Flat 10% & upto 50% off + 10% Cashback + Free additional Courses. Hurry up Learn More

Top C & Data Structure Interview Questions And Answers

Here are top 18 objective type sample C & Data Structure Interview questions and their answers are given just below to them. These sample questions are framed by experts from Intellipaat who trains for Online C Classes Training (https://intellipaat.com/c-data-structures-training/) to give you an idea of type of questions which may be asked in interview. We have taken full care to give correct answers for all the questions. Do comment your thoughts Happy Job Hunting!

Want to Learn C & Data Structure? Click Here (https://intellipaat.com/c-data-structures-training/#certification)

Top Answers to C & Data Structure Interview Questions

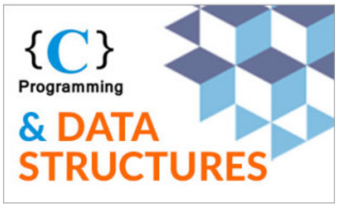
1. What are different techniques for making hash function?

Techniques for making hash function.

- Truncation Method

Search For

RECOMMENDED COURSES



(https://intellipaat.com/c-data-structures-training/)

Self

Paced

L
(HTTPS://
DAT

Lifetime Access and 24/7 Support

Browse all courses (https://intellipaat courses/big-data/)

• **Truncation Method**
This is the simplest method for computing address from a key.In this method we take only a part of the key as address.
• **Midsquare Method**
In this method the key is squared and some digits from the middle of this square are taken as address.
• **Folding Method**
In this technique, the key is divided into different part where the length of each part is same as that of the required address, except possibly the last part.
• **Division Method (Modulo-Division)**
In Modulo-Division method the key is divided by the table size and the remainder is taken as the address of the hash table.
–>Let the table size is n then
H (k) =k mod n

Master C and Data Structure, in this C and Data Structure (https://intellipaat.com/c-data-structures-training/#certification) certification training.

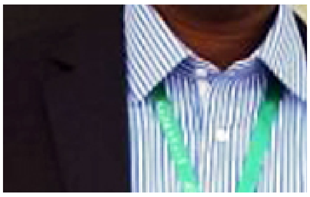
2. What are the issues that hamper the efficiency in sorting a file?

The issues are:

- Length of time required by the programmer in coding a particular sorting program.
- Amount of machine time necessary for running the particular program.
- The amount of space necessary for the particular program.

3. What is the use of volatile keyword?

The modifier 'volatile' tells the compiler that a variable's value may be changed in ways not explicitly specified by the program. For example, a global variable's address may be passed to the operating system's clock routine and used to hold the system time.



Paschal Ositadima

Career-oriented Training.Thanks for r this platform availab to bring more peopl Nigeria to embrace i as the bridge to fillir gaps in their career

INTERVIEW QUESTIONS

Top Python Interview Questions And Answers (https://intellipaat.com/ir question/python-interview-questions/)

Top Java Interview Questions And Answers (https://intellipaat.com/ir question/java-interview-questions/)

Top Linux Interview Questions And Answers (https://intellipaat.com/ir question/linux-interview-questions/)

Top SOA Interview Questions And

In this situation, the contents of the variable are altered without any explicit assignment statements in the program.

This is important because most C compilers automatically optimize certain expressions by assuming that a variable's content is unchanging if it does not occur on the left side of an assignment statement. Thus, it may not be reexamined each time it is referenced. Also, some compilers change the order of evaluation of an expression during the compilation process. The volatile modifier prevents these changes.

Learn C & Data Structure in 16 hrs from experts

E-Mail ID Phone GET STARTED

4. Write a C program without using semicolon to print 'Hello world'

```
void main(){
    if(printf("Hello world")){
    }
}
```

Go through this C Programming tutorial (https://intellipaat.com/tutorial/c-tutorial/some-programs-c/)**, to check more programmes in C.**

5. What are differences between sizeof operator and strlen function?

sizeof is keyword of C that can find size of a String constant including null character, but strlen is function which has been defined string.h and can find number of characters in a string excluding null character.

```
#include
void main(){
    int a,b;
    a=strlen("cquestionbank");
    b=sizeof("cquestionbank");
```

Questions And Answers (https://intellipaat.com/in question/soa-interview-questions/)

Top JBPM interview questions And Answers (https://intellipaat.com/in question/jbpm-interview-questions/)

Top Drools Interview Questions And Answers (https://intellipaat.com/in question/drools-interview-questions/)

Top Junit Interview Questions And Answers (https://intellipaat.com/in question/junit-interview-questions/)

Top Spring Interview Questions And Answers (https://intellipaat.com/in question/spring-interview-questions/)

```
    b=sizeof("cquestionbank");
    printf("%d  %d",a,b);
    getch();
}
```

Read C programming tutorial to learn about various operators in C (https://intellipaat.com/tutorial/c-tutorial/c-operators/).

Want to Learn C & Data Structure? Click Here (https://intellipaat.com/c-data-structures-training/#course-content)

6. What is the difference between

sprintf(…) writes data to the character array. The C library function sprintf () is used to store formatted data as a string. You can also say the sprintf () function is used to create strings as output using formatted data. The syntax of the sprintf () function is as follows:

```
int sprintf (char *string, const char *form, … );
```

Here, the *string will stand for the name of the array that will store the output obtained by working on the formatted data. The *form parameter will show the format of the output.
printf(…) writes data to the standard output device. The printf function is just a useful function from the standard library of functions that are accessible by C programs (https://intellipaat.com/tutorial/c-tutorial/some-programs-c/).

The behavior of printf is defined in the ANSI standard. If the compiler that you’re using conforms to this standard then all the features and properties should be available to you.

7. When does the compiler not implicitly generate the address of the first element of an array?

The compiler does not implicitly generate the address of the first element of an array whenever an array name appears:
– as an operand of the sizeof operator
– as an operand of & operator
– as a string literal initialize for a character array

Get a clear understanding of C Array in this (https://intellipaat.com/tutorial/c-tutorial/c-array/) engrossing C tutorial.

8. Is using exit() the same as using return?

No, the exit() function is used to exit your program and return() controls the operating system.
The return statement (https://intellipaat.com/tutorial/c-tutorial/c-functions/) is used to return from a function and return control to the calling function. If you make a return from the main() function, you are essentially returning control(operating system) to the calling function. In this case, the return statement and exit() function are similar.

Become C & Data Structure Certified in 16 hrs.

GET CERTIFIED

9. What is an lvalue?

An lvalue is an expression to which a value can be assigned. The lvalue expression is located on the left side of an assignment statement whereas an rvalue is located on the right side of an assignment statement.
Each assignment statement must have an lvalue and an rvalue. The lvalue expression must refer a storable variable in memory. It cannot be a constant.

10. What is the difference between goto, longjmp() and setjmp()?

- A goto statement implements a local jump of program execution whereas the longjmp() and setjmp() functions implement a nonlocal or far jump of the program execution.
- Generally, a jump in any execution should be avoided because it is not considered good programming practice to use such statements as goto and longjmp in your program.
- A goto statement simply bypasses code in your program and jumps to a predefined position. To use the goto statement, you give it a labeled position to jump to. This predefined position must be within the same function. You cannot implement goto between functions.
  However, when your program calls setjmp(), the current state of your program is saved in a structure of type jmp_buf. Later, your program can call the longjmp() function to restore the program's state as it was when you called setjmp().Unlike the goto statement, the longjmp() and setjmp() functions do not need to be implemented in the same function.
  There is a major drawback of using these functions: your program, when restored to its previously saved state, it will lose its references to any dynamically allocated memory between the longjmp() and the setjmp(). This means you will waste memory for every malloc() or calloc() you have implemented between your longjmp() and setjmp(), and your program will be inefficient.
  It is highly recommended that you avoid using functions such as longjmp() and setjmp() because they, like the goto statement, are quite often an indication of poor programming practice.

Know more about C Functions in this (https://intellipaat.com/tutorial/c-tutorial/c-functions/) riveting tutorial.

11. What is XOR linked list?

XOR linked list is a Memory Efficient Doubly Linked List. An ordinary Doubly Linked List requires space for two

An ordinary Doubly Linked List requires space for two address fields to store the addresses of previous and next nodes. A memory efficient version of Doubly Linked List can be created using only one space for address field with every node. This memory efficient Doubly Linked List is called XOR Linked List or Memory Efficient as the list uses bitwise XOR operation to save space for one address.
In the XOR linked list, instead of storing actual memory addresses, each node stores the XOR of addresses of previous and next nodes.

XOR List Representation:

Let us call the address variable in XOR representation npx (XOR of next and previous)

```
Node A:
npx = 0 XOR add(B) // bitwise XOR of zero and address of
B
Node B:
npx = add(A) XOR add(C) // bitwise XOR of address of A an
d address of C
Node C:
npx = add(B) XOR add(D) // bitwise XOR of address of B an
d address of D
Node D:
npx = add(C) XOR 0 // bitwise XOR of address of C and 0
```

12. What is ‘trie’ in data structure?

Trie is efficient information retrieval data structure. Using trie, search complexities can be brought to optimal limit (key length). If we store keys in binary search tree, a well balanced BST will need time proportional to M * log N, where M is maximum string length and N is number of keys in tree.
• Using trie, we can search the key in O(M) time. However, the penalty is on trie storage requirements.
• Each node of trie consists of multiple branches. Each branch represents a possible character of keys.
• We need to mark the last node of every key as leaf node.
• A trie node field value will be used to distinguish the node as leaf node (there are other uses of the value field).
• A simple structure to represent nodes of English alphabet can be as follows:

```
struct trie_node
struct trie_node
{
                int value; /* Used to mark leaf nodes */
                 trie_node_t *children[ALPHABET_SIZE];
};
```

Become Master of C & Data Structure by going through this online training course (https://intellipaat.com/c-data-structures-training/#about-course).

13. What do you understand by splay tree?

Splay tree is a self-balancing Binary Search Tree (BST). The main idea of splay tree is to bring the recently accessed item to root of the tree. This makes the recently searched item to be accessible in O (1) time if accessed again. The idea is to use locality of reference (In a typical application: 80% of the access are to 20% of the items).
Imagine a situation, where we have millions or billions of keys and only few of them are accessed frequently, which is very likely in many practical applications.
All splay tree operations run in O(log n) time on average, where n is the number of entries in the tree. Any single operation can take Theta(n) time in the worst case.

Go through this C Video (https://intellipaat.com/php-mysql-training/#course-preview) to get clear understanding of C Programming.

14. What is Treap?

Treap is a Balanced Binary Search Tree, but not guaranteed to have height as O(Log n). The idea is to use Randomization and Binary Heap property to maintain balance with high probability. The expected time complexity of search, insert and delete is O(Log n).
–>Each node of Treap maintains two values.

1. **Key** follows standard BST ordering (left is smaller and right is greater)
2. **Priority** Randomly assigned value that follows Max-Heap property.

15. How to implement LRU caching scheme? What data structures should be used?

We are given total possible page numbers that can be referred. We are also given cache (or memory) size (Number of page frames that cache can hold at a time). The LRU caching scheme is to remove the least recently used frame when the cache is full and a new page is referenced which is not there in cache.
–>We use two data structures to implement an LRU Cache.

1. **A Queue:** which is implemented using a doubly linked list. The maximum size of the queue will be equal to the total number of frames available (cache size).
   The most recently used pages will be near front end and least recently pages will be near rear end.
2. **A Hash:** with page number as key and address of the corresponding queue node as value. When a page is referenced, the required page may be in the memory. If it is in the memory, we need to detach the node of the list and bring it to the front of the queue.
   If the required page is not in the memory, we bring that in memory. In simple words, we add a new node to the front of the queue and update the corresponding node address in the hash. If the queue is full, i.e. all the frames are full, we remove a node from the rear of queue, and add the new node to the front of queue.

16. Suppose, there are two linked lists: L1 and L2 (of same lengths) that intersect at a particular node N1, which is a common endpoint to all other nodes. What are the possibilities to find N1?

Linear solution is possible. Have two pointers say P1 pointing to the first node of L1 and P2 to that of L2. Traverse through both the lists. If P1 reaches L1’s last node, point it to the first node of L2 and continue traversing.

Do the same thing for P2 when it reaches L2’s last node. (By doing this, we are balancing the difference in the length between the linked lists. The shorter one will get over soon and by redirecting to longer list’s head, it will traverse the extra nodes also). Finally, they will meet at the Intersection node.

17. Given two keys K1 & K2, write an algorithm to print all the elements between them with K1<=K2 in a BST.

- Linear solution is possible without using any extra space.
- Perform an inorder traversal.
- Once you find K1, print it and continue traversal now.
- Print all other traversed elements until you reach K2.

18. How many stacks are required to implement a Queue.

Two stacks are required to implement a Queue.

- **For Enqueue:** Take two stacks S1 and S2 and perform push on S1.
- **For Dequeue:** If S2 is empty, pop all the elements from S1 and push it to S2. The last element you popped from S1 is an element to be dequeued. If S2 is not empty, then pop the top element in it.

« Previous (https://intellipaat.com/interview-question/informatica-interview-questions/)
Next » (https://intellipaat.com/interview-question/oracle-dba-interview-questions/)

"2 Responses on Top C & Data Structure Interview Questions And Answers"

log in to reply (https://intellipaat.com/login/?redirect_to=https%3a%2f%2fintellipaat.com%2finterview-question%2fc-data-structure-interview-questions%2f)

K[illegible] Singh...
DECEMBER 21, 2015 AT 5:30 PM (HTTPS://INTELLIPAAT.COM/INTERVIEW-QUESTION/C-DATA-STRUCTURE-INTERVIEW-QUESTIONS/#COMMENT-16451)

great job!!

log in to reply (https://intellipaat.com/login/?redirect_to=https%3a%2f%2fintellipaat.com%2finterview-question%2fc-data-structure-interview-questions%2f)

D says
DECEMBER 20, 2018 AT 5:24 PM (HTTPS://INTELLIPAAT.COM/INTERVIEW-QUESTION/C-DATA-STRUCTURE-INTERVIEW-QUESTIONS/#COMMENT-16447)

Very..very..very nice..Thank you

Leave a Message

You must be logged in (https://intellipaat.com/login/?redirect_to=https%3A%2F%2Fintellipaat.com%2Finterview-question%2Fc-data-structure-interview-questions%2F) to post a comment.

© COPYRIGHT 2011-2019 INTELLIPAAT.COM. ALL RIGHTS RESERVED. VIEW DISCLAIMER

100% SECURE PAYMENTS. ALL MAJOR CREDIT & DEBIT CARDS ACCEPTED OR PAY BY PAYPAL.

TOP