# **Interview questions:**

**Please note people will also rate whether you are polite, and why your behavior properly through the on-site interview, so well behavior yourself.**

1. Why do you think you are a good fit for the job position you’re applying for?
2. Do you have any question for me?
3. I think we can show our interest on the product and team work of team we’re going to join;
4. For example, we can ask more details on the job position we’re applying for.
5. Get to know the company’s history before the job interview.
6. What’s your ideal job/ideal work environment?
7. Introduce yourself to us?
8. What are the most chanllegning problems/issues you debugged before?
9. What’s your expectation of the salary package?
10. How much of your day do you spend coding? How much code you write every day?
11. What’s the technical area you’re interested in the most?
12. What’s your greatest weakness? What’s your greatest strength?
13. How do you deal with the teammate conflict? E.g. for the investigation of an issue, you have different opinion on how to debug the issue with your teammate, how will you handle this conflict?
14. If you’re assigned a task with a strict deadline, but you know you’ll be not able to finish it on time, how will reply to the customer/your supervisor?
15. Why do you want to work as an engineer? Your career goals? Do you want to work as an engineer all the time? Or you plan to work as a manager after a few years?
16. Go over “Behavioral Preparation” in the Cracking the Coding Interview 150 Programming Questions and Solutions.pdf?

# **C programming- Arrays and Strings**

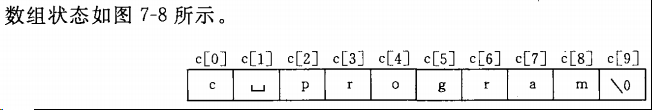
For all questions, please note you may be asked for calculating the running time.

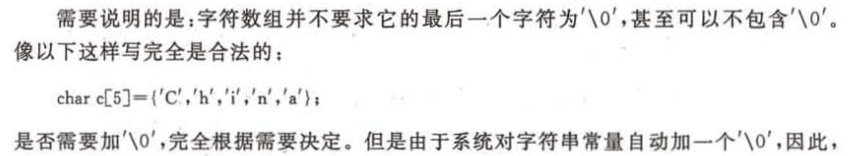
## **Cracking code 1.1:** Implement an algorithm to determine if a string has all unique characters.

What if you can not use additional data structures?

* **Cracking code 1.2:** Write code to reverse a C-Style String (C-String means that “abcd” is represented as five characters, including the null character )?
* **Cracking code 1.3:** Design an algorithm and write code to remove the duplicate characters in a string without using any additional buﬀer NOTE: One or two additional variables are fne  
  An extra copy of the array is not  
  FOLLOW UP  
  Write the test cases for this method
* **Cracking code 1.4:** Write a method to decide if two strings are anagrams or not
* **Cracking code 1.5:** Write a method to replace all spaces in a string with ‘%20’?
* **Cracking code 1.6:** Given an image represented by an NxN matrix, where each pixel in the image is 4  
  bytes, write a method to rotate the image by 90 degrees Can you do this in place?
* **Cracking code 1.7:** Write an algorithm such that if an element in an MxN matrix is 0, its entire row and  
  column is set to 0.
* **Cracking code 1.8:** Assume you have a method isSubstring which checks if one word is a substring of  
  another Given two strings, s1 and s2, write code to check if s2 is a rotation of s1 using  
  only one call to isSubstring (i e , “waterbottle” is a rotation of “erbottlewat”)
* **《From C语言谭浩强》：**
* **数组下标从0开始以及越界：**
* **7.2 用数组处理Fibonacci数列（not important）**
* **7.3 用起泡法对10个数排序**
* **字符串的基础知识：**

P162: 定义字符数组，如果初值个数小于数组长度，则只将这些字符赋值给数组中前面那些元素，其余的元素自动定为空字符（即’\0’）:





* **Write below functions by yourself：**
* Strcpy()
* Strncpy()
* The difference between strcpy() and memcpy()?
* Strcmp()
* Find substring: strstr()
* strlen()
* strcat() （low priority）
* 7.8 输入一行字符，统计其中有多少个单词，单词之间用空格分隔开**(not important)**
* **7.9 有3个字符串，找出其中最大者(not important)**
* **习题7.4 已有一个已排好序的数组，要求输入一个数后，按原来排序的规律将它插入数组中**

# **C programming- Linked Lists**

- Reverse the link list without using another data structure (usually ask for single linked list, practice double linked list as well, but it is not that important)

- Create a new single linked list/double linked list

- Insert a new node into a single linked list/double linked list

- Delete a new node into a single linked list/double linked list

- Find a new node into a single linked list double linked list

- There are sorted two linked list, every node contains a data field; merge the two linked list by sorting the data field again, do not use extra data structure

- cracking code 2.1:

Write code to remove duplicates from an unsorted linked list  
FOLLOW UP  
How would you solve this problem if a temporary buﬀer is not allowed?

- cracking code 2.2: Implement an algorithm to fnd the nth to last element of a singly linked list

- cracking code 2.3:

Implement an algorithm to delete a node in the middle of a single linked list, given  
only access to that node  
EXAMPLE  
Input: the node ‘c’ from the linked list a->b->c->d->e  
Result: nothing is returned, but the new linked list looks like a->b->d->e

-cracking code 2.4:

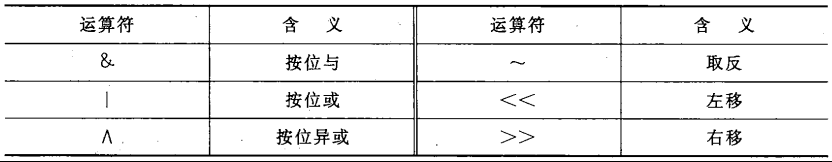
**2 4** You have two numbers represented by a linked list, where each node contains a single digit The digits are stored in reverse order, such that the 1’s digit is at the head of  
the list Write a function that adds the two numbers and returns the sum as a linked  
list  
EXAMPLE  
Input: (3 -> 1 -> 5) + (5 -> 9 -> 2)  
Output: 8 -> 0 -> 8

**-cracking code 2.5:**

**2 5** Given a circular linked list, implement an algorithm which returns node at the beginning of the loop  
DEFINITION  
Circular linked list: A (corrupt) linked list in which a node’s next pointer points to an  
earlier node, so as to make a loop in the linked list  
EXAMPLE  
input: A -> B -> C -> D -> E -> C [the same C as earlier]  
output: C

# **C programming- Bit Manipulation**

* 12.1 位运算



* 例12.1 取一个整数a从右端开始的4～7位
* 例12.2 循环移位: 要求将a进行右循环移位
* apple interview question: SetBit(int num, int bit\_Idx)
* Apple interview questions: ClearBit(int Num, int bit\_Idx)

- cracking code 5.1:

You are given two 32-bit numbers, N and M, and two bit positions, i and j Write a  
method to set all bits between i and j in N equal to M (e g , M becomes a substring of  
N located at i and starting at j).

EXAMPLE:  
Input: N = 10000000000, M = 10101, i = 2, j = 6  
Output: N = 10001010100

- cracking code 5.2:

Given a (decimal - e g 3 72) number that is passed in as a string, print the binary representation If the number can not be represented accurately in binary, print “ERROR

- cracking code 5.3:

Given an integer, print the next smallest and next largest number that have the same  
number of 1 bits in their binary representation

- cracking code 5.4:

Explain what the following code does: ((n & (n-1)) == 0)

Similar question in careercup:

https://www.careercup.com/question?id=14594825

- cracking code 5.5:

Write a function to determine the number of bits required to convert integer A to  
integer B.

Input: 31, 14  
Output: 2

- cracking code 5.6:

Write a program to swap odd and even bits in an integer with as few instructions as  
possible (e g , bit 0 and bit 1 are swapped, bit 2 and bit 3 are swapped, etc)

- cracking code 5.7:

An array A[1 n] contains all the integers from 0 to n except for one number which is  
missing In this problem, we cannot access an entire integer in A with a single operation The elements of A are represented in binary, and the only operation we can use  
to access them is “fetch the jth bit of A[i]”, which takes constant time Write code to  
find the missing integer Can you do it in O(n) time?

* Careercup:

Write a program to swap odd and even bits of a 32-bit unsigned integer

With as few instructions as possible.(bit-0 and bit-1 are swapped, bit-2 and bit-3 are swapped and so on)

<https://www.careercup.com/question?id=14922694>

* Careercup:

<You are given an integer, print its 4th least significant bit. https://www.careercup.com/question?id=5749560358993920>

* Careercup:

[Implement a function, set\_bit\_l\_to\_r(x,y,l,r).   
  
For bits l to r (both inclusive), if they are set in x, also set them in y. Do not change bits of y, if they are not in range l to r, or those bits are not set in x. l and r are 0-based.](https://www.careercup.com/question?id=5714482551586816)

<https://www.careercup.com/question?id=5714482551586816>

* CareerCup:

Swap every two bits in a unsigned char.. eg swap bits at 0 and 1st position, swap 2nd and 3rd position, swap 4th and 5th position etc…

<https://www.careercup.com/question?id=13586687>

* CareerCup:

Given two integers and two bit positions. Set the first integer between the two bit positions to be that of the second integer.

<https://www.careercup.com/question?id=13532675>

* CareerCup:

Given a function, take a number and the bit position and return true if that bit is set to 1 and false otherwise.

<https://www.careercup.com/question?id=13225741>

* CareerCup:

Calculate number of zeros in a given integer

<https://www.careercup.com/page?pid=bit-manipulation-interview-questions&n=2>

* CareerCup:

Deisgned a mask for a string M. which has to replace the bits of N at ith position. Given a the bit-length of sequence N as j. Design a function to return the masked data.

<https://www.careercup.com/question?id=11341181>

* CareerCup:

The function signature looks like this:

Boolean isPalindrome(int x)

It should return true if the bit pattern of x is the same as when you reverse it.

<https://www.careercup.com/question?id=9607833>

* CareerCup:

Write code to convert a hex string to a byte buffer

<https://www.careercup.com/question?id=310664>

* CareerCup:

Given two unsigned integers, write an efficient function which returns the no. of bits needs to be flipped of one to generate the other.

<https://www.careercup.com/question?id=9286520>

* CareerCup:

An array of integers of size n-1, all the elements are form[1,n]. Find the missing number. You can read only one bit in one operation, ie. To read A[i], you need to perform log(A[i]) operations.

<https://www.careercup.com/question?id=2466716>

* CareerCup:

Given an 32-bit integer X, swap the i-th and j-th bit.

<https://www.careercup.com/question?id=1941662>

* **CareerCup: Bit Manipulation Interview Questions**

If having time, go over other questions from below link(low priority):

https://www.careercup.com/page?pid=bit-manipulation-interview-questions&n=1

# **C programming- Brain Teasers(low priority)**

Please refer to chapter 6 of cracking code.(low priority)

# **C programming- Recursion(low priority)**

- Try to understand the concept of recursion; and be able to write a program to implement the code logic either by recursion/for loop

-cracking code: 8.1:

Write a method to generate the nth Fibonacci number.

- If have time, go over other questions from cracking code(low priority)

# **C programming- Sorting and Searching**

- Fundamental concepts: You can go over my UTD ppt for different sorting algorithm first.

- Be able to write program on those sorting algorithm and calculating the time complexity:

Bubble sort

Selection Sort

Merge Sort

Quick Sort

Bucket Sort(not important)

- cracking code 9.1:

You are given two sorted arrays, A and B, and A has a large enough buﬀer at the end  
to hold B Write a method to merge B into A in sorted order.

- cracking code 9.2:

Write a method to sort an array of strings so that all the anagrams are next to each  
other.

- cracking code 9.3:

Given a sorted array of n integers that has been rotated an unknown number of  
times, give an O(log n) algorithm that fnds an element in the array You may assume  
that the array was originally sorted in increasing order  
EXAMPLE:  
Input: fnd 5 in array (15 16 19 20 25 1 3 4 5 7 10 14)  
Output: 8 (the index of 5 in the array)

- cracking code 9.4:

If you have a 2 GB fle with one string per line, which sorting algorithm would you use  
to sort the fle and why?

- cracking code 9.5:

Given a sorted array of strings which is interspersed with empty strings, write a method to fnd the location of a given string  
Example: fnd “ball” in [“at”, “”, “”, “”, “ball”, “”, “”, “car”, “”, “”, “dad”, “”, “”] will return 4  
Example: fnd “ballcar” in [“at”, “”, “”, “”, “”, “ball”, “car”, “”, “”, “dad”, “”, “”] will return -1

-cracking code 9.6:

Given a matrix in which each row and each column is sorted, write a method to fnd  
an element in it

-cracking code 9.7: (low priority)

# **C programming- Stacks and Queues (low priority)**

- Go over my UTD ppt to under the concepts, and be able to write a program on push data into stack or pop data out of stack.

- The difference between Stacks and Queues.

- If have time, take a look the questions from cracking code book(low priority)

# **C programming- Tree (low priority)**

- Quickly go through my UTD ppt to under the concepts on binary-tree(In-order, Pre-order, Post-Order,Insert Node). (low priority)

# **C programming- C pointers**

- Please point out the error on below c function:

void swap(int \*p1, int \*p2)

{

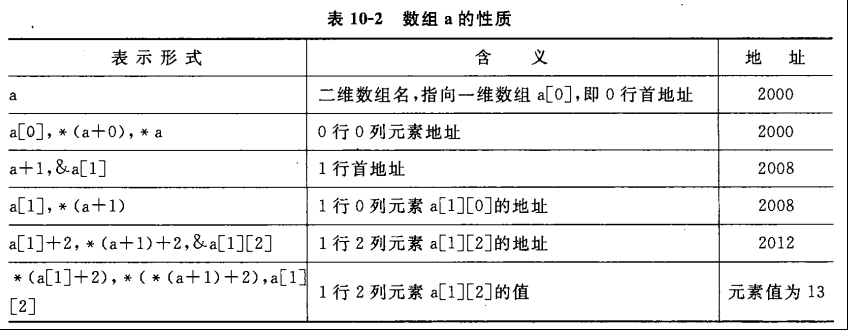
int \*temp;

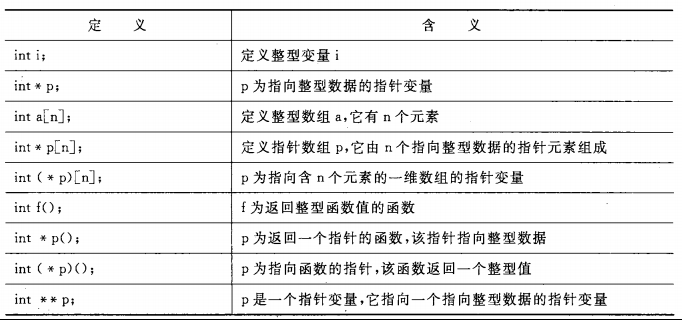
\*temp = \*p1;

\*p1 = \*p2;

\*p2 = \*temp;

}





- 10.20: 用指向指针的指针的方法对5个字符串排序并输出

- C语言指针面试题：

<http://blog.csdn.net/zccst/article/details/4268583>

https://wenku.baidu.com/view/b2cdf27d31b765ce050814e5.html

# **C programming- Macro definition**

- 宏定义面试题：

<http://blog.csdn.net/zjuxsl/article/details/44136209>

-带参数宏定义：

http://hbprotoss.github.io/posts/cyu-yan-hong-de-te-shu-yong-fa-he-ji-ge-keng.html

# **C programming- Typedef**

- [C语言typedef的用法](http://www.cnblogs.com/afarmer/archive/2011/05/05/2038201.html):

http://www.cnblogs.com/afarmer/archive/2011/05/05/2038201.html

# **C programming- byte alignment in c**

- [C语言字节对齐问题详解](http://www.cnblogs.com/clover-toeic/p/3853132.html)

http://www.cnblogs.com/clover-toeic/p/3853132.html

# **C programming- inline function**

http://huxiongwei.spaces.eepw.com.cn/articles/article/item/85841

# **C programming- volatile key word**

- C语言中volatile关键字的作用：

http://blog.csdn.net/tigerjibo/article/details/7427366

# **C programming- concast key word**

- 详解C语言中const关键字的用法：

http://www.jb51.net/article/70831.htm

# **C programming- sizeof**

- Sizeof面试题：

http://www.cnblogs.com/xuyuantao/archive/2010/08/15/1800266.html

# **C programming- static的用法**：

- static有什么用途？（请至少说明两种）

1)在函数体，一个被声明为静态的变量在这一函数被调用过程中维持其值不变。

2) 在模块内（但在函数体外），一个被声明为静态的变量可以被模块内所用函数访问，但不能被模块外其它函数访问。它是一个本地的全局变量。

3) 在模块内，一个被声明为静态的函数只可被这一模块内的其它函数调用。那就是，这个函数被限制在声明它的模块的本地范围内使用

<http://blog.csdn.net/tianmohust/article/details/6686196>

# **Linux -file operation:**

* open()/close()/write()/read()/lseek():

不带缓冲是指直接使用系统调用，在内核与调用者之间没有缓冲，但是在内核与设备之间还是有缓冲高速缓存或页面高速缓存存在的。

May ask you to write a example code on file operation during the interview.

* Difference between open()/read()/write() and fopen()/fread()/fwrite():

<http://www.cnblogs.com/ldp-web/archive/2011/10/21/2220180.html>

* stat函数用法：

int stat(const char \*restrict pathname, struct stat \*restrict buf);  
提供文件名字，获取文件对应属性。  
int fstat(int filedes, struct stat \*buf);  
通过文件描述符获取文件对应的属性。  
int lstat(const char \*restrict pathname, struct stat \*restrict buf);  
连接文件描述命，获取文件属性。

# **Linux process:**

## Fork ()

* fork (): can write simple example code by using fork ()
* what are the process items that are inherited in a child created using fork ()?

What are the process items that are different from the process’s parent?

fork () creates a new process by duplicating the calling process. The new referred to as the child, is an exact duplicate of the calling process, referred to as the parent, except for the following points:

* The child has its own unique process ID, and this PID does not match the ID of any existing process group (setpgid(2)).
* The child's parent process ID is the same as the parent's process ID.
* The child does not inherit its parent's memory locks (mlock(2), mlockall(2)).
* Process resource utilizations (getrusage(2)) and CPU time counters (times(2)) are reset to zero in the child.
* The child's set of pending signals is initially empty (sigpending(2)).
* The child does not inherit semaphore adjustments from its parent (semop(2)).
* The child does not inherit record locks from its parent (fcntl(2)).
* The child does not inherit timers from its parent (setitimer(2), alarm(2), timer\_create(2)).
* The child does not inherit outstanding asynchronous I/O operations from its parent (aio\_read(3), aio\_write(3)), nor does it inherit any asynchronous I/O contexts from its parent (see io\_setup(2)).

The process attributes in the preceding list are all specified in POSIX.1-2001. The parent and child also differ with respect to the following Linux-specific process attributes:

* The child does not inherit directory change notifications (dnotify) from its parent (see the description of F\_NOTIFY in fcntl(2)).
* The prctl(2) PR\_SET\_PDEATHSIG setting is reset so that the child does not receive a signal when its parent terminates.
* Memory mappings that have been marked with the madvise(2) MADV\_DONTFORK flag are not inherited across a fork().
* The termination signal of the child is always SIGCHLD (see clone(2)).

Note the following further points:

* The child process is created with a single thread -- the one that called fork(). The entire virtual address space of the parent is replicated in the child, including the states of mutexes, condition variables, and other pthreads objects; the use of pthread\_atfork(3) may be helpful for dealing with problems that this can cause.
* The child inherits copies of the parent's set of open file descriptors. Each file descriptor in the child refers to the same open file description (see open(2)) as the corresponding file descriptor in the parent. This means that the two descriptors share open file status flags, current file offset, and signal-driven I/O attributes (see the description of F\_SETOWN and F\_SETSIG in fcntl(2)).
* The child inherits copies of the parent's set of open message queue descriptors (see mq\_overview(7)). Each descriptor in the child refers to the same open message queue description as the corresponding descriptor in the parent. This means that the two descriptors share the same flags (mq\_flags).
* The child inherits copies of the parent's set of open directory streams (see opendir(3)). POSIX.1-2001 says that the corresponding directory streams in the parent and child may share the directory stream positioning; on Linux/glibc they do not.
* vfork() vs fork()

The intent of vfork was to eliminate the overhead of copying the whole process image if you only want to do an exec\* in the child. Because exec\* replaces the whole image of the child process, there is no point in copying the image of the parent.

if ((pid = vfork()) == 0) {

execl(..., NULL); /\* after a successful execl the parent should be resumed \*/

exit(127); /\* terminate the child in case execl fails \*/

}

For other kinds of uses, vfork is dangerous and unpredictable.

With most current kernels, however, including Linux, the primary benefit of vfork has disappeared because of the way fork is implemented. Rather than copying the whole image when fork is executed, copy-on-write techniques are used.

* (Amazon)How many processes are created in this snippet?

Main()

{

Fork();

Fork() && fork() || fork()

Fork()

}

<https://www.careercup.com/question?id=189793>

* Fork() question:

[1) How many times "SIVA" will be printed](https://www.careercup.com/question?id=7772670)

[i = 0 ;](https://www.careercup.com/question?id=7772670)

[while (i < n )](https://www.careercup.com/question?id=7772670)

[{](https://www.careercup.com/question?id=7772670)

[fork();](https://www.careercup.com/question?id=7772670)

[printf("\n SIVA");](https://www.careercup.com/question?id=7772670)

[i++;](https://www.careercup.com/question?id=7772670)

[}](https://www.careercup.com/question?id=7772670)

[2) How many times "SIVA" will be printed](https://www.careercup.com/question?id=7772670)

[i = 0 ;](https://www.careercup.com/question?id=7772670)

[while (i < n )](https://www.careercup.com/question?id=7772670)

[{](https://www.careercup.com/question?id=7772670)

[printf("\n SIVA");](https://www.careercup.com/question?id=7772670)

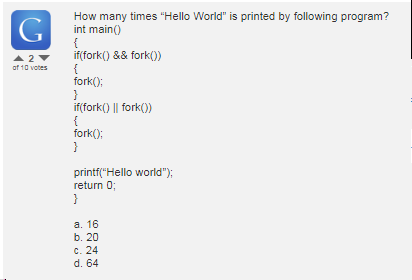
[fork();](https://www.careercup.com/question?id=7772670)

[i++;](https://www.careercup.com/question?id=7772670)

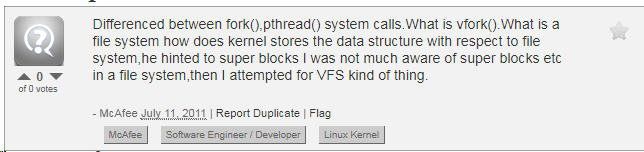
[}](https://www.careercup.com/question?id=7772670)

<https://www.careercup.com/question?id=7772670>

* Fork() question from google:

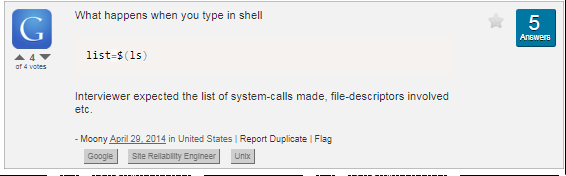


* Linux kernel question from McAfee:



<https://careercup.com/question?id=9728714>

* Linux kernel question from Google:



<https://www.careercup.com/question?id=5726414556889088>

* Qualcomm interview question on write thread:



* <https://www.careercup.com/question?id=14949401>

## Linux Kernel Interview Questions:

[What is the difference between a process and a thread?](https://stackoverflow.com/questions/200469/what-is-the-difference-between-a-process-and-a-thread)

Process  
Each process provides the resources needed to execute a program. A process has a virtual address space, executable code, open handles to system objects, a security context, a unique process identifier, environment variables, a priority class, minimum and maximum working set sizes, and at least one thread of execution. Each process is started with a single thread, often called the primary thread, but can create additional threads from any of its threads.

Thread  
A thread is an entity within a process that can be scheduled for execution. All threads of a process share its virtual address space and system resources. In addition, each thread maintains exception handlers, a scheduling priority, thread local storage, a unique thread identifier, and a set of structures the system will use to save the thread context until it is scheduled. The thread context includes the thread's set of machine registers, the kernel stack, a thread environment block, and a user stack in the address space of the thread's process. Threads can also have their own security context, which can be used for impersonating clients.