

FYBCA – Semester II
US02CBCA01 (Advanced C Programming and Introduction to Data Structures)
Question Bank

Multiple Choice Questions**Unit-1**

1. Which operator is used with a pointer to access the value of the variable whose address is contained in the pointer?

A. Address (&) C. Indirection (*)
B. Assignment (=) D. Selection (->)

Answer: C

2. `int a, *p = &a;`
Which of the following statement will not add 1 to a variable?

A. `a++;` C. `*p = *p + 1;`
B. `a += 1;` D. `*p++;`

Answer: D

3. Given the following declarations:
`int x; double d; int *p; double *q;`
Which of the following expression is allowed?

A. `p = &x;` C. `q = &x;`
B. `p = &d;` D. `p = x;`

Answer: A

4. Which of the following defines a pointer variable to an integer?

A. `int &ptr;` C. `int **ptr;`
B. `int *ptr;` D. `int &&ptr;`

Answer: B

5. Which of the following defines and initializes a pointer to the address of x?

A. `int *ptr = *x;` C. `int *ptr = &x;`
B. `int &ptr = *x;` D. `int *ptr = ^x;`

Answer: C

6. Pointers to pointers is a term used to describe
- A. Any two pointers that point to the same variable
B. Any two pointers that point to variables of the same type
C. Pointers used as formal parameters in a function header
D. Pointers whose contents are the address of another pointer

Answer: D

7. Given the definitions shown below, which answer is not valid?
`int i; float f; int *pd; float *pf;`

A. `pd = pf;` C. `i = 5;`
B. `pd = &i;` D. `pf = &f;`

Answer: A

8. If a is declared as integer, which of the following statement is false?

A. The expression `*&a` and `a` are the same.
B. The expression `*&a` and `&*a` are the same.
C. The expression `int *p = &a` is valid.
D. `printf("%d", *&a);` will print value of a.

Answer: B

9. Which of the following statements about pointers and arrays is true?
- The only way to reference data in array is with index operator.
 - The name of the array is a pointer variable.
 - The following expressions are identical when ary is an array: ary and &ary[0]
 - The following expressions are identical when ary is an array: *ary and &ary[0]

Answer: C

10. Which of the following is not a C memory allocation function?
- malloc()
 - realloc()
 - calloc()
 - alloc()

Answer: D

11. Which of the following statements about releasing memory allocation is false?
- To ensure that allocated memory is released, it should be freed before the program ends.
 - It is an error to dereference a pointer to allocated memory after the memory has been released.
 - Memory should be freed as soon as it is no longer needed.
 - Only one call to free is necessary to release an entire array allocated with calloc.

Answer: A

12. If ary is name of an integer array with 10 elements then which of the following statement is false?
- The two expressions *(ary + 5) and ary[5] are same
 - Name of array **ary** is a pointer constant to the first element of array.
 - The two expressions **ary** and **&ary[0]** are same.
 - If p is an integer pointer variable then **p=ary;** is invalid statement.

Answer: D

13. Which of the following statements about pointer arithmetic is not true?
- A pointer variable can not be pre-fixed or post-fixed with increment or decrement operators.
 - An integer value may be added or subtracted from a pointer variable.
 - When two pointers point to the same array, one pointer variable can be subtracted from another.
 - Comparison is valid with two pointers that point to the same array.

Answer: A

14. Given the following prototype for sum function and the definitions shown below:
- ```
int sum (int *, int *);
int i = 2; int j = 4;
int *pi = &i; int *pj = &j;
```
- Indicate which of the following function call to sum is not valid?
- j = sum (&i, &j);
  - i = sum (pi, &j);
  - i = sum (2, 10);
  - J = sum (pi, pj);

**Answer: C**

15. Which of the following pointer expression is not wrong?
- A. `y = *p1 ** p2;`
  - B. **Comparison like: `p1>p2`, `p1==p2`, and `p1! =p2`**
  - C. `sum+=*p2;`
  - D. `z =5* - *p2 / *p1;`

**Answer: D**

**Unit-2**

16. Given a structure variable named `stu` which is of type `struct STU` and contains a field named `major`, which of the following statements correctly refers to `major`?
- A. `major`
  - B. `stu-major`
  - C. `stu.major`
  - D. `STU.major`

**Answer: C**

17. Which of the following statement is not true about structure?
- A. A field in a structure can itself be a structure.
  - B. Individual members of a structure can be passed to the function.
  - C. Structure definition is just a template which is used to declare variables of type `struct`.
  - D. A structure can not have two fields with same type.

**Answer: D**

18. Given a pointer `ptr` to a structure `stu` containing a field called `name` which of the following statements correctly refer `name`?
- A. `ptr->name`
  - B. `ptr.name`
  - C. `ptr->stu.name`
  - D. `ptr->stu->name`

**Answer: A**

19. Which of the following is not a derived data type?
- A. Arrays
  - B. Float
  - C. Pointers
  - D. Structure

**Answer: B**

20. Which of the following allows a portion of memory to be shared by different types of data?
- A. Array
  - B. Structure
  - C. Union
  - D. File

**Answer:**

21. Determine which of the following statement is true?
- A. A union can have another union as one of the fields.
  - B. When accessing the fields of a structure through a pointer `p`, `(*p).field_name` and `p->field_name` both are same.
  - C. A structure can have another structure as one of the fields.
  - D. A structure can not have an array as one of its elements.

**Answer: D**

22. Determine which of the following statement is false?
- A. A structure can be initialized when it is defined.
  - B. The following two expressions are same if ptr is a pointer to structure:  
\*p.x and p->x
  - C. We can access the members of a structure using member operator.
  - D. We can access the members of a structure through a pointer using member selection operator.

**Answer: B**

23. Which of the following statement is true for structures?
- A. A structure can be copied to another structure of same type using assignment operator.
  - B. Structure definition that does not contain tag can be used in future to create variables of type structure.
  - C. If ptr is a pointer to structure student then ptr->filed\_name is used to refer a member of structure student.
  - D. Definition of structure variable does not reserve space in memory.

**Answer: A**

24. Which of the following method is not valid to send information in a structure to a function?
- A. Pass each member of the structure as an actual argument
  - B. Pass copy of entire structure
  - C. Pass structure definition
  - D. Pass address of structure

**Answer: C**

25. Which of the following can be used to create a new type that can be used anywhere a type is permitted?
- A. typedef
  - B. array
  - C. struct
  - D. Both struct and typedef

**Answer: A**

26. Which of the following statement is false about structure?
- A. Structure can be nested.
  - B. Structure can have many members of same or different data types.
  - C. All elements of array of structure are not created in continuous memory locations.
  - D. Array of structure ca be created.

**Answer: C**

27. Consider following declaration:
- ```
struct employee
{
    int eno;
    char enm[20];
    struct
    {
        int bsal;
        float bonus;
    } salary, sal;
}e1;
```
- Which of the following statement is false?

- A. Above declaration is wrong. It will give error.
- B. We can refer bonus with e1.salary.bonus as well as e1.sal.bonus.
- C. We can refer eno with e1.sal.
- D. salary and sal both can be used to refer members of inner structure.

Answer: A

28. Which of the following statement is true about union?
- A. We can initialize all members of union.
 - B. If you fill in a member of one type and then try to use a different type, you can get unpredictable results.
 - C. All union members can not be accessed at the same time.
 - D. Size of a union is equal to total of size of all members.

Answer: B

29. Structures can not be initialized at
- A. Run time by taking input from user
 - B. Run time by assignment statement
 - C. Compile time at time of declaration of structure variable
 - D. Compile time within structure definition using assignment statement

Answer: D

30. Which of the following statement is false?
- A. Each variable / data field within a structure is called a **member of the structure OR structure elements**.
 - B. Definition of structure describes a format called **template** that can be used later in a program to declare structures.
 - C. Individual structure members can be accessed like other variables because they are like variable.
 - D. Structure variables are known as an **instance**.

Answer: C

31. What are two predefined FILE pointers in C?
- A. stdout and stderr
 - B. console and error
 - C. stdout and stdio
 - D. stdio and stderr

Answer: C

32. Code:
- ```
FILE *f = fopen(fileName, "r");
readdata(f);
if(???)
{ puts("End of file was reached");
}
```

Which one of the following can replace the???? in the code above to determine if the end of a file has been reached?

- A. feof( f )
- B. f == NULL
- C. eof( f )
- D. !f

**Answer: A**

33. f = fopen( filename, "r" );  
Referring to the code above, what is the proper definition for the variable f?
- A. FILE f;
  - B. FILE \*f;
  - C. struct FILE f;
  - D. int f;

**Answer: B**

34. Which one of the following is valid for opening a file for only reading?
- |                             |                            |
|-----------------------------|----------------------------|
| A. fileOpen (filenm, "r");  | C. fopen (filenm, "r");    |
| B. fileOpen (filenm, "ra"); | D. fopen (filenm, "read"); |

**Answer: C**

35. What is a proper method of opening a file test.bin for writing as binary file?
- |                                            |
|--------------------------------------------|
| A. FILE *f = fwrite( "test.bin", "b" );    |
| B. FILE *f = fopenb( "test.bin", "w" );    |
| C. FILE *f = fopen( "test.bin", "wb" );    |
| D. FILE *f = fopen( "test.bin", "write" ); |

**Answer: C**

36. fputs function is used to
- write characters to a file
  - takes 2 parameters
  - returns a character
  - requires a file pointer
- |                  |                              |
|------------------|------------------------------|
| A. all are true  | C. only i and ii are true    |
| B. all are false | D. only i,ii and iv are true |

**Answer: D****Unit-3**

37. Files are a \_\_\_\_\_ type of Data Structure.
- |              |                  |
|--------------|------------------|
| A. Linear    | C. Non-Primitive |
| B. Primitive | D. Non-Linear    |

**Answer: C**

38. Which of the following data structures are indexed structures?
- |                  |                  |
|------------------|------------------|
| A. linear arrays | C. Both A and B  |
| B. linked lists  | D. None of these |

**Answer: A**

39. Two dimensional arrays are also called?
- |                  |                  |
|------------------|------------------|
| A. tables arrays | C. both A and B  |
| B. matrix arrays | D. None of these |

**Answer: C**

40. A variable P is called pointer if?
- |                                                                       |
|-----------------------------------------------------------------------|
| A. P contain the value of element DATA                                |
| B. P contains the address of an element DATA.                         |
| C. P contain the value of element DATA and the address of DATA        |
| D. P can store only constant memory address and it can not be changed |

**Answer: B**

41. Which of the following data structure store the homogeneous data elements??
- |            |                  |
|------------|------------------|
| A. Arrays  | C. Pointers      |
| B. Records | D. None of these |

**Answer: A**

42. The difference between linear array and a record is?
- A. An array is suitable for homogeneous data but the data items in a record may have different data type
  - B. In a record, there may not be a natural ordering in opposed to linear array.
  - C. A record form a hierarchical structure but a linear array does not.
  - D. All of above

**Answer: D**

43. When new data are to be inserted into a data structure, but there is no available space; this situation is usually called?
- A. underflow
  - B. overflow
  - C. housefull
  - D. saturated

**Answer: B**

44. The term "push" and "pop" is related to the?
- A. array
  - B. queue
  - C. stacks
  - D. All of these

**Answer: C**

45. A data structure where elements can be added or removed at either end but not in the middle?
- A. Linked lists
  - B. Stacks
  - C. Queues
  - D. Deque

**Answer: D**

46. A stack is \_\_\_\_ type of data structure
- A. Linear
  - B. Non-Linear
  - C. Both (A) and (B)
  - D. None of the Above

**Answer: A**

47. Which of the following statement is FALSE for the Stack data structure?
- A. Its nature is LIFO
  - B. Its nature is FIFO
  - C. It is a non- primitive data structure
  - D. It is a Linear data structure

**Answer: B**

48. A data structure in which insertion and deletion of an elements occurs at only one end is known as \_\_\_\_\_.
- A. Queue
  - B. Stack
  - C. Tree
  - D. Graph

**Answer: B**

49. Which of the following is an operation of a Stack data structure?
- A. Top
  - B. Bottom
  - C. Peep
  - D. None of the Above

**Answer: C**

50. Which of the following is an operation of a Stack data structure?
- A. Push
  - B. Change
  - C. Peep
  - D. All of the Above

**Answer: D**

51. An operation that is used to insert an element on a stack is known as \_\_\_\_\_.  
 A. Push C. Peep  
 B. Pop D. Change

**Answer: A**

52. An operation that is used to delete an element from a stack is known as \_\_\_\_\_.  
 A. Push C. Peep  
 B. Pop D. Change

**Answer: B**

53. An operation that is used to give the value of an element at a particular position from a top of a stack is known as \_\_\_\_\_.  
 A. Push C. Peep  
 B. Pop D. Change

**Answer: C**

54. An operation that is used to change the value of an element at a particular position from a top of a stack is known as \_\_\_\_\_.  
 A. Push C. Peep  
 B. Pop D. Change

**Answer: D**

55. Which of the following is TRUE for a Stack data structure?  
 A. Linear C. Both (A) and (B)  
 B. Non-primitive D. None of the Above

**Answer: C**

56. Which of the following is NOT an application of a Stack data structure?  
 A. Stack Machine C. Evaluation of an Expressions  
 B. Recursion D. Creates a folder

**Answer: D**

#### Unit-4

57. A linked list is \_\_\_\_\_ type of data structure?  
 A. Linear C. Both (A) and (B)  
 B. Non-Linear D. None of the Above

**Answer: A**

58. Which of the following is NOT the type of Singly linked list?  
 A. Two-way list C. Three-way list  
 B. Doubly Linked list D. Circular linked list

**Answer: C**

59. Which of the following is the type of Singly linked list?  
 A. One-way list C. Three-way list  
 B. Two-way list D. Four-way list

**Answer: B**

60. A data structure that contains not only a data field but also contains pointer field is known as \_\_\_\_\_.  
 A. Queue C. Tree  
 B. Stack D. Linked List



**Answer: D**

61. A linked list in which last node pointing to the first node is known as \_\_\_\_\_.  
A. Singly linked list  
B. Doubly linked list  
C. Circular linked list  
D. None of the above

**Answer: C**

62. A storage representation of a linked list in a memory is \_\_\_\_\_.  
A. Linear  
B. Non-linear  
C. Both (A) and (B)  
D. Either (A) or (B)

**Answer: B**

63. Which of the following statement is FALSE for the Queue data structure?  
A. Its nature is LIFO  
B. Its nature is FIFO  
C. It is a non-primitive data structure  
D. It is a Linear data structure

**Answer: A**

64. A data structure in which insertion of an element occurs at one end and deletion of an element occurs at other end is known as \_\_\_\_\_.  
A. Tree  
B. Graph  
C. Queue  
D. Stack

**Answer: C**

65. A data structure in which insertion and deletion of an elements occurs at both the end is known as \_\_\_\_\_.  
A. Stack  
B. Queue  
C. Priority Queue  
D. Deque

**Answer: D**

66. Which of the following is TRUE for a Queue data structure?  
A. Linear  
B. Non-primitive  
C. Both (A) and (B)  
D. None of the Above

**Answer: C**

**Short Questions****Unit-1**

1. Define: indirection operator, pointer variable
2. Give the concept of pointers to array.
3. Differentiate between '\*' and '&' operators in pointers.
4. main()

```

{
 int x = 7;
 int *y = &x;
 int *z = &y;
 *y = 51;
 (*y)++;
 (*z)--;
}

```

What will be the value of y and z, if any, for the above program segment?

5. Explain realloc function.
6. Differentiate malloc() and calloc().
7. Explain how compile time and runtime memory allocation process differ?
8. Differentiate: a pointer and a pointer variable
9. List out benefits of pointers.
10. List different pointer declaration style. Which one is preferable?
11. What is scale factor? Explain with example in brief.
12. List out operations that can be performed on pointers.

**Unit-2**

13. Define: structure, member operator
14. Differentiate: structure and union
15. Can entire structure variable be assigned to another structure variable, provided both variables having same structure? Answer with example.
16. Can entire structure variable be passed to a function as an argument? Answer with example.
17. Define a structure called college which contains an integer quantity called college\_code and a string called c\_name with length 20. Create structure called courses within structure college which contains integer quantity called course\_no and a float quantity called class\_result. Create one variable called college1 along with definition.
18. Define a union called "item" consisting of an integer called code, character string called nm and float value called price. Declare union variable called product along with definition.
19. Explain typedef in brief with suitable example.
20. Differentiate '.' and '->' operators.
21. Consider ptr is a pointer pointing to struct stud and rollno is a member of struct stud. Mention two methods of accessing member rollno of struct stud through pointer ptr.
22. Consider ptr is a pointer pointing to struct stud and rollno is a member of struct stud. We can access member rollno of struct stud through pointer ptr using **(\*ptr).rollno**  
Why we need to use parenthesis around \*ptr?
23. What we mean by structure definition? What is the importance of tag in structure definition?
24. List file modes available to manage the file in C.
25. Explain the Append mode with example.
26. Write down the syntax to open the text file both in read and write mode.
27. Explain the fclose() function with example.
28. What do you mean by file pointer?
29. Which functions are used to read the character from the file and write character to the file?
30. Explain function(s) used to read the integer number from the file.
31. Differentiate: append mode and right mode

32. Differentiate: printf and fprintf
33. Differentiate: scanf and fscanf
34. Differentiate: getc and getchar

**Unit-3**

35. Define: Data Structure, Stack
36. Define and explain in brief: Primitive Data Structure
37. What is non-primitive Data Structure?
38. Give the Example of Primitive Data Structure.
39. Give the Example of Non-Primitive Data Structure.
40. What do you mean Linear Data Structure?
41. What do you mean by Non-Linear Data Structure?
42. Draw the Hierarchical Structure of Data Structure.
43. List out different applications of data Structure.
44. Which are the main operations that can be performed on Data Structure?
45. Differentiate: primitive and non primitive data structure
46. What is a Stack? Give examples.
47. State various Applications of Stack.
48. List operations on a Stack. Define ANY ONE of them.
49. Give representation of a Stack data structure.
50. What do you mean by Top and Bottom of a Stack?

**Unit-4**

51. What is a Linked List? How is it represented?
52. State various Applications of Linked List.
53. Describes the different types of Linked List.
54. What is a Singly Linked list?
55. What is a Circular Linked list?
56. What is a doubly Linked list?
57. Differentiate between singly linked list and doubly linked list.
58. What is a Queue? Give examples.
59. State various types of queue.
60. Define: Queue and Deque.
61. Give representation of a Queue data structure.
62. Define : Circular Queue and Priority Queue.
63. Differentiate between stack and queue data structure.

**Long Questions****Unit-1**

1. Define pointer variable. How can we declare and initialize pointer variable? How can we access value of variable through pointer type variable?
2. Define: Pointer. How is it declared? Also explain how arithmetic operations can be performed on the pointer variable by taking example.
3. What are pointers? How can they be used with arrays? Explain pointer to an array using appropriate examples.
4. Explain the importance of pointers in functions by taking suitable example. How pointers can be used to return multiple values to functions?
5. Explain pointer to structure using suitable example.
6. Write a note on Dynamic memory allocation.
7. Explain pointer arithmetic with example.
8. Write note on: pointer to pointer

**Unit-2**

9. What is structure? Explain its definition, declaration and assigning values to members of structure. Also mention advantages of structure over other data types.
10. Explain array of structures using suitable examples.
11. What is union? Explain its definition, declaration and assigning values to members of union.
12. What is union? Explain its storage representation. How a member of union is assigned an initial value? Explain in brief with example.
13. Write note on: structure within structure
14. Explain array within structure using suitable example.
15. Write note on: pointer to structure
16. Explain pointer to structure array using appropriate example.
17. Describe the usage and limitation of function getc and putc.
18. Explain fprintf and fscanf function with example.
19. Explain the all the modes of file management with example.
20. Explain the getw and putw function with example.

**Unit-3**

21. Explain the data structure with c briefly.
22. Write down advantages of data structure.
23. Write a short note on primitive data structure operations.
24. Write a short note on linear data structure.
25. Write a short note on non linear data structure.
26. Explain the linear and non linear data structure briefly.
27. Write an algorithm to insert an element into a Stack.
28. Write an algorithm to delete an element from a Stack.
29. Write an algorithm for Peep operation of a Stack.
30. Write an algorithm for Change operation of a Stack.
31. Explain a STACK with an example. Write along with for various operations performed over a stack.

**Unit-4**

32. Write an algorithm to insert an element at the beginning of a Singly linked list.
33. Write an algorithm to delete an element from a Singly linked list.
34. Write an algorithm to insert an element at the ending of a Singly linked list.
35. Write an algorithm to insert an element into a Singly linked list that maintains ascending order of elements.
36. Write a short note on Singly linked List.
37. Write an algorithm to insert an element into a simple queue.

38. Write an algorithm to delete an element from a simple queue.
39. Explain a Queue with an example. Write along with for various operations performed over a queue.

US02CBCA01