*Here, we are not only providing the important C programs. We are also giving you the list of important questions on that particular topic. Hope you will prepare them very well to crack the interviews successfully!*

**Technical Aptitude Cse – It**

LIST OF IMPORTANT C QUESTIONS :

* + **Linked Lists**
  + How do you reverse a singly linked list? How do you reverse a doubly linked list?
  + Write a C program to do the same as above.
  + Given only a pointer to a node to be deleted in a singly linked list, how do you delete it?
  + How do you sort a linked list? Write a C program to sort a linked list.
  + How to declare a structure of a linked list?
  + Write a C program to implement a Generic Linked List.
  + How do you reverse a linked list without using any C pointers?
  + How would you detect a loop in a linked list?
  + Write a C program to detect a loop in a linked list.
  + How do you find the middle of a linked list?
  + Write a C program to return the middle of a linked list
  + If you are using C language to implement the heterogeneous linked list, what pointer type will you use?
  + How to compare two linked lists? Write a C program to compare two linked lists.
  + How to create a copy of a linked list? Write a C program to create a copy of a linked list.
  + Write a C program to free the nodes of a linked list
  + Can we do a Binary search on a linked list?
  + Write a C program to return the nth node from the end of a linked list.
  + How would you find out if one of the pointers in a linked list is corrupted or not?
  + Write a C program to insert nodes into a linked list in a sorted fashion
  + Write a C program to remove duplicates from a sorted linked list
  + How to read a singly linked list backwards?
  + How can I search for data in a linked list?

* + **Write your own ....** Write your own C program to implement the atoi() function
  + Implement the memmove() function.
  + What is the difference between the memmove() and memcpy() function?
  + Write C code to implement the strstr() (search for a substring) function.
  + Write your own printf() function in C
  + Implement the strcpy() function.
  + Implement the strcmp(str1, str2) function.
  + Implement the substr() function in C.
  + Write your own copy() function
  + Write C programs to implement the toupper() and the isupper() functions
  + Write a C program to implement your own strdup() function.
  + Write a C program to implement the strlen() function
  + Write your own strcat() function
  + ***Programs***
  + Write a C program to swap two variables without using a temporary variable
  + What is the 8 queens problem? Write a C program to solve it.
  + Write a C program to print a square matrix helically.
  + Write a C program to reverse a string
  + Write a C program to reverse the words in a sentence in place.
  + Write a C program generate permutations.
  + Write a C program for calculating the factorial of a number
  + Write a C program to calculate pow(x,n)?
  + Write a C program which does wildcard pattern matching algorithm
  + How do you calculate the maximum subarray of a list of numbers?
  + How to generate fibonacci numbers? How to find out if a given number is a fibonacci number or not? Write C programs to do both.
  + Solve the Rat In A Maze problem using backtracking.
  + What Little-Endian and Big-Endian? How can I determine whether a machine's byte order is big-endian or little endian? How can we convert from one to another?
  + Write C code to solve the Tower of Hanoi problem.
  + Write C code to return a string from a function
  + Write a C program which produces its own source code as its output
  + Write a C progam to convert from decimal to any base (binary, hex, oct etc...)
  + Write C code to check if an integer is a power of 2 or not in a single line?
  + Write a C program to find the GCD of two numbers.
  + Finding a duplicated integer problem
  + Write code to remove duplicates in a sorted array.
  + Find the maximum of three integers using the ternary operator.
  + How do you initialize a pointer inside a function?
  + Write C code to dynamically allocate one, two and three dimensional arrays (using malloc())
  + How would you find the size of structure without using sizeof()?
  + Write a C program to multiply two matrices.
  + Write a C program to check for palindromes.
  + Write a C program to convert a decimal number into a binary number.
  + Write C code to implement the Binary Search algorithm.
  + Wite code to evaluate a polynomial.
  + Write code to add two polynomials
  + Write a program to add two long positive numbers (each represented by linked lists).
  + How do you compare floating point numbers?
  + What's a good way to implement complex numbers in C?
  + How can I display a percentage-done indication on the screen?
  + Write a program to check if a given year is a leap year or not?
  + Is there something we can do in C but not in C++?
  + How to swap the two nibbles in a byte ?
  + How to scan a string till we hit a new line using scanf()?
  + Write pseudocode to compare versions (like 115.10.1 vs 115.11.5).
  + How do you get the line numbers in C?
  + How to fast multiply a number by 7?
  + Write a simple piece of code to split a string at equal intervals
  + Is there a way to multiply matrices in lesser than o(n^3) time complexity?
  + How do you find out if a machine is 32 bit or 64 bit?
  + Write a program to have the output go two places at once (to the screen and to a file also)
  + Write code to round numbers
  + How can we sum the digits of a given number in single statement?
  + Given two strings A and B, how would you find out if the characters in B were a subset of the characters in A?
  + Write a program to merge two arrays in sorted order, so that if an integer is in both the arrays, it gets added into the final array only once. \*
  + Write a program to check if the stack grows up or down
  + How to add two numbers without using the plus operator?
  + How to generate prime numbers? How to generate the next prime after a given prime?
  + Write a program to print numbers from 1 to 100 without using loops!
  + Write your own trim() or squeeze() function to remove the spaces from a string.
  + Write your own random number generator function in C.\*
  + Write your own sqrt() function in C\*
  + **Trees**
  + Write a C program to find the depth or height of a tree.
  + Write a C program to determine the number of elements (or size) in a tree.
  + Write a C program to delete a tree (i.e, free up its nodes)
  + Write C code to determine if two trees are identical
  + Write a C program to find the mininum value in a binary search tree.
  + Write a C program to compute the maximum depth in a tree?
  + Write a C program to create a mirror copy of a tree (left nodes become right and right nodes become left)!
  + Write C code to return a pointer to the nth node of an inorder traversal of a BST.
  + Write C code to implement the preorder(), inorder() and postorder() traversals. Whats their time complexities?
  + Write a C program to create a copy of a tree
  + Write C code to check if a given binary tree is a binary search tree or not?
  + Write C code to implement level order traversal of a tree.
  + Write a C program to delete a node from a Binary Search Tree?
  + Write C code to search for a value in a binary search tree (BST).
  + Write C code to count the number of leaves in a tree
  + Write C code for iterative preorder, inorder and postorder tree traversals
  + Can you construct a tree using postorder and preorder traversal?
  + Construct a tree given its inorder and preorder traversal strings. Similarly construct a tree given its inorder and post order traversal strings.
  + Find the closest ancestor of two nodes in a tree.
  + Given an expression tree, evaluate the expression and obtain a paranthesized form of the expression.
  + How do you convert a tree into an array?
  + What is an AVL tree?
  + How many different trees can be constructed using n nodes?
  + A full N-ary tree has M non-leaf nodes, how many leaf nodes does it have?
  + Implement Breadth First Search (BFS) and Depth First Search (DFS)
  + Write pseudocode to add a new node to a Binary Search Tree (BST)
  + What is a threaded binary tree?
  + **Bit Fiddling**
  + Write a C program to count bits set in an integer?
  + What purpose do the bitwise and, or, xor and the shift operators serve?
  + How to reverse the bits in an interger?
  + Check if the 20th bit of a 32 bit integer is on or off?
  + How to reverse the odd bits of an integer?
  + How would you count the number of bits set in a floating point number?\*
  + **Sorting Techniques**
  + What is heap sort?
  + What is the difference between Merge Sort and Quick sort?
  + Give pseudocode for the mergesort algorithm
  + Implement the bubble sort algorithm. How can it be improved? Write the code for selection sort, quick sort, insertion sort.
  + How can I sort things that are too large to bring into memory?
  + **C Pointers**
  + What does \*p++ do? Does it increment p or the value pointed by p?
  + What is a NULL pointer? How is it different from an unitialized pointer? How is a NULL pointer defined?
  + What is a null pointer assignment error?
  + Does an array always get converted to a pointer? What is the difference between arr and &arr? How does one declare a pointer to an entire array?
  + Is the cast to malloc() required at all?
  + What does malloc() , calloc(), realloc(), free() do? What are the common problems with malloc()? Is there a way to find out how much memory a pointer was allocated?
  + What's the difference between const char \*p, char \* const p and const char \* const p?
  + What is a void pointer? Why can't we perform arithmetic on a void \* pointer?
  + What do Segmentation fault, access violation, core dump and Bus error mean?
  + What is the difference between an array of pointers and a pointer to an array?
  + What is a memory leak?
  + What are brk() and sbrk() used for? How are they different from malloc()?
  + What is a dangling pointer? What are reference counters with respect to pointers?
  + What do pointers contain?
  + Is \*(\*(p+i)+j) is equivalent to p[i][j]? Is num[i] == i[num] == \*(num + i) == \*(i + num)?
  + What operations are valid on pointers? When does one get the Illegal use of pointer in function error?
  + What are near, far and huge pointers?
  + What is the difference between malloc() and calloc()?
  + Why is sizeof() an operator and not a function?
  + What is an opaque pointer?
  + What are the common causes of pointer bugs?
  + **C Functions**
  + How to declare a pointer to a function?
  + Does extern in a function declaration mean anything?
  + How can I return multiple values from a function?
  + Does C support function overloading?
  + What is the purpose of a function prototype?
  + What are inline functions?
  + How to declare an array of N pointers to functions returning pointers to functions returning pointers to characters?
  + Can we declare a function that can return a pointer to a function of the same type?
  + How can I write a function that takes a variable number of arguments? What are the limitations with this? What is vprintf()?
  + With respect to function parameter passing, what is the difference between call-by-value and call-by-reference? Which method does C use?
  + If I have the name of a function in the form of a string, how can I invoke that function?
  + What does the error, invalid redeclaration of a function mean?
  + How can I pass the variable argument list passed to one function to another function.
  + How do I pass a variable number of function pointers to a variable argument (va\_arg) function?
  + Will C allow passing more or less arguments than required to a function.
  + **C Statements**
  + Whats short-circuiting in C expressions?
  + Whats wrong with the expression a[i]=i++; ? Whats a sequence point?
  + Does the ?: (ternary operator) return a lvalue? How can I assign a value to the output of the ternary operator?
  + Is 5[array] the same as array[5]?
  + What are #pragmas?
  + What is the difference between if(0 == x) and if(x == 0)?
  + Should we use goto or not?
  + Is ++i really faster than i = i + 1?
  + What do lvalue and rvalue mean?
  + What does the term cast refer to? Why is it used?
  + What is the difference between a statement and a block?
  + Can comments be nested in C?
  + What is type checking?
  + Why can't you nest structure definitions?
  + What is a forward reference?
  + What is the difference between the & and && operators and the | and || operators?
  + Is C case sensitive (ie: does C differentiate between upper and lower case letters)?
  + Can goto be used to jump across functions?
  + Whats wrong with #define myptr int \*?
  + What purpose do #if, #else, #elif, #endif, #ifdef, #ifndef serve?
  + Can we use variables inside a switch statement? Can we use floating point numbers? Can we use expressions?
  + What is more efficient? A switch() or an if() else()?
  + What is the difference between a deep copy and a shallow copy?
  + What is operator precedence?\*
  + **C Arrays**
  + How to write functions which accept two-dimensional arrays when the width is not known before hand?
  + Is char a[3] = "abc"; legal? What does it mean?
  + If a is an array, is a++ valid?
  + How can we find out the length of an array dynamically in C?
  + **C Variables**
  + What is the difference between the declaration and the definition of a variable?
  + Do Global variables start out as zero?
  + Does C have boolean variable type?
  + Where may variables be defined in C?
  + To what does the term storage class refer? What are auto, static, extern, volatile, const classes?
  + What does the typedef keyword do?
  + What is the difference between constants defined through #define and the constant keyword?
  + What are Trigraph characters?
  + How are floating point numbers stored? Whats the IEEE format?
  + When should the register modifier be used?
  + When should a type cast be used?
  + **C Structures**
  + Can structures be assigned to variables and passed to and from functions?
  + Can we directly compare two structures using the == operator?
  + Can we pass constant values to functions which accept structure arguments?
  + How does one use fread() and fwrite()? Can we read/write structures to/from files?
  + Why do structures get padded? Why does sizeof() return a larger size?
  + Can we determine the offset of a field within a structure and directly access that element?
  + What are bit fields in structures?
  + What is a union? Where does one use unions? What are the limitations of unions?
  + **C Macros**
  + How should we write a multi-statement macro?
  + How can I write a macro which takes a variable number of arguments?
  + What is the token pasting operator and stringizing operator in C?
  + Define a macro called SQR which squares a number.
  + **C Headers**
  + What should go in header files? How to prevent a header file being included twice? Whats wrong with including more headers?
  + Is there a limit on the number of characters in the name of a header file?
  + Is it acceptable to declare/define a variable in a C header?
  + **C File operations**
  + How do stat(), fstat(), vstat() work? How to check whether a file exists?
  + How can I insert or delete a line (or record) in the middle of a file?
  + How can I recover the file name using its file descriptor?
  + How can I delete a file? How do I copy files? How can I read a directory in a C
  + program?
  + Whats the use of fopen(), fclose(), fprintf(), getc(), putc(), getw(), putw(), fscanf(), feof(), ftell(), fseek(), rewind(), fread(), fwrite(), fgets(), fputs(), freopen(), fflush(), ungetc()?
  + How to check if a file is a binary file or an ascii file?
  + **C Declarations and Definitions**
  + What is the difference between char \*a and char a[]?
  + How can I declare an array with only one element and still access elements beyond the first element (in a valid fashion)?
  + What is the difference between enumeration variables and the preprocessor #defines?
  + **C Functions - built-in**
  + Whats the difference between gets() and fgets()? Whats the correct way to use fgets() when reading a file?
  + How can I have a variable field width with printf?
  + How can I specify a variable width in a scanf() format string?
  + How can I convert numbers to strings (the opposite of atoi)?
  + Why should one use strncpy() and not strcpy()? What are the problems with strncpy()?
  + How does the function strtok() work?
  + Why do we get the floating point formats not linked error?
  + Why do some people put void cast before each call to printf()?
  + What is assert() and when would I use it?
  + What do memcpy(), memchr(), memcmp(), memset(), strdup(), strncat(), strcmp(), strncmp(), strcpy(), strncpy(), strlen(), strchr(), strchr(), strpbrk(), strspn(), strcspn(), strtok() do?
  + What does alloca() do?
  + Can you compare two strings like string1==string2? Why do we need strcmp()?
  + What does printf() return?
  + What do setjmp() and longjump() functions do?
  + **C Functions - The main function**
  + Whats the prototype of main()? Can main() return a structure?
  + Is exit(status) equivalent to returning the same status from main()?
  + Can main() be called recursively?
  + How to print the arguments recieved by main()?

Examples of C programs :

- c program for prime number

#include <stdio.h>

main()

{

int n, c;

printf("Enter a number**\n**");

scanf("%d", &n);

if ( n == 2 )

printf("Prime number.**\n**");

else

{

for ( c = 2 ; c <= n - 1 ; c++ )

{

if ( n % c == 0 )

**break**;

}

if ( c != n )

printf("Not prime.**\n**");

else

printf("Prime number.**\n**");

}

return 0;

}

- command line arguments

#include <stdio.h>

main(int argc, char \*argv[])

{

int c;

printf("Number of command line arguments passed: %d**\n**", argc);

for ( c = 0 ; c < argc ; c++)

printf("%d. Command line argument passed is %s**\n**", c+1, argv[c]);

return 0;

}

Above c program prints the number and all arguments which are passed to it.

- Array program

#include <stdio.h>

main()

{

int array[100], n, c;

printf("Enter the number of elements in array**\n**");

scanf("%d", &n);

printf("Enter %d elements**\n**", n);

for ( c = 0 ; c < n ; c++ )

scanf("%d", &array[c]);

printf("Array elements entered by you are:**\n**");

for ( c = 0 ; c < n ; c++ )

printf("array[%d] = %d**\n**", c, array[c]);

return 0;

}

- function program

#include <stdio.h>

void my\_function();

main()

{

printf("Main function.**\n**");

my\_function();

printf("Back in function main.**\n**");

return 0;

}

void my\_function()

{

printf("Welcome to my function. Feel at home.**\n**");

}

9 - Using comments in a program

#include <stdio.h>

main()

{

*// Single line comment in c source code*

printf("Writing comments is very useful.**\n**");

*/\**

*\* Multi line comment syntax*

*\* Comments help us to understand code later easily.*

*\* Will you write comments while developing programs ?*

*\*/*

printf("Good luck c programmer.**\n**");

return 0;

}

- using structures in c programming

#include <stdio.h>

struct programming

{

float constant;

char \*pointer;

};

main()

{

struct programming variable;

char string[] = "Programming in Software Development.";

variable.constant = 1.23;

variable.pointer = string;

printf("%f**\n**", variable.constant);

printf("%s**\n**", variable.pointer);

return 0;

}

- c program for Fibonacci series

#include <stdio.h>

main()

{

int n, first = 0, second = 1, next, c;

printf("Enter the number of terms**\n**");

scanf("%d",&n);

printf("First %d terms of Fibonacci series are :-**\n**",n);

for ( c = 0 ; c < n ; c++ )

{

if ( c <= 1 )

next = c;

else

{

next = first + second;

first = second;

second = next;

}

printf("%d**\n**",next);

}

return 0;

}

- c graphics programming

#include <graphics.h>

#include <conio.h>

main()

{

int gd = DETECT, gm;

initgraph(&gd, &gm,"C:**\\**TC**\\**BGI");

outtextxy(10,20, "Graphics source code example.");

circle(200, 200, 50);

setcolor(BLUE);

line(350, 250, 450, 50);

getch();

closegraph( );

return 0;

}

**For GCC users**

If you are using GCC on Linux operating system then you need to modify programs. For example consider the following program which prints first ten natural numbers

#include <stdio.h>

#include <conio.h>

int main()

{

int c;

for ( c = 1 ; c <= 10 ; c++ )

printf("%d**\n**", c);

getch();

return 0;

}

Above source code includes a header file <conio.h> and uses function getch, but this file is Borland specific so it works in turbo c compiler but not in GCC. So the code for GCC should be like

#include <stdio.h>

int main()

{

int c;

*/\* for loop \*/*

for ( c = 1 ; c <= 10 ; c++ )

printf("%d**\n**", c);

return 0;

}

More Solution is Attached in a separate file “LinkedListProblems.pdf”

Do prepare all sorts of programs that are mentioned here very well.

Always do practice as many of questions as you can.

Prepare basics very well then move to harder sections.

Always do prepare Linked List, Stack, Queue, Tree and Sorting Algorithms very well.

AVL trees and DFS, BFS are asked during interviews. Be ready to show them with examples.

You don’t need to write the full programs possibly so do prepare the list of functions very well. Hope all our hard works for you people is worthy at the end of the day.

Do Like Our Page **Technical Aptitude Cse – It .**

**Facebook Profile Link:** <https://www.facebook.com/bollywooda2z>

**Website**: <http://technicalinterviewa2z.blogspot.in/>