3, b=0 and c=4, what is the value of the expression a && b || c
  - A. 1
  - B. 2
  - C. 3
  - D. 4
7. The logical „OR“ operator is denoted by a \_\_\_\_\_ symbol in C program

- A. &&  
B. ||  
C. |  
D. &
8. Which one of the following is a logical operator?  
A. =  
B. &&  
C. <>  
D. +
9. Two – way selection is implemented using the \_\_\_\_\_ statement.  
A. If-else  
B. for  
C. switch  
D. Nested if else
10. „switch“ statement is used to make a decision \_\_\_\_\_  
A. To switch the processor to execute some other program  
B. Between two alternatives  
C. Amongst many alternatives  
D. None of these
11. The \_\_\_\_\_ format specification is used to read or to write a Short integer variable.  
A. %c  
B. %d  
C. %hd  
D. %f
12. ‘break’ statement is used \_\_\_\_\_  
A. To terminate a loop and execute the next statement  
B. To skip a loop and terminate the program  
C. To continue a loop and execute next statement  
D. Execute a next statement
13. Which of the following statements determines if the contents of string1 are same as string2?  
(Where string1 and string2 are well formed string.)  
A. if ( string1 == string2)  
B. if (string1, string2)  
C. if (strcmp (string1,string2) ==0)  
D. if (strcmp (string1, string2) <0)
14. The binary expressions are formed by an \_\_\_\_\_ Combination.  
A. Operand-operator-operand  
B. Operator-operand  
C. Operator-operand-operator  
D. Operand-operator
15. The ‘sizeof’ operator tells us the size of a type or a primary expression in terms of number of \_\_\_\_\_.  
A. Bytes  
B. Bits  
C. Nibbles  
D. Words
16. Which one of the following is an Arithmetic operator?  
A. \*  
B. ||

- C. &  
D. None above
17. The loop condition is tested at the \_\_\_\_\_ of the 'do while' construct  
A. Start  
B. End  
C. Middle  
D. Start & End
18. variables are named area of \_\_\_\_\_ that is used to hold data  
a.Memory location(s)  
b.Row and column number on a monitor  
c. Row and column number on a printer  
d.None of the above
19. In do-while loop, loop condition is checked at the \_\_\_\_\_.  
A. Beginning of loop  
B. End of loop  
C. End of program  
D. Start of program
20. Which of the following control structures are used in the iteration logic-  
A. if then if else  
B. do while repeat Until  
C. do & while  
D. do while if else
21. goto statement is used to –  
A. Pass the control anywhere in the program.  
B. Execute a statement for multiple statements.  
C. Execute a single statement from set of multiple statements.  
D. All of above
22. Switch statement allows us to –  
A. Make a decision from the number of choices.  
B. Execute a statement at least ones before checking a condition  
C. Execute a statement for multiple times  
D. None of the above
23. In case statement (case <xxx>)we can give  
A. Character or integer constant  
B. Expression with variable  
C. Character or integer variable  
D. All of the above
24. We can use \_\_\_\_\_ to perform a set of instructions repeatedly.  
A. Switch  
B. Loop  
C. Header file  
D. Conditional statement
25. =n the syntax “while ( xxx )” xxx denotes –  
A. Condition  
B. Statement  
C. Function  
D. Variable
26. Out of the following \_\_\_\_\_is a loop.  
A. Switch

- B. While  
C. Continue  
D. Break
27. The condition in a loop should become \_\_\_\_\_ Sometime, otherwise loop would be executed forever (infinite loop).  
A. False  
B. Equal  
C. True  
D. None
28. =n in the syntax “xxx(int a a 10 a++’ xxx means –  
A. While  
B. For  
C. If  
D. Switch
29. n “else” statement is always associate with –  
A. For  
B. While  
C. Case  
D. If
30. Statement should be ended with –  
A. Semicolon  
B. Full stop  
C. Hyphen  
D. Comma
31. Which loop executes statements within a loop at least ones?  
A. While  
B. Both A & C  
C. for  
D. do While
32. What does continue statement do?  
A. Take the control back to the starting of loop, bypassing the remaining statement.  
B. Executes all remaining statements concurrently.  
C. Break the loop and take the control outside of loop  
D. None of the above
33. Switch can be replaced by –  
A. For loop  
B. =f else statements  
C. While loop  
D. All above
34. While statement can be used to show menu at least ones in menu drive program.  
A. Do while  
B. While  
C. For  
D. =f Else
35. A block which accepts parameters and can return a value is called as –  
A. Loop  
B. Preprocessor  
C. Preprocessor  
D. Function

36. Select valid function call for function “void display() ,printf(“:ellow World”) -”  
A. display()  
B. Call display()  
C. Display;  
D. display();
37. Arguments/Parameters are use to –  
A. Get return value for a function  
B. Pass input value to a function  
C. To call a function  
D. Define a function
38. Any function by default return an \_\_\_\_\_ value.  
A. Int  
B. Char  
C. Float  
D. Double

**Answers**

1.C	2.C	3.A	4.C	5.A	6.C	7.A	8.B	9.B	10.A	11.C	12.C
13.A	14.C	15.A	16.B	17.A	18.B	19.A	20.B	21.B	22.A	23.A	
24.A	25.B	26.A	27.B	28.A	29.B	30.D	31.A	32.D	33.A	34.B	
35.A	36.D	37.D	38.B								

# C programming I & II

- Multiple choice & one line questions

1) The C language consist of ____ number of keywords. A] 32      B] 40      C] 24      D] 56
2) Which of the following is a keyword used for a storage class? A] Printf      B] external      C] auto      D] scanf
3) The prototype of the function in the header file is- A]Stdio.h      B] stdlib.h      C] conio.h      D] io.h
4) Preprocessor Directives are used for - A] Macro Expansion      B] File Inclusion C] Conditional Compilation      D] All of these
5) Which operator has the lowest priority ? A] ++      B] %      C] +      D]
6) The type cast operator is- A] (type)      B] cast()      C] //      D] “ “
7) File manipulation functions in C are available in which header file ? A] streams.h      B] stdio.h      C] stdlib.h      D] files.h
8) Which pair of functions below are used for single xharacter I/O ? A] getchar() and putchar()      B] scanf() and printf() C] input() and output()      D] Non of these
9) Qhich function is used to read character as you type ? A] getchar()      B] getch()      C] getche()      D] Both (B) and (C)
10) What is the output of this program ? void main() { int a=b=c=10; a=b=c=50; printf("\n %d %d %d",a,b,c); } A] 50 50 50      B] Compile Time Error      C] 10 10 10      D] Three Gaebage Value
11) Which format specifier is used to print the values of double type variable A]%If      B]%Id      C]%Iu      D] %f

12)	What will be the output of the following program?
	Void main () {
	Double x=28;
	Int r;
	R= x%5;
	Printf ("\n r=%d", r); }
	A] r= 3      B] Run time Error      C]Compile time Erroe      D]None of the
Above	
13)	What the follwing function call mean?
	Strcpy(s1 , s2 );
	A]copies s1 string into s2      B]copies s2 string into s1
	C]copies both s1 and s2      D] None of these
14)	What will be the output of the following program?
	Void main() {
	Int x []= {10,20,30,40,50};
	Print f ("\n %d %d %d %d ", x [4] ,3[x] ,x[2] ,1[x] ,x[0] ); }
	A]Error      B]10 20 30 40 50      C]50 40 30 20 10      D]None of these
15)	Which of the following is not s keyword of 'C' ?
	A]auto      B]register      C]int      D]function
16)	What will be the out put ?
	Void main () {
	Char a[] = "INFO" ;
	a + +;
	printf ("\n %s", a); }
	A] Error      B] INFO      C] NFO      D] None of these
17)	Which of the following operatorhas right to left associativity?
	A] &&      B] //      C] %      D] sizeof
18)	What wiil be the out put ?
	Void main () {
	Int I ;
	I=0x10+ 010+10;
	Printf ("\nx=%ox", i); }
	A] x= 34      B] i= 34      C] I = 22      D]Error
19)	Explicite type conversion is known as
	A] conversion      B] disjunction      C] separation      D] casting
20)	What will be the output ?
	#define SQUARE(X) X * X
	void main () {
	printf ("\n Square =%d", SQUARE(10+2)); }
	A] Square = 144      B] Square =32      C] Square =122      D]Square =12

<p>21) By default a function returns a value of type  A] int      B] char      C] void      D] None of these</p>
<p>22) What will be the value of x after executing the program ?  void main () {  int x;  x = printf("I See, Sea in C");  printf("\n x=%d", x); }  A] x= 15    B] x=2    C] Garbage value    D] Error</p>
<p>23) What is sizeof In 'C' ?  A] Operator    B] Reserve Worf    C] Both (A) and (B)    D] Function</p>
<p>24) Study the following C program  Void main () {  Int a=0;  For ( ; a );  A++; }  What will be the value of the variable a, on the execution of the above program  A] I    B] 0    C] -1    D] None of these</p>
<p>25) Which is not keyword in 'C' ?  A]typedef    B] const    C] near    D] complex</p>
<p>26) What will be the output of the following program code ?  void main () {  char a[]="Hello World";  char *p;  p=a;  printf("\n%d%d%d%d",sizeof(a), sizeof(p), strlen(a), strlen(p)); }  A] 11 11 10 10    B] 10 10 10 10    C] 12 12 11 11    D] 12 2 11 11</p>
<p>27) The meaning of arrow operator in a-&gt;b  A] (*a).b    B] a.(*b)    C] a.b    D] None of these</p>
<p>28) What will be the output of the following program code?  Void main () {  Printf("\n ABC\b\b\bInfo World"); }  A] Info world    B] ABC Info world    C] strxfrm    D] strcut</p>
<p>29) Which is valid string function ?  A] strpbrk    B] strlen    C] strxfrm    D] strcut</p>

<p>30) What will be the size of following structure?</p> <pre>Struct sample {     Static int x;     int y,z; } ; A]6 bytes      B] 2 bytes      C] + bytes      D] None of these</pre>
<p>31) Which of the following function not convert floating point number to string ?</p> <p>A] fcvt      B] gevt      C] eevt      D] hcvt</p> <p>32) What will be the output ?</p> <pre>void main () {     printf("%d",'B' &lt; 'A' ); A] Error      B] 1      C] 0      D] None of these</pre>
<p>33) Which one of the following is condirional directive ?</p> <p>A] #nifdefn      B] #ifdefn      C] # ifdefn      D] #nifdef</p>
<p>34) What will be the output ?</p> <pre>void main () {     int x;     unsigned y;     printf("\n%d %d", sizeof(x), sizeof(y) ); A] 22      B] 24      C] 44      D] None of these</pre>
<p>35) int **x;</p> <p>A]x is a pointer to pointer      B] x is not pointer      C] x is long      D] None of these</p>
<p>36) What will be the output ?</p> <pre>void main () {     printf("\n %d %d", 10&amp;20, 10/ 20); } A] 00      B] 10 10      C] 0 30      D] 20 20</pre>
<p>37) Which of the following is used as a string termination character ?</p> <p>A] 0      B] \0      C] /0      D] None of these</p>
<p>38) What will be the output ?</p> <pre>void main () {     int I= 48;     printf("\n %c %d" ,I,I); } A] Error      B] 48 48      C] 1 48      D] 0 48</pre>
<p>39) A static variable by default gets initialized to</p> <p>A] 0      B] blank space      C] 1      D] garbage value</p>

40) Find out on which line no . you will get an error ?

Line 1: void main ( )

Line 2: {

Line 3: print("\n Hello World")

Line 4: }

- A] Line 1      B] Line 2      C] Line 3      D] Line 4

41) What will be the output of the following program ?

void main ( ) {

int x=10,y=20;

printf ("\n %d",x,y); }

- A] 10      B] 20      C] 10 20      D] None of these

42) Which function reallocates memory ?

- A] realloc      B] alloc      C] malloc      D] None of these

43) What will be the size of following union declaration?

Union Test {

Int x;

Char y;

Float z; } ;

- A] 7 bytes      B] 4bytes      C] 1byte      D] 4 bytes

44) A declaration float a,b; occupies \_\_\_\_\_ of memory ?

- A] 1 bytes      B] 4bytes      C] 8byte      D] 16 bytes

45) What is the output of the following program ?

void main() {

int x=40;y=30;z=80;

if(x<y<z)

printf("\n Hello world");

else

printf("\nGood by");

- A] Hello world      B]Good by      C]Compile time error      D]None of these

46) Which of the following is not a relational operator?

A]!

B] !=

C]>=

D]<

47)	what will be the output ? void main(){ char *p="Hello world"; int *q; p++; q = (int *)p; q++; printf("\n %s\n%s,p,q); } A] ello world Ello world C] ello world Lo world	B]Error D]ello world llo world
48)	which of the following is an operator in 'C'? A], B] \$ C] @ D] None of these	
49)	What is the output of the following code? Void main() { Int c=0, d=5,e=10,a; A=c>1?d>1  e>1?100:200:300; Printf("a=%d",a); } A] a=300 B]a=100 C] a=200 D] None of these	
50)	Which among the following is a unconditional control structure? A] do-while B] if –else C] goto D] for	
51)	Which of the following language is predecessor to C Programming Language? A] A B]B C] BCPL D]C++	
52)	C programming language was developed by A] Dennis Ritchie B]Ken Thompson C] Bill Gates D] Peter Norton	
53)	C was developed in the year ____ A] 1970 B] 1972 C] 1976 D] 1980	
54)	C is a ___ language A] High Level B] Low Level C] Middle Level D] Machine Level	
55)	C language is available for which of the following Operating Systems? A] DOS B] Windows C]Unix D] All of these	
56)	Which of the following symbol is used to denote a pre-processor statement? A] ! B]# C] ~ D];	
57)	Which of the following is a Scalar Data type A1 Float B1 Union C1 Array D1 Pointer	

58) Which of the following are tokens in C?
A] Keywords B] Variables C] Constants D] All of the above
59) What is the valid range of numbers for int type of data?
A] 0 to 256 B] -32768 to +32767 C] -65536 to +65536 D] No specific range
<b>60)</b> Which symbol is used as a statement terminator in C?
A] ! B] # C] ~ D] ;
<b>61)</b> Which escape character can be used to begin a new line in C?
A] \a B] \b C] \m D] \n
<b>62)</b> Which escape character can be used to beep from speaker in C?
A] \a B] \b C] \m D] \n
<b>63)</b> Character constants should be enclosed between _____
A] Single quotes B] Double quotes C] Both a and b D] None of these
<b>64)</b> String constants should be enclosed between _____
A] Single quotes B] Double quotes C] Both a and b D] None of these
<b>65)</b> Which of the following is invalid?
A] '' B] "" C] 'a' D] 'abc'
<b>66)</b> The maximum length of a variable in C is _____
A] 8 B] 16 C] 32 D] 64
<b>67)</b> What will be the maximum size of a float variable?
A] 1 byte B] 2 bytes C] 4 bytes D] 8 bytes
<b>68)</b> What will be the maximum size of a double variable?
A] 1 byte B] 4 bytes C] 8 bytes D] 16 bytes
<b>69)</b> A declaration float a,b; occupies _____ of memory
A] 1 byte B] 4 bytes C] 8 bytes D] 16 bytes
<b>70)</b> The size of a String variable is
A] 1 byte B] 8 bytes C] 16 bytes D] None
<b>71)</b> Which of the following is an example of compounded assignment statement?
A] a = 5 B] a += 5 C] a = b = c D] a = b
<b>72)</b> The operator && is an example for _____ operator.
A] Assignment B] Increment C] Logical D] Rational
<b>73)</b> The operator & is used for
A] Bitwise AND B] Bitwise OR C] Logical AND D] Logical OR
<b>74)</b> The operator / can be applied to
A] integer values B] float values C] double values D] All of these
<b>75)</b> The equality operator is represented by
A] := B] .EQ. C] = D]==

**76)** Operators have hierarchy. It is used to know which operator  
A] is most important B] is used first C] is faster D] operates on  
large numbers

**77)** The bitwise AND operator is used for  
A] Masking B] Comparison C] Division D] Shifting bits

**78)** The bitwise OR operator is used to  
A] set the desired bits to 1 B] set the desired bits to 0 C] divide  
numbers  
D] multiply numbers

**79)** Which of the following operator has the highest precedence?  
A] \* B] == C] => D] +

**80)** The associativity of ! operator is  
A] Right to Left B] Left to Right C] (a) for Arithmetic and (b) for  
Relational  
D] (a) for Relational and (b) for Arithmetic

**81)** Which operator has the lowest priority?  
A]++ B]% C]+ D]||

**82)** Which operator has the highest priority?  
A]++ B]% C]+ D]||

**83)** Operators have precedence. A Precedence determines which  
operator is  
A]faster B] takes less memory C] evaluated first D]takes no  
arguments

**84)** Integer Division results in  
A] Rounding the fractional part B] truncating the fractional part  
C] Floating value D]An Error is generated

**85)** . Which of the following is a ternary operator?  
A] ? B] \* C] sizeof D]^

**86)** What will be the output of the expression  $11 \wedge 5$ ?  
A]5 B]6 C] 11 D]None of these

**87)** The type cast operator is  
A] (type) B]cast() C]// D] " "

**88)** Explicit type conversion is known as  
A] Casting B] Conversion C] Disjunction D] Separation

**89)** The operator + in  $a+=4$  means  
A]  $a = a + 4$  B]  $a + 4 = a$  C]  $a = 4$  D]  $a = 4 + 4$

**90)**  $p++$  executes faster than  $p+1$  because  
A] p uses registers B]  $p++$  is a single instruction C]  $++$  is faster  
than +  
D] None of these

91) Which of the following statements is true?
A] C Library functions provide I/O facilities B] C inherent I/O facilities C] C doesn't have I/O facilities D] Both (a) and (c)
92) Header files in C contain
A] Compiler commands B] Library functions C] Header information of C programs D] Operators for files
93) Which pair of functions below are used for single character I/O.
A] Getchar() and putchar() B] Scanf() and printf() C] Input() and output() D] None of these
94) The printf() function retunes which value when an error occurs?
A] Positive value B] Zero C] Negative value D] None of these
95) Identify the wrong statement
A] putchar(65) B] putchar('x') C] putchar("x") D] putchar('\n')
96) Which of the following is character oriented console I/O function?
A] getchar() and putchar() B] gets() and puts() C] scanf() and printf() D] fgets() and fputs()
97) The output of printf("%u", -1) is
A] -1 B] minimum int value C] maximum int value D] Error message
98) An Ampersand before the name of a variable denotes
A] Actual Value B] Variable Name C] Address D] Data Type
99) Symbolic constants can be defined using
A] # define B] const C] symbols D] None of these
100) Null character is represented by
A] \n B] \0 C] \o D] \e
101) Which header file is essential for using strcmp() function?
A] string.h B] strings.h C] text.h D] strcmp.h
102) malloc() function used in dynamic allocation is available in which header file?
A] stdio.h B] stdlib.h C] conio.h D] mem.h
103) File manipulation functions in C are available in which header file?
A] streams.h B] stdio.h C] stdlib.h D] files.h
104) C supports how many basic looping constructs
A] 2 B] 3 C] 4 D] 6
105) A statement differs from expression by terminating with a
A] ; B] : C] NULL D] .
106) What should be the expression return value for a do-while to terminate

A] 1 B] 0 C] -1 D] NULL

107) Which among the following is a unconditional control structure  
A] do-while B] if-else C] goto D] for

108) continue statement is used  
A] to go to the next iteration in a loop  
B] come out of a loop  
C] exit and return to the main function  
D] restarts iterations from beginning of loop

109) Which operator in C is called a ternary operator  
A] if..then B] ++ C] ? D] ()

110) Which of the following header file is required for strcpy()  
function?  
A ] string.h B] strings.h C] files.h D] strcpy()

111) The meaning of conversion character for data input is  
A] Data item is a long integer  
B] Data item is an unsigned decimal integer  
C] Data item is a short integer  
D] None of the above

112) The conversion characters for data input means that the data  
item is  
A] An unsigned decimal integer  
B] A short integer  
C] A hexadecimal integer  
D] A string followed by white space

113) An expression contains relational, assign. ment and arithmetic  
operators. If Parenthesis are not present, the order will be  
A] Assignment, arithmetic, relational  
B] Relational, arithmetic, assignment  
C] Assignment, relational, arithmetic  
D] Arithmetic, relational, assignment

114) Which of the following is a key word is used for a storage class  
A] printf B] external C] auto D]scanf

115) In the C language 'a' represents  
A] a digit B] an integer C] a character D] a word

116) The number of the relational operators in the C language is  
A] Four B] Six C] Three D] One

117) A compound statement is a group of statements included  
between a pair of  
A] double quote B] curly braces C] parenthesis D] a pair of /'s

118) A Link is  
A] a compiler B] an active debugger C] a C interpreter D] a  
analyzing tool in C

- 119) The continue command cannot be used with  
A] for B] switch C] do D] while
- 120) In C, a Union is  
A] memory location B] memory store C] memory screen D] None of these
- 121) When the main function is called, it is called with the arguments  
A] argc B] argv C] None of these D] both a & b
- 122) A multidimensional array can be expressed in terms of  
A] array of pointers rather than as pointers to a group of contiguous array  
B] array without the group of contiguous array  
C] data type arrays  
D] None of these
- 123) C allows arrays of greater than two dimensions, who will determine this  
A] programmer  
B] compiler  
C] parameter  
D] None of these
- 124) A pointer to a pointer in a form of  
A] multiple indirection B] a chain of pointers C] both a and b D] None of these
- 125) Pointers are of  
A] integer data type B] character data type C] unsigned integer data types  
D] None of these
- 126) Maximum number of elements in the array declaration int a[5][8] is  
A] 28 B] 32 C] 35 D] 40
- 127) If the size of the array is less than the number of initializers then,  
A] Extra values are being ignored  
B] Generates an error message  
C] Size of Array is increased  
D] Size is neglected when values are given
- 128) Array subscripts in C always start at  
A] -1 B] 1 C] 0 D] Value provided by user
- 129) A Structure  
A] can be read as a single entity  
B] cannot be read as a single entity  
C] can be displayed as a single entity  
D] has member variables that cannot be read individually

- 130) Identify the invalid pointer arithmetic**
- A] Addition of float value to a pointer
  - B] Comparision of pointers that do not point to the element of the same array
  - C] Subtracting an integer from a pointer
  - D] Assigning the value 0 to a pointer variable

### One Line Questions

1. What is variable?
2. What is constant?
3. How many bytes are required to store integer type value?
4. How many bytes are required to store float type value?
5. How many bytes are required to store char type value?
6. How many bytes are required to store double type value?
7. What is main difference between variable and constant?
8. What is logical variable?
9. What is global variable?
10. How long is word?
11. How long is a byte?
12. How does a programmer finds coding errors?
13. Describe the appearance of machine code?
14. Whether the program in c can be executed by computer directly ?
15. What is language processor?
16. What is purpose of language processor?
17. What are major disadvantages of machine code?
18. Give the general syntax of conditional operator?
19. Which are relational operator?
20. Which are logical Operators?
21. Which are Bitwise Operators?
22. Which are unformatted input output functions?
23. Which are formatted input output functions?
24. What is the use of getchar() function?
25. What is the use of getch() function?
26. What is the use of getche() function?
27. What is Disk IO Function?
28. What do you mean by consol IO functions?
29. Give syntax of simple if stmt
30. Give syntax of simple if – else stmt
31. Give syntax of simple nested if – else stmt?
32. Define Program
33. What is nested loop?
34. What is process loop?
35. What is Syntax Error?
36. What is Logical Error?
37. What is Run Time Error?

38. Define Array
39. Give general Syntax to declare One dimensional array
40. Give general Syntax to declare two dimensional array
41. What is function?
42. What is built in function?
43. What is use of return statement?
44. What is the use of strcat() function?
45. What is the use of strcmp() function?
46. What is the use of strrev() function?
47. What is the use of strlen() function?
48. What is the use of strcpy() function?
49. What is recursive function?
50. What do you mean by call by value?
51. What do you mean by call by reference?
52. What is pointer?
53. What is structure?
54. What is main difference between structure and union?
55. What is use of typedef?
56. Whether Structured programming helps in reducing errors?
57. Give the syntax of defining rectangle()
58. Give the syntax of defining putpixel()
59. Give the syntax of defining ellipse()
60. Give the syntax of defining line()
61. Give the syntax of defining arc()
62. What is preprocessor ?
63. Give any two features of preprocessor
64. Give the syntax for defining File
65. Give the syntax for opening File
66. Give the syntax for Closing File
67. What is fopen()?
68. What is fclose()?
69. What is the use of ftell()?
70. What is the use of fseek()?
71. What is the use of rewind()?
72. What is the use of feof()?

- Question for two marks

1. What are the three constants used in C?
2. Explain bitwise left shift operator?
3. What is unary operator?
4. Explain putchar()?
5. What is an expression? How is an expression differing from variables?
6. Explain primary data types used in C?
7. Comment “C is mid level language”?

8. Explain escape sequence character in C?

9. What is use of if statement?

10. write a syntax of while loop?

11. what is output of following program?

```
int m=1,n=2;
for(j=1;j<=2;j=j+1)
{ m=m+1;
  n=n*j;
  printf("%d \t %d\t",m,n);
}
```

12. which of different types of loop statement used in C?

13. Explain for loop?

14. Enlist different format specifier in C?

15. What is an array?

16. What is output of following program?

```
main()
{
  int a[7]={11,12,13,14,15,16,17};
  int i;
  printf("content of array");
  for(i=0;i<=6;++i);
  { printf("%d\t",a[i]); }
}
```

17. What are the rules to declare one dimensional array?

18. What is multidimensional array?

19. explain- a) strlen()    b) strcat()

20. What is function?

21. Explain recursion?

22. What is call by value?

23. Explain any two string functions?

24. What is variable?

25. Write disadvantages of goto statement.

26. What is pointer?

27. Difference between union and structure.

28. What is structure?

29. explain – a) line()    b) circle()

30. Explain any two graphics function?

31. Explain -    a) getc()    b) putc()

32. Explain any two file handling function?

33. Explain macros?

34. Explain #include?

35. Explain dynamic memory allocation?

36. State three advantages of function?

37. What is purpose of keyword void?

38. Determine the value of each of following expression-

```
int i=8 , j=j ; a) (3 * i - 2 * j ) %( 2 * d -c) b) ( i > 0) && (j < 5)
```

39. What is mean by associativity?

40. If  $x = 8.8, y = 3.5, z = -5.2$ , then determine value of following expression
  - a)  $2 * y + 3 * (x - z)$
  - b)  $2 * x / (3 * y)$
41. Describe arithmetic operator?
42. Explain `scanf()`;
43. Difference between formatted & unformatted statement ?
44. State features of pre-processor?
45. What is command line argument?
46. Define variable & constant?
47. How static variable are define and initialized?
48. What is mean by storage class of variable?
49. List any three file mode in C?
50. Write a program to find largest between two numbers?

## Questions for 4 marks

1. Enlist the features of C.
2. Explain different data types used in C language?
3. Explain type identifiers in C?
4. Explain in brief structure of C programming?
5. what is operator enlist all operators used in C?
6. what is data type explain the any four data types used in C language?
7. Explain the difference between '=' and '==' operator explain with example?
8. Write a short note precedence & order of evaluation?
9. Differentiate between relational and logical operators used in C?
10. Write short note on Input & Output functions used in C (i.e. `print` & `scanf` functions?)
11. what is variable? What are the rules for defining variables?
12. Differentiate between local variable and global variable?
13. Explain symbolic constants used in C?

14. Explain any two bitwise operator with suitable example.
15. Explain with example `++i` and `i++`.
16. Explain logical operators and expressions used in C?
17. Explain the following g functions
  - i) `getch()` and ii) `clrscr()`
18. Explain `printf()` function with an example
19. Explain `scanf()` function with an example
20. Explain syntax and use of `Do_While` statement
21. Which looping statements does C provides? Explain any one.
22. Explain explain **continue** And **break** statements
23. Explain switch statement with its syntax and example.
24. What is Nested if else explain with an example?
25. Explain nested for loop with an example
26. What is array? How to declare array? Explain with suitable example.
27. Expalin one dimensional array with an example
28. Explain Two dimensional array with an example
29. Explain applications of array
30. Explain any 4 string functions with suitable example?
31. What is the difference between call by value and call by reference
32. What is recursion explain with suitable example.
33. Explain Automatic storage class specifier

34. Explain Static storage classs
35. Explain Register storage class
36. Explain Extern storage class
37. For what purpose '\0' is used in string operations explain with suitable example.
38. What is function ?how function is defined.
  
39. Explain the difference between calling function and called function?
40. Explain void function?
41. Explain what is pointer?expalin with suitable example
42. Explain pointer to structure in detail.
43. Explain pointer to function in detail.
44. Explain explain any one dynamic memory allocation
45. Explain how to access a value using pointer?give a suitable example.
46. Write a short note on pointer to pointer
47. Distinguish bet malloc and calloc()
48. What is structure?explain with suitable example
49. Explain array of structure with example
50. Explain Nested structure with example.
51. Distinguish between Structure and Union
52. Differentiate structure and array

53. Describe programming approach
54. Explain the use of Typedef
55. What is graphics? Enlist the different types of functions used in C
56. Explain the following graphic function by writing a small program **circle()**
57. Explain the following graphic function by writing a small program **arc()**
58. Explain the following graphic function by writing a small program  
**ellips()**
59. Explain the following graphic function by writing a small program  
**initgraph()**
60. Explain the following graphic function by writing a small program  
**line()**
61. Explain the following graphic function by writing a small program  
**rectangle()**
62. Explain directives
63. Explain features of preprocessors
64. What is a pre-processor explain #include ,#define

65. Write a short note on C preprocessors
66. Explain i)fprintf() ii)fscanf()
67. Distinguish between i)fprintf() ii)fscanf()
68. Write a short note on file handling in C
69. Explain the following i)rewind ii)feof
70. Distinguish between getch and getc
71. Distinguish between putch and putc
72. Explain putc and getc in brief
73. What is command line argument
74. Explain command line argument with example
75. Distinguish between fprintf() and printf()
76. Explain fprintf() with example
77. Explain fscanf() with example
78. What is macro? Explain with example
79. Differentiate between if-else-if and switch statement.
80. Explain function with argument and return type.
81. Explain array of pointers.
82. Define array and how two – dimensional array is initialize?
83. Explain dynamic memory allocation in brief.
84. Write a note on pointer to pointer.
85. What are the similarities and difference between structure and union.
86. Explain nested structure.

87. Explain array of structure.
88. Explain pointer to structure.
89. Explain the following types of file
  - i)sequential
  - ii)Index sequential
  - iii)Direct file
90. What is advantage of representing an array of string by an array of pointer to string.
91. Explain the following functions with example
  - 1)getchar()
  - 2)putchar()
92. Explain the following functions with example
  - 1)getche()
  - 2)putche()
93. Explain the following functions with example
  - 1)getch()
  - 2)putch()
94. Explain sizeof operator with example.
95. Explain conditional operator with example.
96. What is user defined functions and built-in functions. Enlist them.
97. What is null string ?What is it's length?
98. What is union?Explain with example.
99. Write short note on expression used in C.

100. What are static variable? Compare with standard local variable.
101. Write a rule for declaring character constant.
102. Write a rule for declaring string constant.
103. Write a rule for declaring numeric constant.
104. What is structure? Explain with example.
105. Explain \* operator and & operator with example.
106. What are the rule of to use period(.) operator.
107. Explain is EOF and BOF.
108. What is EOF and what value does usually have ?
109. What are identifier and keywords? Explain it with suitable example.
110. What is type casting? Explain it with suitable example.
111. What is swaping? Explain it with suitable example.
112. Write a short note on ternary operator and cast operator.
113. What is string constant?How is string constant is differ from character constant?
114. What is character constant? How is character constant is differ from integer constant?
115. List out the five arithmetic operators in C.
116. What is the associativity rules involve in this operator.
117. What is mean by the comparision and logical operator?How are they different from the arithmetic and assignment operator?

118. List out the different operators involve for comparision and logical decision making in C.
119. What is mean by the equality operator? How do these differ from an assignment operator.
120. Explain the following bitwise operators  
i)Bitwise AND ii) Bitwise OR iii) BitwiseXOR iv) Bitwise Left Shift v)  
Bitwise Right Shift
121. What is unary operator? List out the different operator involve in the unary operator.
122. Distinguish between binary minus and unary minus.
123. What is modulus operator and how does it operate in C.
124. What is an expression? How is an expression different from the variables?
125. What are the different type of statement used in C.
126. What are the salient features of standard input and output file
127. Explain the following stements:  
i)getchar() ii)putchar() iii)EOF
128. What is the scanf() and how does it differ from the getchar().
129. What are the format codes used along with the scanf().Display the various data types in C.
130. What is the printf() and compare with putchar().
131. What is mean by conditional expression?
132. What is looping in C? What are the advantages of looping?

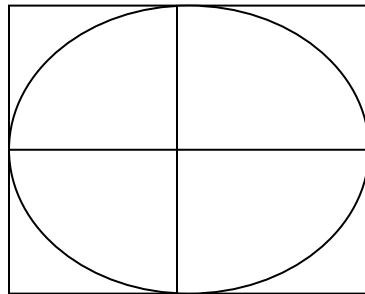
133. What is the nested for loop?
134. Compare while loop and for loop with example.
135. )What is crucial importance of main() in C.
136. What is use of continue in C.
137. List out applications of C language.
138. List out the advantages of function.
139. What is mean by call by reference & call by value.
140. What is the difference between call by reference & call by value.
  
141. What is the purpose of return statement
142. List out the rules used in return statement
143. What is mean by register variable and what the scope of it?
144. What role does the fseek() plays and how many arguments does it  
Have?
  
145. What is static variable and what is its scope?
146. What is the use of external data type in c?
147. What is the storage class used in recursive function
148. What is the recursive function. List out their merits and demerits.
  
149. How does the fopen() works?Explain it with example.
150. What is an array and how array variable differs from ordinary variable.

151. What is an array indexing explain with an example
152. When sorting the elements of an array is it necessary to use another array to store the sorted elements explain?
153. What is the function and list out advantages and disadvantages of functions
154. What is mean by function argument, function call and return value
155. What is the automatic variable and what is the use of it.
156. )How can data be initialized in the automatic variable
157. How are the data elements initialized in the case of static type variable
158. What is the use of external data type in C
159. How is the #include directive is used?
160. How can #define directive be continued to anew line
161. What are the rules used to declare a one dimensional array
162. What are the rules used to declare a two dimensional array
163. What are the rules used to declare a multi dimensional array
164. What is character array how it differs from other data types.
165. Distinguish between character array and string
166. Explain applications of array
167. What is a pointer?What is the use of pointer in C
168. What is the role played by the break statement within the switch statement.Explain with example.

169. What is the difference between the array of pointer and pointer to the array
170. Summerise the purpose of string.h function.
171. What is the structure and what are the uses of it.
172. Distinguish structure data type with other data type variables.,
173. How structure different from array
174. What is mean by member or field of structure
175. What is the difference between structure declaration and structure initialization
176. What is the advantage of UNION in C?
177. Explain the salient features of typedef ?
178. Explain the various modes used in file operation?
179. Comment on “May comments are nested”?
180. Distinguish between binary and unary minus with example
181. What is the modulus operator and how does it works explain it with example
182. Why is go to not necessary for the structured programming language like C?
183. What is the purpose of comma operator within which statement does the comma operator usually appear.
184. Explain Getw() & Putw function

\*\*\*\*\* Programming Segment \*\*\*\*\*

1. Write a C program to find the maximum of three numbers using conditional operators
2. Write a C Program to sort an array in ascending order
3. Write a C Program to sort an array in descending order
4. Write a C Program to find sum of digits in a given number
5. Write a C Program to print square of all numbers 1 to 20 and print sum squares
6. Write a C Program to check if given number is present in an array or not
7. Write a C Program to find the position of given number in array
8. Write a C Program to print transpose of matrix
9. Write a C Program to print equivalent binary number of given decimal number
10. Write a C Program to print equivalent octal number of given decimal number
11. Write a C Program to print equivalent hex number of given decimal number
12. Write a C Program to draw a circle with radius 10
13. Write a C Program to calculate factorial of a given number using recursion
14. Write a C Program to draw star symbol at the center of the screen
15. Write a C Program to copy contents of text.dat file to txt2.data file
16. Write a C Program to draw following object



17. Write a C Program to print all numbers between 1 to n divisible by 7
18. Write a C Program to find sum of  $1 + 2 + 3 + \dots + n$
19. Write a C Program to find sum of  $2 + 4 + 6 + \dots + n$
20. Write a C Program to find sum of  $7 + 14 + 21 + \dots + n$
21. Write a C Program to find sum of  $1/1 + 1/2 + 1/3 + \dots + 1/n$
22. Write a C Program to print 15 terms of 1, 2, 4, 7, 11, 16, ....
23. Write a C Program to print even and odd number from an array
24. Write a C Program to read character from keyboard and display message whether character is alphabet , digit or special symbol
25. Write a C Program to read a string and count number of vowels in it

26. What will be output of the following code or find errors if present in the code

a)

```
main()
{
    Int = 10 ;
    Char = "d" , i
    K=pow(5,6);
    I+=k;
}
```

b) j = 2 ;

```
nbegin=10; n = 0; nend = 3;
for (i=0;i<=nend;i++)
{
    n=nbegin+i*nend;
    printf("%d",n+j);
    j++;
}
```

c) main()

```
{
    int x , y ;
    x = 753;
    y = 722;
    printf(" x & y is %@ " , x & y );
}
```

d) main()

```
{
    char c[2] = "A";
    printf("\n%c",c[0]);
    printf("\n%s",c);
}
```

e) main()

```
{
    int b[] = { 10,20,30,40,50 };
    int i, *k;
    k = b;
    for ( i = 0 ; i<= 4 ; i++)
    {
        printf("\n%d" , *k);
        k++;
    }
}
```

```
f) main()
{
    int *a , b = 30;
    a = &b;
    b = *a + 40;
    a = b % 5;
    printf("%d %d",*a , b);
}
```

```
g) main()
{
    int i , j = 3;
    xyz(&i,&j);
    printf("%d%d",i,j);
}
```

```
xyz(int *i , int*j)
{
    *i = *i * *j;
    *j = *j * *j;
}
```

```
h) main()
{
    int x , y;
    x = 2003;
    x++;
    y = x++;
    y = x;
    y++;
    x--;
    x--
    printf ("%d%d",x,y);
}
```

```
i) main()
{
    int x ;
    x = 18;
    while ( x > 1)
    {
        printf("\n%d",x);
        x=x-1;
    }
}
```

```

j) main()
{
    int a , x;
    a = 18;
    x = a >> 1;
    printf("%d%d",a,x);
}

k) int i = 10 j = 20 ;
    float a,b,c;
    a = i / j;
    b = 1.0 * i / j ;
    c = i / j * 1.0;
    printf("%f %f %f ",a,b,c);

l) int i = 4 , j ;
    j = ++i * i++;
    i*=j;
    printf("%d %d",i,j);

m) typedef int integer;
    integer m , n;
    while ( n != 0)
    {
        m = n mod 5;
        printf("%c",m);
        n /== 10;
    }

n) main()
{
    int i = 0 , sum =0 , sum_sq = 0;
    for ( i = 2; i < 10 ; i+=2)
    {
        sum+=i;
        sum_sq+=i*i;
    }

    printf("Sum is : %d ", sum);
    printf("Sum of Square is : %d ", sum_sq);
}

o) main()
{
    static int i;
    printf("%d",i);
}

```

```

p) main()
{
    int x = 4 ;
    int y = x << 1;
    printf("%d%d",x,y);
}

q) main()
{
    int a;
    a = 30;
    a = a << 2 ;
    printf("a = %d",a);
}

r) main()
{
    int i = 3;
    i++;
    printf("Multiplication is %d",i++ * i++)
}

s) main()
{
    char x[] = "F.Y. B.Sc. ", *p;
    int i = 0;
    p = x;
    while ( i != 10)
    {
        i = i + 2;
        p++;
        printf("%c",*p);
    }
}

```

## Question for 6 marks

1. Write a C language program to display the largest element in the matrix.

2. Write a C language program to swap two numbers using pointers and function.
3. Write a C language program to calculate the series-  
 $1/1! + 2/2! + 3/3! + \dots$  Up to n terms.
4. Write down C language program to find out number of occurrences of a character in a file.
5. Write a C language program to display the student result sheet using the data stored in a file.

Student      structure

Name	character(25)
Rollno	integer
Marks1	integer
Marks2	integer
Marks3	integer

6. Write a C language program to display the content of file using command line argument.
7. Write a C language program to copy the contents of one file to another file.
8. Write a C language program to count number of lines and words in a file.
9. Write a C language program to find out sum of the following series  
 $1! + 2! + 3! + \dots + n!$
10. Write a C language program to enter n elements in array and find second smallest number from an array.
11. Write a C language program to check whether given number is prime or not.
12. Write a C language program to create a file stud.dat

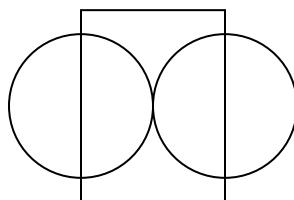
Field name      type

Rollno	integer
Name	character(25)
Age	integer

Enter information about 10 students.

13. Write a C language program using recursive function to enter 4 digit number and find the sum of all digits of the number .
14. Write a C language program to print all armstrong numbers between 1 to 500.  
(e.g.  $153 = 1^3 + 5^3 + 3^3 = 153$ )

15. Write a C language program to find whether given number is palindrome or not.
16. Write a C language program to find GCD of given two numbers.
17. Write a C language program which will read string and count the number of characters and words in it.
18. Write a C language program to read two matrices and add them.
19. Write a C language program to read two matrices and multiply them.
20. Write a C language program to read one matrix and find the sum of its diagonal elements.
21. Write a C language program to input number and find a largest digit in a given number and print it in word with appropriate message.  
(e.g. n=5273 - “SEVEN is largest”)
22. Write a C language program to compute following series  
 $G = 1 + x^3/3! + x^5/5! + x^7/7! + \dots$  up to n terms.
23. Write a C language program to draw ‘+’ sign at the center of the screen.
24. Write a C language program to read n numbers in an array and split the array into two arrays even and odd such that the array even contains all the even numbers and other is odd. So the output will be—  
(e.g. Original array is 7,9,4,6,5,3,2,10,18  
Odd array is 7,9,5,3  
Even array is 4,6,2,10,18 )
25. Write a C language program to check whether the string is palindrome or not.
26. Write a C language program to display—



27. Write a C language program to read records from file created in binary mode.
28. Write a C language program to create file in binary mode to store students record.
29. Write a C language program to add, list, delete record and modify the current record.

30. Write a C language program to read “mark.dat” file containing rollno ,name,marks of three subjects and calculate total mark, result in grade and store same in “result.dat” file.

31. Write a C language program to create a file “ele.dat” containing cust no,name,current & previous reading.

32. Write a C language program to read a cust.dat file containing meter number ,name, current reading & previous reading. Read the same file calculate unit and total amounts according to the following rules—

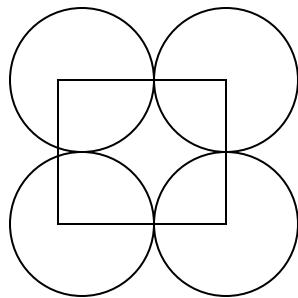
Unit	rate
------	------

0-50	1.00
51-100	1.50
> 100	2.00

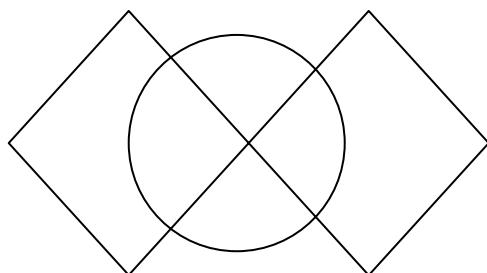
Store meter number ,name ,unit & amount in master.dat file.

33. Write a C language program to create file “odd” to store all odd numbers between 1 and n.

34. Write a C language program to draw –



35. Write a C language program to draw –



36. Write a C language program using structure to define employee record containing employee number , name and salary. Read 10 records.

37. Write a C language program to demonstrate the use of union.

38. Write a C language program to define structure for class containing class, name, no. of students and block no. Read 5 records and display it.
39. Write a C language program using command line argument to add three numbers.
40. Write a C language program using command line argument to calculate area of a rectangle.
41. Write a C language program using recursion n terms of Fibonacci series.
42. Write a C language program using recursion to calculate  $m^n$ .
43. Write a C language program using recursion to calculate factorial of given number.
44. Distinguish between character constant and string constant.
45. Describe all operators used in C language with example.
46. Explain in detail three parts of C program.
47. Write a short note on precedence and order of evolution.
48. Explain in detail bitwise operators with example.
49. State and explain formatted input-output statements and standard input-output statements with example.
50. Explain in detail call by value and call by reference with example.
51. Explain the following using general syntax and example .  
i) if            ii) if-else            iii) nested if-else
- 52 . Explain break and continue statements using syntax and example .
53. Explain following  
i) while        ii) do-while    iii) for
54. Define array. Explain different types of array in detail.
55. State and explain various types of standard function with example.
56. State and explain different phases used in user defined function.
57. Explain function with return and function with arguments with example.
58. State and explain different types of string functions with example.
59. Explain dynamic memory allocation and releasing dynamically allocated memory.
60. Define structure and union. Explain the way of declaring and accessing them.
61. Explain nested structure and self referential structure with example.
62. Explain in detail array of structure and pointer to structure.
63. State and explain various modes of file opening and file closing.
64. What do you mean by pre-processor? Explain in detail macros.

65. What do you mean by pre-processor directives? List and explain its different categories.
66. Explain following in detail
  - i) initgraph()
  - ii) circle()
  - iii) arc()
  - iv) ellipse()
  - v) line()
  - vi) rectangle()
67. Explain the concept of files,records and fields.
68. Explain any three of the following with example
  - i) fprintf()
  - ii) fscanf()
  - iii) getc()
  - iv) feof()
69. Explain any three of the following with example
  - i) getw()
  - ii) putw()
  - iii) feof()
  - iv) fgets()
70. Explain any three of the following with example
  - i) rewind()
  - ii) fseek()
  - iii) ftell()
  - iv) fputs()
71. Write a C language program to enter n elements in array and find second largest number from array.