



AKSHEYAA COLLEGE OF ENGINEERING

Puludivakkam, (Near Nelvay Junction), Maduranthagam Taluk,
Kancheepuram 603 314, Tamil Nadu, India.
Phone: + 91 44 27568004-02-05, email: principal@ace.ac.in
(An ISO 9001:2008 Certified Institution)



DEPARTMENT OF SCIENCE AND HUMANITIES

Sem/Year/Branch: II/I/CSE

CS6202-PROGRAMMING AND DATA STRUCTURES I QUESTION BANK

UNIT I

PART A

1. Give two examples of C preprocessors with syntax.(APR/MAY 2015)
2. What are function pointers in C? Explain with example. (APR/MAY 2015)
3. Define array. Give an example.(Nov/Dec 2014)
4. Give example on call by reference.(Nov/Dec 2104)
5. With the help of the printf function show how C handles functions with variable number of arguments.(May/June 2014)
6. Define macros with an example.(May/June 2014)
7. What is Size of operator in C?
8. Distinguish between Call by value Call by reference.
9. What is the difference between an array and pointer?
10. Define function. Why they are needed? What are its types?
11. What is a Pointer and how it is declared? What are the uses of Pointers?
12. Define pre-processor in C.
13. Define recursive function.
14. Differentiate between pointer to constants and constants pointers
15. Define global declaration.

PART A

1. Explain the various control statements in C language with example in detail. (16) (Nov/Dec 2014)
2. Briefly discuss about :(16) (Nov/Dec 2014)
 - i) Function with number of arguments
 - ii) Function pointers
3. Explain the C preprocessor operators, each with a neat example that is used to create macros.(8) (Apr/May 2015)
4. Explain in detail about various Pre-processor directives with suitable examples.(16)
5. Explain about passing parameters to the function with and without using pointers with suitable example.(or) Explain about Call by Value and Call by Reference with an example program.(16)
6. Write a function that returns a pointer to the maximum value of an array of double's. If the array is empty, return NULL. (8) (APR/MAY 2015)
7. Write a C program to find all the roots of a quadratic equation.(8) (APR/MAY 2015)
8. Write a C program using function to check if the given input number is palindrome or not.(8) (APR/MAY 2015)
9. Write a C program to print the Fibonacci numbers. (May/June 2014)

Prepared By M.GIRUBA NANDHINI, AP/CSE

10. Write a C program to multiply two matrices that are represented as pointers. Use function pointers to the function multiply which takes the two matrices as parameter and prints the result of the multiplication. (May/June 2014)

UNIT II

PART A

1. Give applications in which unions rather than structures can be used. (May/June 2014)
2. Will the following declarations work justify your answer? (May/June 2014)
3. What are the statements used for reading a file? (Nov/Dec 2014)
4. Define the need for union in C. (Nov/Dec 2014)
5. What is the difference between `getc()` and `getchar()` ? Explain (Apr/May 2015)
6. Explain the syntax as given below: (Apr/May 2015)
7. Compare arrays and structures.
8. Compare structures and unions.
9. Define Structure in C.
10. Define file.
11. Define binary files.
12. List out the file handling functions.
13. Define opening a file.
14. What is meant by Union in C.?
15. What are the files attributes in C?

PART B

1. Write a C program to read the contents of a file "in.txt" from last to first and write the contents to "out.txt"(8) (May/June 2014)
2. Write the function prototype and explain how files are manipulated in C.(8) (May/June 2014)
3. Create a structure to store a complex number and write functions (for addition) that handle this new structure. (8) (May/June 2014)
4. Write program to perform the following operations for the customers of a bank using the concept of structures. (8) (May/June 2014)
5. Explain the difference between structure and Union with examples. (16) (Nov/Dec 2014) or State the advantages and disadvantages of structures and unions in C programming.(4) (Apr/May 2015)
6. Explain about file manipulation in detail with suitable program.(16) (Nov/Dec 2014)
7. Write a C program that uses functions to perform the following operations using structure :(12) (Apr/May 2015)
 - Reading a complex number
 - Writing a complex number
 - Addition of two complex numbers
 - Multiplication of two complex numbers
8. Using C programming, display the contents of a file on screen.(4) (Apr/May 2015)
9. Explain structure in detail with a neat example(16)
10. Explain union in detail with suitable example.(16)

UNIT III

PART A

1. What is advantage of an ADT? (May/June 2014)
2. What are abstract data types? (Nov/Dec 2014)
3. What is a circular linked list? (Nov/Dec 2014)
4. Define ADT. (Apr/May 2015)
5. What is a static linked list? State any two applications of it. (Apr/May 2015)
6. Define ADT. What are abstract data types? (Nov/Dec 2014)(Apr/May 2015)
7. Define non-linear data structure.
8. What is meant by a linked list?
9. What is a singly listed list?
10. Define double circularly linked list?
11. Define polynomial ADT.

PART B

1. Write an algorithm to perform insertion and deletion on a doubly linked list.(16) (May/June 2014)
2. Consider an array A [1: n]. Given a position, write an algorithm to insert an element in the array. If the position is empty, the element is inserted easily. If the position is already occupied the element should be inserted with the minimum number of shifts. (Note: The elements can shift to the left to the right to make the minimum number of moves).(16) (May/June 2014)
3. Describe the creation of a doubly linked list and appending the list. Give relevant coding in C.(16) (Nov/Dec 2014)
4. Explain the following.(i)Application of lists. (ii)polynomial manipulation.(16) (Nov/Dec 2014)
5. Write a C program to perform addition, subtraction and multiplication operations on polynomial using linked list.(16) (Apr/May 2015)
6. Write a C code for circular link list with create, insert, delete, display operations using structure pointer.(16) (Apr/May 2015)

UNIT IV

PART A

1. Define double ended queue. (May/June 2014)
2. List the application of a queue. (May/June 2014)
3. Give the application of stack. (Nov/Dec 2014)
4. What is doubly ended queue? (Nov/Dec 2014)
5. Write the syntax of Calloc () and Realloc () and mention its application in linked list. (Apr/May 2015)
6. Given the prefix for an expression, write its postfix.-*+abc/ef-g/hi (Apr/May 2015)
7. What are the two operations of stack?
8. What is circular queue?
9. What is priority queue?
10. State the difference between stacks and linked lists.

PART B

1. Write an algorithm to convert an infix expression to a postfix expression. Trace the algorithm to convert an infix expression $*(a+b)*c/d+e/f*$ to a postfix expression. Explain the need for infix and postfix expressions.(16) (May/June 2014)
2. Write an algorithm to perform the four operations in a double ended queue that is implemented as an array.(16) (May/June 2014)
3. Discuss about stack ADT in detail. Explain any one application of stack.(16) (Nov/Dec 2014)
4. Explain about queue ADT in detail. Explain any one application of queue with suitable example.(16) (Nov/Dec 2014)
5. Write a C program that checks if expression is correctly parenthesized using stack.(12) (Apr/May 2015)
6. Write the function to check for stack status as full() or empty(). (4) (Apr/May 2015)
7. Explain stack using linked list implementation.(16)
8. Explain stack using array implementation.(16)
9. Explain evaluation of postfix expression in detail.(16)

UNIT V

PART A

1. What is the time complexity of binary search? (May/June 2014)
2. List the sorting algorithm which uses logarithmic time complexity. (May/June 2014)
3. Define extendible hashing. (Nov/Dec 2014)
4. Differentiate internal and external sorting. (Nov/Dec 2014)
5. What is meant by internal and external sorting? Give four examples of each type.
6. State the applications of linear and binary search techniques.
7. Define quick sort.
8. Define insertion sort.
9. Define merge sort.
10. Define bubble sort.
11. Differentiate between merge sort and quick sort?
12. Define hashing.
13. Define hash table.
14. Define rehashing.
15. List out the different types of hashing functions.

PART B

1. Write short notes on hashing and the various collision resolution techniques.(16) (May/June 2014)
2. Write an algorithm to sort 'n' numbers using quick sort. Show how the following numbers are sorted using quicksort: 45, 28, 90, 1, 46, and 39. (16) (May/June 2014)
3. What are the different types of hashing techniques? Explain them in detail with example.(16) (Nov/Dec 2014)
4. Write an algorithm to sort a set of 'N' numbers using quick sort. Trace the algorithm for the following set of numbers:88, 11, 22, 44, 66, 99, 32, 67, 54, 10(Nov/Dec 2014)
5. Sort the given integers and show the intermediate results using shell sort
35, 12, 14, 9, 15, 45, 32, 95, 40, 5 (8m) (Apr/May 2015)
6. Write C code to sort an integer array using shell sort.(8) (Apr/May 2015)
7. Write a C code to perform binary search. (10m) (Apr/May 2015)
8. Explain the Rehashing techniques. (6) (Apr/May 2015)

9. Explain insertion sort in detail.(16)
10. Write a C code to perform linear search.(16)