

Microsoft interview-SDET

Interview_1 : 3 men red and blue hat problem

Interview_2 : Why I wanted to work for them

Interview_3 : Why should we hire you?

Interview_4 : Mainly DSA and binary tree

Interview_5 : Lowest common substring, BFS based questions

Interview_6 : Coding and general questions. how do you print duplicate characters from a string?

How do you check if two strings are anagrams of each other?

Interview_7 : Projects and resume related questions

Interview_8 : All sorts of brain teasers (monte hall problem, etc)

Interview_9 : How to insert a Node in a Linked list.

Interview_10 : Coding, logical thinking, testing strategies, problem solving

Interview_11 : Are two words palindrome of each other?

Interview_12 : Divide two numbers without using / or % and return quotient.

Interview_13 : Difference between == and === in javascript? How to use selectors in Selenium?

Difference between readonly and const in c#? Mock testing for services? Use of .this keyword?

Interview_14 : How can you test a calculator.

Interview_15 : The interviewer had the code from the previous interviewee still on the white board in the room. He asked me to identify what the code did (printed the contents of a binary tree through a BFS using a queue). Then asked me to identify the bugs with the code and suggest how to fix it (null-pointer exceptions when printing out the values from the queue and checking for nulls before adding a value to the queue).

Interview_16 : Measure 4 liters from 2 jars with 5 liters and 3 liters

Interview_17 : Write a query to get 3rd largest value from a table

Interview_18 : How would you reverse a string?

Interview_19 : Design an elevator control system for a busy 5 floor building with 4 elevators.

Interview_20 : What is radix sort?

Interview_21 : * Merge two sorted linked list. The merged list should also be sorted.

* Compact a string. i.e remove spaces

* traverse a link list containing char* as data. test cases for the same.

* reverse a string. test cases for the same.

* remove all the given characters from a string.

Interview_22 : Non-technical:

1-Why did I choose to work in Microsoft?

Microsoft interview-SDET

2-Why should they choose me?

3-What has been my most challenging project?

4-And how was my english?

Interview_23 : Technical:

Most of the questions consisted in problems related with Strings in the language I put as my main in the resume. I remember 2 questions (sorry for not remembering more)

1- Make a method which receives an String and returns its reversed version.

2- Find the longest palindrome in a sentence.

Interview_24 : College interview was easy. Simple Merge Sort.

Interview_25 : Inorder tree traversal + a logic question

Interview_26 : Navigating through a matrix of 0s and 1s, Dynamic programming

Interview_27 : Removing duplicates from a single linked list

Interview_28 : Given stock prices over a period, find out the buy-sell times for the most profit possible. No shorting is allowed.

Interview_29 : Given a Stack with n elements, write a method that always keeps the max of the stack?

Interview_30 : Why Microsoft

Interview_31 : Lots of "is this ready to ship to the world" questions, can't remember specifics.

Interview_32 : what the most complex question I have had to solve is and how I tackled it

Interview_33 : Question about Data encryption over HTTPS websites and he drilled really deep asking questions like how secret and private keys work, how the secret key is shared, and how encryption of data actually takes place on websites.

Interview_34 : Nothing that difficult. Just know your basics about data Structures, Algorithms and Testing if interviewing for a SDET position.

Interview_35 : You are given two buckets. One holds 3 gallons and the other holds 5 gallons. Have do you make 4 gallons of water?

Interview_36 : How do you simulate the functionality of a queue using 2 stacks

Interview_37 : Can't remember the actual questions now, but those were usual interview ones. No crazy trees or graphs, but mostly combinatorics-like, string/array manipulation, basic geometrical applications etc. No brain teasers, only technical.

Interview_38 : The reverse in-order traversal stumped me slightly; my solution was to do an in-order traversal, adding elements to a FIFO stack. Then after the traversal, pop off the elements of the stack.

Interview_39 : Given a regular phone number dial pad, write code that will list out all the possible letter combinations that can be made given three numerical digits as input.

Microsoft interview-SDET

For example, the base function would look like this:

```
function numberToChars(int *numbers) {  
    // would return an array of different character combinations  
    // if input was 123  
    // output: adg, adh, adi, bdg, etc...  
}
```

Interview_40 : How would you test a function this function and make sure it works? The function would take in two file paths as parameters and then copy the contents of the first file path into the second one?

Interview_41 : Explain how to test this code, write atoi, etc

Interview_42 : Design a backup system. (no requirements, no more details, and very vague)

Interview_43 : The most difficult question was the second question.

Interview_44 : nothing was like any unexpected questions. all questions were almost easy.

Interview_45 : Why do you want to work here?

Interview_46 : Honestly don't remember, but I was not stuck anywhere so there was nothing too ridiculous. But they do expect you to think of edge cases and test cases, especially for an SDET.

Interview_47 : One or two design questions.

Remaining all were on DS and algo e.g. BST, linked lists, practical implementation of these concepts.

Some were on the OS basics.

Interview_48 : Should we ship the product that has known bugs and issues?

Interview_49 : Puzzle to tests data structures

Interview_50 : Every thing was basics. I screwed it up

Interview_51 : Initial round was a online mcq round comprising of 15 technical questions. Duration was 30 minutes. Unfortunately kicked out of first round

Interview_52 : First round interview asked to write program that took a file of text, and then counted and stored all words from file.

Interview_53 : Given a array of integers, write code to find the max possible sum of sub-arrays. Use arithmetic progression.

Interview_54 : Given any two nodes in a binary tree, develop program that determined their highest common ancestor.

Interview_55 : Technical coding question

Microsoft interview-SDET

Interview_56 : weakness

Interview_57 : Mainly questions relating to data structures and testing them in extreme cases.

Interview_58 : handle two int_max inputs

Interview_59 : Hardest challenge faced?

Interview_60 : None really, how would you implement a queue with 2 stacks.

Interview_61 : Why do you choose M.S.

Interview_62 : Most question were moderate to difficult.

Interview_63 : all normal questions on book, algorithms, data structures etc.

Interview_64 : Shortest path question.

Interview_65 : Implement a Find-Replace function given a string of text, search string, and replace string. Also a seemingly simple question became very difficult and complex when looked at through the eyes of a tester. A question involved writing a function to classify a triangle given its side lengths.

Interview_66 : final round onsite interview was probably the group manager which you will be selected on, so don't fail it

Interview_67 : The last interviewer asked some network and OS questions, which are unexpected.

Interview_68 : How would test a machine on moon that does arithmetic operation based on input from earth and returns value to earth

Interview_69 : They asked me to first design a function to do something (don't remember) and code it on the whiteboard. Then they asked me how I would test it.

Interview_70 : Motivation and some technical question to resolve real problem

Interview_71 : Find if two nodes in a binary tree have a common ancestor.

Interview_72 : Need to be very precise about the test case than just writing the code.

Interview_73 : what are decision support systems?

Interview_74 : process "as the most important strategic planning your company needs." Why? Because every new employee will either improve the organization or lower it – and managers should be seeking and hiring employees who will enable the company to grow and to become more profitable.

McKeown provides four tips for hiring great people: 1) Forget trying to ask magic bullet questions, 2) Ensure you have clearly defined what it will take to be successful in the position, 3) Test candidates, and 4) Include others in the hiring process.

Interview_75 : Test an Function that converts string to integer. String here can be of any length!

Interview_76 : Please implement a memory allocate function

Interview_77 : some expected technical and business problems

Microsoft interview-SDET

Interview_78 : Given a 2D array of boolean values determine the largest square sub-array containing only 1s.

Interview_79 : Hardest question was how to sort a linked list using recursion only. I knew the iterative solution but couldn't come up with recursion. Very difficult.

Interview_80 : Reverse a linked list

Interview_81 : Program to check if a "ransom note", has been taken from a particular news paper article. Which meant you were given two arrays of strings, and you had to check if the first array, could have been constructed from the second set

Interview_82 : Phone interview coding question was to check if a string was a pallindrome

Interview_83 : Most difficult question: How would you develop a phone for the blind.

Interview_84 : What data structure did you use in your past project.

Interview_85 : Everything pretty easy and saw multiple times in interview question websites. Linked-lists, BSTs mostly

Interview_86 : How does the internet work?

Interview_87 : Time complexity

Interview_88 : All basic algorithm and data structure questions with some talk about design and testing.

Interview_89 : Basic algorithm and data structure questions.

Interview_90 : Similar to questions on glassdoor, careercup and cracking the coding interview. How to test a vending machine.

Interview_91 : They really wanted you too apply concepts from your other projects to your other projects which makes you think since you might think those projects are separate.

Interview_92 : Explain everything that happens from the time you enter a web address in the browser, until the page has loaded. Be as detailed as possible.

Interview_93 : Numerical array. Find a consecutive part of the array with the largest sum.

Interview_94 : Testing a toaster in space

Interview_95 : Need to be good in data structures

Interview_96 : Search a string to see if it contains a given substring. The given substring can have a '*' - a wild card character which corresponds to skipping one or more characters in the string being searched. Asked qualifiers: Can there be a wildcard character in the string being searched? [yes] ascii? [unicode]

Interview_97 : behavior questions and technical questions

Interview_98 : Standard coding and design questions. Some questions about my background.

A tedious question was to convert a string representing a roman number to an integer. This question is not hard to solve, but tedious to code.

Microsoft interview-SDET

Interview_99 : One I remember is to rotate an image 90 degrees in place, and how to test your code.

Interview_100 : No difficult question. Some questions are tricky.

Interview_101 : Question on binary trees and test cases

Interview_102 : how to make it $O(n)$

Interview_103 : given two linked lists with a digit in each node, add the two linked lists together. the result must be a linked list with one digit in each node. use only one iteration of the two input lists.

Interview_104 : basic programming problems, recursive functions, linked list.

Interview_105 : On campus interview questions:

- 1) Find the middle of a linked list
- 2) Find all the paths in a binary tree that sum up to a given value
- 3) Write test cases for bing (yeah, really!)

On site interviews:

- Interview 1: After discussing past projects. Write test cases for a routine that takes input 2 Cartesian coordinates and draws a line joining them, but the display pane can show points only such that x and y are between -100 and 100. How would you design it for a blind person. Choose the best 6 test cases and why.

- Interview 2: After discussing my resume. Write code to find all the palindromes in a string. Code both naive and efficient approaches. What if the string had spaces also, like "race car", which is also a palindrome. What if characters similar to an alphabet, from other languages were also be considered as the same character. What if you were implementing a web service with a page that takes input string in a textbox and displays all the palindromes. What test cases would you write to see the service works fine. What of your service was being used only by people in Germany. How can you optimize?

- Interview 3: After resume discussion, talks for 5 minutes about his team, MS Office Sharepoint. Just 1 question for the entire interview. Given an API with a string argument and an int return type, what could it be doing. Imagine the different possibilities and how would you decide the testcases, based on the functionality of the API you thought of. Think on test that assure sanity, +ve examples, -ve examples, security, scalability of the API.

- Interview 4: Discusses my resume, talks about his team and shortly about his experience of 20 years at MSFT (at which I instantly remarked, I would have never been able to pull that off!). Says that I see from your feedback that you are more of a problem solver, so I would like to test your imagination. Asks me to implement `string.Split(some char)` API of C#. Write test cases that would fail my code. Write code to fix those failures, iterate till all are resolved. At the end, tries to sell me the SDET position, says it's lots of fun and SDE/SDET switching happens every couple of years.

Microsoft interview-SDET

Otherwise, MSFT pampers its on-site invitees. They pick you up from the hotel and airport in a limo, \$75/day just for meals, crazy discounts in the MSFT store, free taxi/rental car to explore the city and \$50 reimbursable, apart from meals and taxis if you want to buy other stuff.

Interview_106 : There were no difficult or unexpected questions.

for instance:

- sort an array with just two types of elements, optimize your algorithm

- find common letters in two strings and put them in a sorted order

- compare sorting algorithms

Interview_107 : Nothing scary. Standard brainteasers and behavioral questions with some simple coding any student should be able to do.

Interview_108 : The question is easy, it seems they care more about how you think.

Interview_109 : No questions were difficult

1) How to reverse a string

2) How do you increase profits in a retail store.

Interview_110 : The coding question was how to detect if a singularly linked-list is cyclic. Also discuss optimization of the solution and code it in any language of your choice.

Interview_111 : Pretty standard questions about technical background, why you want to work there, etc.

Interview_112 : Reverse a string without using a buffer?

Interview_113 : balance, 9 balls, one weighs less, how many times do you have to use the balance to find the smallest ball

Interview_114 : write char * * argv

Interview_115 : string matching, unicode

Interview_116 : Write the algorithm for this: If you have a string aaabbc the output should be a3b2c1, for the string aabcc the output should be a2b1c2 and so on. You should not allocate memory for a new string and you can do this only by reading each character once.

Ps: if the result is bigger than the original string (like abc -> a1b1c1) just return null or some message that makes that clear.

Interview_117 : Array sort.

Interview_118 : he asked questions about pointers in C/C++

Interview_119 : They asked me a lot about linked lists!

Microsoft interview-SDET

Interview_120 : Write a function that will remove all repeated elements of a given array of integers, then return the shortened array.

Interview_121 : The most difficult part is I am not that interested in this position.

Interview_122 : "How would you test the new Microsoft 'Spoon'" -- Basically they told me that microsoft was coming out with the latest and greatest in spoon technology and I was to test it. They were vague on all the questions I asked (everyone goes out of their way to tell you to ask plenty of questions, but it seemed to help absolutely nothing in this case).

Interview_123 : Random objects were held up, and I was asked how I would test them.

Interview_124 : No other question asked.

Interview_125 : Suppose there's a rabbit on the edge of a stream. In the stream, at unit sized spaces, there are potentially rocks, i.e. for each space in the stream, there is either a rock there, or there isn't. The rabbit can hop onto a rock, but can't swim.

If the rabbit can initially jump one space, then for each subsequent jump, jump $n + 1$, n , or $n - 1$ spaces, where n is the number of spaces previously jumped, design an algorithm to determine if the rabbit can successfully cross an arbitrary stream.

Interview_126 : Make a deep copy of a linked list where each node contains two pointers, next and random. Random points to some node in the list.

Interview_127 : Nothing too difficult. I was expecting some algorithm related questions.

Interview_128 : Just be thorough with the data structures (trees, strings, recursion etc) and practise a lot. Most of the questions are similar to programming questions found in a book, and if you practise enough, you will be confident for all the questions. Also, testing questions are always unique (like test a vending machine), and make sure you cover all the test cases.

Interview_129 : Write a program to return true if the string is a palindrome

Interview_130 : Nothing really out of the ordinary. If you practice with cracking the coding interview and similar books (practice with, mind you, not learn from) you should be golden.

Interview_131 : Start with a random linked list--a linked list with a second node pointer that can point to some random node in the list, or to nothing at all. Implement a method that will generate a deep copy of the list

Interview_132 : Write a compiler subroutine for ensuring all brackets and parentheses are paired. If an unpaired bracket/parenthesis is found, report its line number.

Interview_133 : Write a function that takes in 3 sides (lengths) of a triangle and determine if it is a valid triangle. (see, it's not too bad, remember the basics and you'll be fine)

Interview_134 : Given a linked list, swap every pair of elements (so elements 1 and 2 swap, 3 and 4 swap, etc.)

Interview_135 : You have a tree where every node has 1 sibling and 1 child. My interviewer said this structure would be used to implement a File menu. Each node had a boolean T/F value. I had to

Microsoft interview-SDET

write code to search the tree and find the first value that was T. All siblings of the same level had to be searched first, before any children were searched.

Interview_136 : Sort a data file containing 1 million elements.

Interview_137 : merge two sorted arrays in sorted order?

Interview_138 : Technical questions are very difficult, but if you are well prepared on the requirement of the job position you will answer to most of them. Programming excersice are very difficul and challenging.

Interview_139 : the first phone screening was about my previous tasks and projects. the second one started with my experience and then some testing questions.

Interview_140 : Look up career cup

Interview_141 : Tell me about a project you have worked on and any problems you faced

Interview_142 : Creating a detailed test plan for functions related to gaming, e.g. collision detection

Interview_143 : Problem of concurrent transactions done by two persons of a joint account at two different ATM Machines. How is it managed without introducing any inconsistency in balance of the account holders?

Interview_144 : Read Binary Tree from file and efficient store it back.

Interview_145 : The questions that are most difficult are the technical questions, where interviewers will ask a candidate to solve a real-life problems and write software code in the interview

Interview_146 : Write a method to reverse an arbitrary integer. Then find all the cases to test it.

Interview_147 : where u want to see in 5 years

Interview_148 : Various algorithm - sweep lines, coloring

Interview_149 : About a Database of Project that i did in College. I was asked to implement that in alternate way.

Interview_150 : what was the one question you had expected ut were not asked?

Interview_151 : There was nothing difficult or unexpected about the interview.

Interview_152 : Implement a single-linked list class.

Interview_153 : Reverse the words in a sentence but keep the space.

Interview_154 : Consider a stack of N number of cards which are piled up and in facing down. Each card has a unique number from the range 1 to N. The card is stacked in such a way that it exhibits the following behavior:

Take the first card and put it under the stack without revealing. Now the next card on the top will have the number 1 on it. Next take 2 cards one after the other and put is under the stack without revealing. Yes you guessed it right - the next card on the top will reveal a value of 2. This goes on.

Microsoft interview-SDET

Eg. for such a series : 9,1,8,5,2,4,7,6,3,10 [for N=10]

Write a program to generate such a series for a given N number of cards so that this behavior can be exercised.

Interview_155 : I was asked to count from 1 to N among M people in a loop. People counted as N will be removed and keeping counting from 1 to N until only one people left. Output the position of the last people.

Interview_156 : The third question is a bit unexpected. I was spending a lot of time thinking how to write this efficiently, but didn't finish it. I should just write a straight forward one, and try to improve it later.

Interview_157 : Data structures and algorithms along with a few design questions

Interview_158 : It includes a lot of behavior questions such as why Microsoft, talk about what you will do when meet trouble in work. and one technical questions that need me to reverse a input string.

Interview_159 : given a long input string with lots of delimiters in the white board, write a c-code to change the given string it to another format based on some rules.

Interview_160 : Design MSN messenger. This was not difficult for me but unexpected.

Interview_161 : I did not get any particularly challenging algorithms/data structures questions, but one problem had a constant-time solution that was not immediately obvious. So before jumping in think whether there could be a trick that trivializes the problem.

Interview_162 : The most difficult in that interview is linked list, although it's not difficult at all. The quesiton is to remove all even numbers from the linked list

Interview_163 : Given a binary tree, how would you set the keys/values of all the nodes and their child pointers to null. No language restriction.

Do it iteratively in $O(N)$ time with $O(1)$ space complexity where N is the number of nodes in the tree.

Other Details:

- Tree is just a regular Binary Tree and doesn't have the BST property.
- It is not guaranteed to be balanced.
- You may do whatever you want to the tree however, you must ensure that all the nodes in the tree and their left/right pointers are set to null.

Interview_164 : Maze question.

Interview_165 : Boundary case of the code question.

Interview_166 : Actually, the questions are not hard. For example, binary search, validate parentheses, find the kth element, etc.

Microsoft interview-SDET

Interview_167 : Write an algorithm that takes function names from a program and will figure out if it is made up of valid words (1 or more).

Interview_168 : Convert the roman numbers to interger values

Interview_169 : tell everything you know about enum in java.

Interview_170 : c) WAP to Print last n lines of a big log file.

Interview_171 : Questions were easy. Asked about Queue

Interview_172 : Find if given number matched sum of two elements in sorted array.

Interview_173 : Nothing you shouldnt expect.

Interview_174 : 1. Give a number k, output how many prime numbers between 2 and k (inclusive)

2. Give two strings a and b, tell whether b is a sub sequence of a (not substring), using recursion first, then using loops, then tell which one is better.

3. If a database query manager buffer is filled without any more space, what kick-out mechanisms would you use? FIFO, Oldest, Least Recently Used etc. Why?

Interview_175 : Linked List Related

Interview_176 : Nothing specific.

Interview_177 : Recursivity, BST, string manipulation

Interview_178 : All problem had some tricks but be calm as they are looking for problem solving approach..

Interview_179 : Data structure questions

Interview_180 : Standard questions found in interview prep sites and books.

Interview_181 : Well, I try to use non-recursive method to generate the permutations of a given string, but I forget that I only have 10min to write the code.

Interview_182 : Nothing to was out of the ordinary

Interview_183 : Alarm clock for blind people.

Interview_184 : BFS

Interview_185 : In your opinion what do you consider "good" code?

Interview_186 : Nothing really difficult.

Interview_187 : The algorithm question

Interview_188 : Given a Binary Tree (not Search Tree) , find the lowest common ancestor.

Interview_189 : Difference between calloc() and malloc()

synopsys-rd-interview-questions

0

of 0 votes

13 Answers

What are the differences between new (in C++) & malloc (in).

Interview_190 : None. CS fundamentals. LList, Sorting, Arrays etc.

Interview_191 : Given an array of a 999,999 numbers, from 1 to a 1,000,000, there's one number missing. The array is not sorted, and you can only access the array using a function `getNumber()`, which pops one number from the array. Each number in the array can only be accessed once. There's one number missing, from 1 to 1 million. which number is this?

Interview_192 : give an array of integer, find the medium number without sorting the array

Interview_193 : what was your most difficult programming project and how were you able to work through it?

Interview_194 : how many lines was your longest coding project and what did it do?

Interview_195 : How would you test a wireless keyboard?

Interview_196 : How would you design a calculator for second graders?

Interview_197 : You have 9 marbles and one is slightly heavier than the rest. You also have a scale. How would you determine the heavier marble using the scale only twice?

Interview_198 : Given a string, find whether it has any permutation of another string. Need to be efficient

Interview_199 : Write a program takes in a string and a delimiter, and uses that delimiter to split a string and then will reverse the characters in every word (or jumble of characters between the delimiters), stuffing them back into a string when finished. ('The dog walks' becomes..... 'ehT god sklaw')

Interview_200 : Write a program that reads in a string and prints the character once it has been found to have duplicates (2 or more). The program should only print the character once, not every time a duplicate is found. The program should be faster than $O(n^2)$.

Interview_201 : Write a program that takes in a string and removes any duplicate spaces (including leading and trailing spaces – there should be none when finished). This program should not have any embedded loops, and you can only traverse the string ONCE. Should definitely be faster than $O(n^2)$.

Interview_202 : Write a program that multiplies 2 numbers (integers, floats, etc), without using the multiplication operator.

Interview_203 : The most difficult / unexpected question I was asked was how to optimize an algorithm for removing duplicates from a sorted list beyond $O(n)$. It turns out that the solution was still in $O(n)$, but involved parallelism to improve run-time.

Interview_204 : Binary tree algorithms

Interview_205 : Given a histogram find the rectangle with the maximum area.

Interview_206 : What problems will you point out in any products of Microsoft if you're going to improve it?

Microsoft interview-SDET

Interview_207 : Given a range of numbers ordered from 1 to 1 million, what is the most efficient method to find a number that is missing within the range.

Interview_208 : Q2. How you tackle the problem from this project?

Interview_209 : Write a kernel module which tests a home page and handle multiple requests from the server?

Interview_210 : If you had a conflict with a manager about how something would be implemented, what would you do?

Interview_211 : Find the temperature outside a room without going outside the room. No use of technology Either.

Interview_212 : Given a keyboard, how would you test it?

Interview_213 : How you reached the interview location?

Interview_214 : At the final step I was asked about a logic quiz. Moving 5 people from one through a bridge with only 1 flash-light. The people are moving in different phases.

Interview_215 : In what ways might you test a spoon for durability, consistency and usability?

Interview_216 : Since this is only the first interview, everything was easy

Interview_217 : During lunch, one interviewer asked me how I would test a pencil to prepare it for release.

Interview_218 : Company X has created a child's toy. The toy is battery powered with a speaker and motor system to drive a song and dance that it performs. How do you test that this toy is ready for production sales.

Interview_219 : None

Interview_220 : Implementation of strtok

Interview_221 : Given a triangle, determine if its a scalene, equilateral, isosceles or neither... required knowledge of triangle properties, I learnt these properties about two decades ago so ofcourse I was fuzzy on the details, completely unexpected

Interview_222 : First interview question involved 3-D vectors and normalization. Was difficult because I had forgotten these concepts (I was a math major, but hadn't applied the knowledge in years).

Interview_223 : Given a running sequence of numbers and u know that one number occurs more than 50% of the times, find that number. Do it in $O(1)$ space

Interview_224 : Are there anyone you know working in Microsoft?

Interview_225 : Asked about my experiences in test and using test techniques and tools in my projects.

Interview_226 : They asks from Data structure algorithms to logic puzzles.

Interview_227 : Write a test to validate whether a tree is a BST or not and write every case tested.

Interview_228 : Algorithm.....

Microsoft interview-SDET

Interview_229 : None..basic programming questions

Interview_230 : Sorting and Selecting from an array.

Interview_231 : List the different types of testing (and apply as many as you can).

Interview_232 : Given a linked list and a number k, find the k-th node from the end and return it.

An example:

List: A-B-C-D-E-F K

K: 2

Return F:

Interview_233 : the most difficult question was when they asked me about the most difficult question

Interview_234 : Finding if a stream of bits is divisible by 3

Interview_235 : How would you build something and test it?

Interview_236 : Find longest common substring.

Interview_237 : Be careful when you use `str.Split(' ');` because it may generate empty strings.

Interview_238 : Design and implement a "inner join" function using 2 sorted lists/arrays. Make it run in $O(n)$ time.

Interview_239 : Half the questions were unexpected because I didn't expect to be interviewed for an SDET position.

Interview_240 : Standard technical questions if you do your homework (literally and figuratively) you should be fine.

Interview_241 : How to Solve Supply Chain Issues

Interview_242 : Problem solving question: You have a list of numbers from 1 to 1000000 and one number is missing, but you don't know what it is. If you can only go through the numbers once, how do you find the missing number.

Interview_243 : How would you find a book in a library given the author and title, but no catalog.

Interview_244 : How would you call fire brigade if your office catches fire? Consider you don't have internet, telephone, or any means of communication.

Interview_245 : Q. Tree balancing questions

Q. design a notepad

Q. merge two arrays with sorting etc.

Interview_246 : Design a remote control.

Interview_247 : What are your weaknesses?

Interview_248 : How would you implement a shuffling algorithm

Microsoft interview-SDET

Interview_249 : i havent had a great experience as i was rejected in first round itself

Interview_250 : none

Interview_251 : How would you know the temperature outside without stepping out the room or using any technology

Interview_252 : You can place weights on both side of weighing balance and you need to measure all weights between 1 and 1000. For example if you have weights 1 and 3,now you can measure 1,3 and 4 like earlier case, and also you can measure 2,by placing 3 on one side and 1 on the side which contain the substance to be weighed. So question again is how many minimum weights and of what denominations you need to measure all weights from 1kg to 1000kg.

Interview_253 : Lots of data algorithms - make sure you know this stuff

Interview_254 : He asked me how to reverse a string in C. I was a bit rusty, and the answer he was looking for was to iterate over the length of the string store one character in a temporary character, then store the end of the array in the new location.

```
char<-str[i]
```

```
str[i++]<-str[end]
```

```
str[end--]<-char
```

Something like that in pseudocode.

Interview_255 : Given an integer write a function that converts the input into a linkedList where each node corresponds to a number of the integer. Eg: 25697 == 2 -> 5 -> 6 -> 9 -> 7

Then write a function that takes 2 linkedList, add the corresponding integers and return a third list with the result.

Interview_256 : Given a list of integers, return the longest contiguous subsequence that adds up to the largest value.

Interview_257 : I found it surprisingly easy overall.

Interview_258 : Given a histogram, design an algorithm and write code to find the largest rectangle that can fit within the bars.

Interview_259 : Find the loop in a linked list whose length is not specified

Interview_260 : Q: Write test cases of a notebook

Interview_261 : Given m memory blocks. memory filling rate is a blocks per unit time. memory emptying rate is b blocks per unit time.

Question: Related to rate and memory

Interview_262 : String related question about reversing a sentence.

Interview_263 : Questions are mostly about data base or data structure

Microsoft interview-SDET

Interview_264 : All the questions asked in the interview where coding questions as I had applied for a SDE position.

Interview_265 : implement sqrt

Interview_266 : How would you test a keyboard for design, durability and functionality

Interview_267 : Open ended question on determining outside temperature from a room.

Interview_268 : Require best algorithm both in time and space

Interview_269 : Reverse a string of words.

Interview_270 : You are given an analog clock. Calculate the minimum degree angle between the hour hand and minute hand for any time.

Interview_271 : Create a copy of a singly-linked list. Each node has a value and a next pointer, but also has a pointer to a random node: It can be null, or point to any other node in the list. The solution must take $O(n)$ time and use $O(1)$ space.

Interview_272 : 2nd Interview round was very difficult. I was asked to code a question. I coded in recursively and then I was asked to do it in iterative way. The question was new and there was not even a bit of help from interviewer.

Interview_273 : I've heard rumors about the weird questions they could potentially ask you. Maybe I was prepared because I didn't find any questions difficult or unexpected. Most of the questions were technical in nature. How would I do this and that; draw certain algorithms on a white board. etc

Interview_274 : N/A

Interview_275 : Interface vs Abstract Class?

Interview_276 : Nothing too difficult about the technical questions, I came from a public school in CA near the start of my JR year, and by the end of the year every type of question that was asked was covered in class. They don't look for the absolute best answer, as long as it's not overly complicated.

Interview_277 : all was easy

Interview_278 : If you have 9 balls. and one of the ball is lighter. All others are equal in weight. So then, if you have a single scale and you are given two turns how would you find the lighter ball?

Interview_279 : You have an image and you have to find the average of the image i.e. average of all the pixels into one pixel. The pixel has 3 values, R, G and B. Each R, G and B is a 8 bit number.

Write the code for it in C.

Interview_280 : I was asked not to disclose the questions so I cant but it was along the lines of reimplementing a given code where the code output a numerical series to make your code out put the same series.

Interview_281 : What do you want to do with your career? Why do you like learning? How would you build a calculator?

Interview_282 : Expected question. I was prepared for the interview and I was able to answer the question.

Microsoft interview-SDET

Interview_283 : There are different currencies being used all over the world. In Japan, it is Yen, in America, its \$. America has coins in denominations of 1, 5, 10, 25 and 50. Say if the input is 36, then $(1+10+25) = 36$. So 36 can be got by suing 1 one coin, 1 ten coin and 1 twenty five coin. So for getting 36, I need to use three coins. Write a pgm for that in all currencies of the world.

Interview_284 : Coding questions can be really difficult and time consuming related mostly to data structures. The MCQs are mostly related to Math and and logical reasoning

Interview_285 : Implement a stack that has a pop and push operations.

Interview_286 : How would you test certain feature.

Interview_287 : The most difficult question was, they gave me a puzzle and asked me to write an algorithm to solve it

Interview_288 : I was asked about the most challenging project in my past experience. I talked about one project I accomplished, but I supplemented (which I really shouldn't have done so) an unsuccessful research project which I didn't put on the resume. That project was too hard for me to finish in a few months, so I eventually abandoned it. It is not uncommon that a research-oriented project fails in the end, but the interviewer might think you are a loser.

Interview_289 : Questions regarding Algorithms used for development.

Interview_290 : Design Hibernate framework. Hibernate is a ORM layer

Design log4j framework. log4j is logging framework.

Interview_291 : I prepared for Data structures in Fifth round but encountered an ordinary question(Design for calender and write a code using oops tp print it) in oops . As i was expecting DS, i couldnt answer clearly .

Interview_292 : given a linked list which has two types of pointers, a normal next pointer which points to next element in the list and random pointer which points to random element in the list. Question was to clone this linked list

Interview_293 : What are you passionate about?

Interview_294 : A Maze is given as $N*N$ binary matrix of blocks where source block is the upper left most block i.e., `maze[0][0]` and destination block is lower rightmost block i.e., `maze[N-1][N-1]`. A rat starts from source and has to reach destination. The rat can move only in two directions: forward and down.

In the maze matrix, 0 means the block is dead end and 1 means the block can be used in the path from source to destination. Note that this is a simple version of the typical Maze problem. For example, a more complex version can be that the rat can move in 4 directions and a more complex version can be with limited number of moves.

Interview_295 : no diffcult question.. all they want to test is your knowledge and grasp.. the questions i was asked were:

1. given a data structure .one node with 2 pointers.. determine whether it is a doubly linked list , tree or nothing
2. design an atoi function

3. implement stack using 2 queues

4. tell me about your project

5. basic os questions

Interview_296 : Write a function that puts 2 unordered arrays into a 3rd order array

Interview_297 : How to write a string class

Interview_298 : How to test a file IO save solution (example was the save/load for excel)

Interview_299 : How to write the string to float function atof()

Interview_300 : Stack to queue

Interview_301 : All questions were average just be careful not to make mistakes.

Interview_302 : Why should we hire you?

Interview_303 : While taking coffee on the lobby from the automatic coffee maker one interviewer asked me How to test a coffee machine ?

Interview_304 : Most questions are about data structure, algorithms. so these days it seems like many interviewers don't ask unexpected questions.

Interview_305 : Write an class which support all functionality of Outlook client.

Interview_306 : Given an array filled with 'n' random numbers, each number may or may not be repeated again in the array, (mix of duplicates and unique numbers) shift all non-duplicates to the start of the array. for example, if array is {4,2,17,2,56,2,4} output should be {4,2,17,56...} the remaining part of array can be modified to anything, doesnt matter

Interview_307 : there is no difficult interview quesitons

Interview_308 : Find all words the input numbers on a phone may represent

Interview_309 : What was the most difficult problem you solved?

Interview_310 : first question was on addition of two very larg number then he ask which data structure i will use and he asked me to write the code . then he asked about different testcase possible

Interview_311 : in final 10 min he give me problem to find median of two shorted arrays

Interview_312 : They always ask you about possible test scenario of arbitrary algorithms of systems or even anything. You might want to practice some about this subject. Also, the reading suggestions of theirs are all good recommendations.

Interview_313 : reverse a linked-list

Interview_314 : how can a particular application be tested apart from testing its functionality

Interview_315 : Write a function that finds all the times a clocks hours and minutes hand are at 90 degrees.

Interview_316 : Write a function that reverses the letters in each word using only one char buffer. E.g.: "I work at Microsoft" to "I krow ta tfosorciM"

Microsoft interview-SDET

Interview_317 : Discuss how a queue works and how to add and remove items from the queue.

Interview_318 : How would you test a vending machine

Interview_319 : Sorting problem. About array

Interview_320 : What is polymorphism?

Interview_321 : What is a stack?

Interview_322 : How would you sort an array of 5+ random chars?

Interview_323 : How would you test a stack of variable Max size?

Interview_324 : In a sequence of alphabets (like aaabbd daabbcc) write a program to find the number of the consecutive alphabets in and print the alphabet and number.

example :if input is aaabbd daabbcc then output should be 3a, 2b, 2d, 2a, 2b, 2c

Interview_325 : Write a program to find out in a sorted array the sum of any two numbers present in the array is closest to the a number given.

if you have an array 4, 6, 8,24,36 and the given number is 31 then output should be $24 + 6 = 30$

Interview_326 : In place, move the duplicates in an array to the end. [Hint: first define duplicate. Ex. 02155530. Move 55 to end, OR 55 and 55, resulting in moving 555 to end.]

Interview_327 : Reverse a string

Interview_328 : How would you test Bing Translate?

Interview_329 : How would you test this feature of Excel?

Interview_330 : Test a feature in a windows product (product depends on whatever division you're applying for)

Interview_331 : Write a test code to test the program you just wrote.

Interview_332 : compare the value of two meaningful strings;

Interview_333 : insertion operation on binary trees with some restrictions.

Interview_334 : how to test a certain electric device.

Interview_335 : convert strings into int types;

Interview_336 : searching for patterns in strings; how to destroy your functions with different testing cases;

Interview_337 : How could you solve the subset sum problem in minimal asymptotic run time?

Interview_338 : How do u get the exact performance when we playing with a dies, if we throw it up and it get backs to our hand like that in a program. This is the question, where we spend somuch time and at any point i was not clear about the question but not sure what is the intension of the interviewer.

Interview_339 : LinkList questions, weighted binary tree

Microsoft interview-SDET

Interview_340 : How many gas stations are there in the United States?

Interview_341 : Given the root of a binary search tree, link all the nodes at the same level, by using an additional Node* level.

Interview_342 : How to test a calculator?

Interview_343 : how to find if two rectangles intersect

Interview_344 : lowest common ancestor in binary tree,

Interview_345 : given modules and shud come up with test cases

Interview_346 : Write a function that would count the number of words in a sentence.

Interview_347 : Convert a (binary) tree to a flat list.

Interview_348 : Skype is having complains by its customers. People are complaining that they do not see the list of all their friends. How would you solve this problem?

Interview_349 : Write a program that would arrange birth dates in an ascending order.

Interview_350 : Under NDA, can't say. Know your different test cases and brush up on coding (Strings, arrays, linked lists etc.)

Interview_351 : How would you test a map?

Interview_352 : How would you test the find function on notepad?

Interview_353 : WAP to check if strings are anagrams or not.

Interview_354 : guess the temperature outside a conference room without going outside

Interview_355 : check for the durability and funtionality of a keyborad

Interview_356 : How can you write a recursive function calculating the exponential of a number?

Interview_357 : A,B,C Input, determine if its a triangle.

Interview_358 : Find max sum of sequential integers in an array

Interview_359 : Come up with a design for 3 elevator

Interview_360 : Implement a double ended stack

Interview_361 : Count no. of words in a string

Interview_362 : count no. of each character & word in a file

Interview_363 : Why chose CS?

Interview_364 : If you were given a product, how would you enhance it?

Interview_365 : write a sub string in a string

Interview_366 : Find the heaviest ball among 9 other similar balls using a libra

Interview_367 : Give an example of a problem you encountered at * and how you were able to overcome it.

Microsoft interview-SDET

Interview_368 : Shuffle Linked List

Interview_369 : Delete duplicate characters

Interview_370 : Use 3 types of different data structure to delete a list of objects

Interview_371 : Reverse the string

Interview_372 : find duplicated item from an array, output the duplicated item with their times

Interview_373 : given three integers, determine whether they can form a triangle?

example 1: 3, 4, 5 --> yes

example 1: 3, 4, 15 --> no

Interview_374 : Check if tic-tac-toe has a winner

Interview_375 : Delete n-th element from linked list. Note, you don't have a pointer to the Head of the list.

Interview_376 : Sorting Algorithms Optimizations from 2 loops to just 1 loop

Interview_377 : Deleting Nodes & Reversing Strings

Interview_378 : How can u measure the temperature outside when you stay inside?

Interview_379 : The most unexpected question was about how I would spread the work between my team mates and myself.

Interview_380 : #1 design an elevator in your strongest OK language

#2 Assembly problem on a the board using JMP

#3 how would you test a random number generator?

#4 read code and optimize it

Interview_381 : Questions on Data structures and algorithms, code the answers and write test cases for the same.

Interview_382 : Lots of IQ questions like why are manholes round?

Interview_383 : Implement a Linked List in (Java, C#, C++)

Interview_384 : List all times (from 0:00 to 11:59) for which the angle between the hour and the minute hand form right angles.

Interview_385 : Reverse the individual words in a string (words are delimited by spaces). Example - "Hello world" becomes "olleH dlrow".

Interview_386 : Lunch interview discussed a bit about my work. Rest of the interview, as mentioned above under interview process.

Interview_387 : What is a stored procedure ?

Microsoft interview-SDET

Interview_388 : All of the questions are fairly easy. Be prepared for communication skills and big oh analysis.

Interview_389 : Reserve link listed pair-wise

Interview_390 : How to test a clock

Interview_391 : Relatively simple questions based on arrays and trees.

Interview_392 : How to test a calculator?

Interview_393 : - Given an array of integers (of any value, from MIN_INT to MAX_INT), how would you find the consecutive integers with the largest sum, and return their sum?

Interview_394 : - Given a string, how would you determine if that string contains a palindrome?

Interview_395 : - How would you search a string for a substring, in C? This guy also asked me some questions about hardware and security, such as "describe how a buffer overflow attack works".

Interview_396 : - How would you count the NUMBER of palindromes in a string?

Interview_397 : How many lines of code I wrote for the project I worked on?

Interview_398 : Write code to tell if two rectangles are intersecting.

Interview_399 : What are some possible causes of deadlocks?

Interview_400 : How would you test the Start button on windows

Interview_401 : Explain Linked List with an example and a code

Interview_402 : Test an alarm clock that automatically syncs the correct time from a tower owned by the government.

Interview_403 : why do you want to work for Microsoft

Interview_404 : Why do you want to be a SDE

Interview_405 : Reverse a string in a matrix in place. you are given the direction and length of the string.

Interview_406 : What sort of errors have you found before in Microsoft software?

Interview_407 : The questions were easy on my understanding. If i have to rank the questions, then the most difficult among them was, to write a program to generate Roman numerals from the decimal numbers.

Interview_408 : Correct all syntax errors and identify boundary conditions in this code.

Interview_409 : In the first interview, the coding problem was to generate a well-known data set. I first considered how to generate the nth iteration of the dataset, but she quickly steered me to solving iterations 1-n (which is much easier).

Interview_410 : Data structures and Algorithms related

Interview_411 : your data centers in japan have been hit by the recent earthquake.As chief infrastructure engineer,what steps would you take to minimize loss of data?

Microsoft interview-SDET

Interview_412 : Find an element in the array in $O(\log n)$. Array is circular, shifted by k units, and sorted (before shift).

Interview_413 : What should we test the computer mouse against to ensure quality and fitting to users' needs ?

Interview_414 : Tell me something about neural networks ??

Interview_415 : How do you write test cases for a software application that prints out triangles? What all do you consider when writing your tests?

Interview_416 : Name a product (doesn't have to be Microsoft's product) that you use frequently.

Interview_417 : If you had to add a feature to that product (asked in Question # 1), what would it be?

Interview_418 : If I give you a can of coke / pepsi (unopened), how would you test it?

Interview_419 : How to test an elevator?

Interview_420 : Why do you want to work at MS?

Interview_421 : Why Microsoft?

Interview_422 : backtrack problem, the max or min sum of a matrix.

Interview_423 : tree problem. reverse a tree (left to right)

Interview_424 : Enqueue and Dequeue

Interview_425 : How to find the different part of two array of integers.

Interview_426 : how do you test a remote control?

Interview_427 : Derive the formula for a line

Interview_428 : What is that thing on the desk...turned out to be a binary clock..... What do all the lights mean...

Interview_429 : Pretty standard questions. The first interview is designed to be easy (fib. sequence) and throughout the day, they'll get more complicated, mostly involving one or more data structures.

Interview_430 : Write a simple code which traverses a binary tree

Interview_431 : Given a matrix of $n \times n$ dimensions, write an algorithm such that each number (from 1 to n) appears only once in a column and a row. (Look up magic square for an example)

Interview_432 : Write a function that takes a String as a parameter. Count the number of A's in the String before you reach a C.

Interview_433 : Find a query pattern in a string

Interview_434 : Test a vending machine

Interview_435 : how will test a coffee vending machine ?

Interview_436 : What constitutes a good code

Interview_437 : How would you test an elevator built in a hospital

Microsoft interview-SDET

Interview_438 : In a BST write a program to find 2 nodes x and y such that $X+y=k$

Interview_439 : Consider two arrays of integers, v1 and v2, with allocate memory of $(n+m)*\text{sizeof}(\text{int})$ and $m*\text{sizeof}(\text{int})$, respectively;

In array v1, you have the first n positions filled with integers, sorted in ascendent order.

In array v2, you have all m positions filled with integers, also sorted in ascendent order.

Write/implement a function that return a vector of size $(n+m)*\text{sizeof}(\text{int})$ that have all the elements of v1 and v2, sorted in an ascendent order

Interview_440 : How do you reverse a linked list?

Interview_441 : reverse words in a string

Interview_442 : what's your most difficult decision?

Interview_443 : find two elements from an array summing up to a certain value

Interview_444 : How do you design a good product

Interview_445 : design a secure and usable login system

Interview_446 : implement factorial method

Interview_447 : Given a set of numbers -50 to 50, find all pairs that add up to a certain sum that is passed in.

What's the O notation for what you just wrote?

Can you make it faster? Can you find an $O(n)$ solution?

Implement the $O(n)$ solution

Interview_448 : find if string1 exists in string2, give test conditions

Interview_449 : How would you test a calculator that is going to be used by elementary school aged children?

Interview_450 : Implement enqueue and dequeue using stacks.

Interview_451 : Given an unsorted array of integers, sort them in such a way that all negatives come first, followed by zeroes, followed by all positives.

5,-6,8,0,4,-10,11

could become

-6,-10,0,8,4,11 (does not have to be sorted into increasing order)

Microsoft interview-SDET

What is the run time of this? Can we reduce the run time, if so, how (implement the new way)?
Doing black box testing, how can we determine the run time? How can we determine if we have a memory leak?

Interview_452 : Given a string (understood to be a sentence), reverse the order of the words.

"Hello world" becomes "world Hello"

Interview_453 : Design an elevator that is going to be used in a hospital

Interview_454 : array of +ve nums, find pairs that add up to a given value

Interview_455 : test a function that inputs 3 sides of a triangle

Interview_456 : How would you test a pen?

Interview_457 : check for palindrome

Interview_458 : How to improve performance of a web server?

Interview_459 : Given an 2 dimensional array of pixels on a monitor, how to convert that into a single dimensional array & what would then be the input to control a pixel?

Interview_460 : Find all the words that you can identify associated to a telephone number.

Interview_461 : Reverse words in a String

Interview_462 : Write the data structure for dictionary

Interview_463 : given two sorted linked list, merge them into one linked list.

Interview_464 : Convert a Hex value into IP address.

Interview_465 : Given a set of structure/object which has 2 value. Sort the set with respect to 1'st number and then with 2nd number

Interview_466 : Wrote a function on white board and asked me to test the function

Interview_467 : Implement a graph class, find the minimum spanning tree?

Interview_468 : 1st round:

write `integertoasciiconversion(char *s , int n)` , here the discussion on who would allocate memory pointed by `s` , if its the user what wuld be the adv ?

Interview_469 : 2nd round: test wireless mouse, given two string remove the occurances of characters in one string from another

Interview_470 : 3rd round: lunch interview, discussion of my resume, intersted projects, 2 more coding questions , sub sum of numbers in an array

Interview_471 : 4 round: given 2 rect's find the intersection here the discussion is about the choosing of structures, design

Interview_472 : 5th round: HR with manager, justify why SDET(this where i scrwed and got declined) , one puzzle about LED

Microsoft interview-SDET

Interview_473 : Write a program to print array in spiral order .

Interview_474 : Write a program to find anagrams of a string

Interview_475 : technical/ computer science related questions. complex NP-Complete algorithms.

Interview_476 : c# coding question on whiteboard. find first duplicate char in string but not in a foreach loop

Interview_477 : What precautions will u take to transfer money from one state to another

Interview_478 : Write a function that will calculate the angle between the the hour and minute hands of a clock given the time. Also, as I was interviewing for a test position they asked me to test the function after it was written

Interview_479 : find common ancestor in binary tree

Interview_480 : remove repeated letter in paragraph

Interview_481 : sort array

Interview_482 : How do you test a pen?

Interview_483 : string manipulate in text file.

Interview_484 : test questions

Interview_485 : merge two sorted arrays or lists

Interview_486 : level by level print BST

Interview_487 : Write a program to find out the median of two sorted array.

Interview_488 : Find if given tree is binary or not.

Interview_489 : Manipualte a string to place even char in leftt n odd in the right

Interview_490 : How do you test a keyboard

Interview_491 : What's your understanding of software test?

Interview_492 : Count # of A's in a sentence

Interview_493 : Write a program in java that determines whether a string is a palindrome.

Interview_494 : Write a method (in a language that you are comfortable with) that determines whether one given string is a substring of another given string.

Interview_495 : Given an array of integers in random order from 1 to 100, but one number was missing, name 3 ways you can find out which number was missing.

Interview_496 : Write a program that acts as an application scheduler that runs executables at given times.

Interview_497 : Print the binary tree in zig-zag order.

Interview_498 : How would you test a bottle of water

Interview_499 : Given a series of numbers, replace each number with its nearest possible nuumber

Microsoft interview-SDET

Interview_500 : Ordinary questions. Since the position is SDET, I did have some questions on testing. (like how to test a door, an application)

Interview_501 : What is the difference between a process and a thread?

Interview_502 : Print the order of numbers in reverse in an array using pointers. c-program

Interview_503 : reverse an int

Interview_504 : find number of anagrams of a given word in the given text file, also print them.

Interview_505 : According to you what is a good code?

Interview_506 : Why Microsoft and not some company like Honeywell?

Interview_507 : Describe alpha beta pruning.

Interview_508 : Given an airplane with 50 rows and 3 rows on each seat and the weights of all 150 passengers. How will you place the passengers so that the torque is minimized around the middle. Like balancing a seesaw.

Interview_509 : 1)Line of code in C++

Interview_510 : 2)Test a keyboard for a command

Interview_511 : 3)Design a keyboard for a child

Interview_512 : 4)Design a alarm clock for a child

Interview_513 : Reverse a list

Interview_514 : Tell me your most interesting technical contribution and why?

Interview_515 : Design and implement Run Length Encoding data compression system.

Interview_516 : Wasn't there any difficult question really but the interview covers a lot of aspect, between technical and non technical question.

Interview_517 : Why do you want to work for Microsoft?

Interview_518 : Write code in your favorite programming language that will accept two strings and return true if they are anagrams.

Interview_519 : I was asked to write some code on paper. The question was to create a function that took in two rectangles and to return true if they overlap and false if they don't. I was then asked to flip the paper and to create 10 cases to test my code.

Interview_520 : `int Solve(string str);`

Given the signature above, implement the simplest method that returns the results of the following arguments:

1. "3+5" (8)
2. "10+2-8" (4)

3. "5+10*3" (35)

4. "3+5--6" (14)

Interview_521 : How would you test a function that determines if 3 integers represent the sides of a triangle

Interview_522 : Abstract technical question: Interviewer draws a blender on the whiteboard and asks how I would test it. We talked about this for a good 25-30 minutes afterwards.

Interview_523 : what was the toughest challenge you faced so far while developing software?

how did you deal with that?

Interview_524 : Binary tree Program, Test Case for calculator

Interview_525 : What's engineering?

Interview_526 : How to do a test for the code?

Interview_527 : How is your feeling about doing test for other people?

Interview_528 : Write a method to find if 2 strings are anagrams.

Write code to give all possible phone numbers given the digits in the number. This is basically translating something like 1800-myinterview to actual numbers.

Interview_529 : Implement a function that turns a string into a number with sign.

Interview_530 : Implement a queue using only the stack data structure

Interview_531 : How to reverse the words in a string.

Interview_532 : What are the various means by which you can generate a fibonacci series?

Interview_533 : The differences between recursive and iterative implementations?

Interview_534 : How do you know if three points make a triangle?

Interview_535 : Write a function that takes another function as an argument and then defines a third function which calls the first and second function recursively.

Interview_536 : Design a B+ Tree

Interview_537 : The most unexpected question had to do with designing a messaging system, and involved some concepts I wasn't familiar with. This was my last interview and the interviewer was very friendly and patient, and guided me throughout the process.

Interview_538 : Walking the binary tree. I wasn't expecting this question and should have brushed up on my trees a bit more.

Interview_539 : write a function to return the attribute value if input is in the form of attribute="value". write test cases

Interview_540 : write function to remove the repeating characters from a given string

Interview_541 : Write a service that aggregates data from a web service, xml, and sql

Microsoft interview-SDET

Interview_542 : Trim a URL address by parsing it and if a "/../" string is obtained remove previous folder and print the final URL.

Interview_543 : Find two numbers that sum to zero in a list.

Interview_544 : How many cars in Chicago?

Interview_545 : In a given sorted array of integers remove all the duplicates.

Interview_546 : reverse a string in place

Interview_547 : Check if a given Sudoku is solved.

Interview_548 : None - all standard CS.

Interview_549 : Given an array of integers and sum, check whether any combination adds upto the sum. For solve then program in any language.

Interview_550 : What was the biggest challenge in your career so far and how did you overcome it?

Interview_551 : (without letting you read the job description again) tell me how well you fit the requirements of this position?

Interview_552 : What is your favorite product?

Interview_553 : how can you improve it? Which do you hate?

Interview_554 : How does the virtual keyword function internally?

Interview_555 : Phone Interview-

1. Behavioral questions based on resume.
2. Test a software that would create triangle based on length entered.
3. Palindrome using recursion.
4. Test alarm clock

Interview_556 : 1. Implement stack using queue

2. Merge two arrays and sort them in order as specified at runtime. The two arrays may share common entries between them, but the resultant array must not have duplicates.

Interview_557 : Implement strstr in the language of your choice.

Interview_558 : Give the pseudocode for a function that returns the n-th value of a Fibonacci sequence. Give both the iterative and recursive solution and their run times.

Interview_559 : If you are a recruiter and had a resume with no contact information, what are all the ways you can think of to contact the candidate?

Interview_560 : How would I optimize memory copy routine for different hardware platforms

Interview_561 : Don't recall...most questions were reasonable if you prepare well

Interview_562 : Ask you to perform some testing on a particular feature in one of their products.

Interview_563 : What low level features of Windows 7 are interesting?

Microsoft interview-SDET

Interview_564 : Design a circular queue

Interview_565 : Describe your testing procedure

Interview_566 : Generate all permutations of a string and determine if each is in the dictionary. Find an efficient data structure for the dictionary.

Interview_567 : Delete the Nth node from the end of a linked list.

Interview_568 : find the path in a binary tree from root to leaf, if the sum of nodes values equals a given number

Interview_569 : You are hired to make entries into a database. Your boss edits and entry but does not save it at the moment. You edit the same entry and save it. After this your boss goes and saves his edit thereby overwriting your entry.

What would you change in the database to avoid such a situation?

Interview_570 : Where will you be in the next ten years

Interview_571 : how to test onenotes

Interview_572 : How do you sort a linked list using the most efficient algorithm?

Interview_573 : Write a method (in Java) that would find the longest common substring in two strings

Interview_574 : Asked one question to solve: The "Ransom Note" problem.

Interview_575 : Print a tree one row at a time, zigzag the direction of printing the row as u go down the tree.

Interview_576 : how would you test the font button in microsoft word.

Interview_577 : how to merge two linked lists without using temp node

Interview_578 : given a monochrome display, light the pixel at (x,y) code

Interview_579 : Test a safety card

Interview_580 : Write an algorithm to pick the mayor of a town. There is a random number that is chosen and every nth person on the list is deleted until the mayor is chosen.

Interview_581 : Why do you choose to work as a tester

Interview_582 : Describe your most difficult team work experience

Interview_583 : How would you test Excel?

Interview_584 : How would you test a program that counts minivans on a highway

Interview_585 : Shortest path algo, BFS, Linklist as polynomial and operations on it

Interview_586 : How to test a Canoe

Interview_587 : Given N, find the Nth element from last in a linked list

Microsoft interview-SDET

Interview_588 : Implement itoa() in c.

Interview_589 : There is a long contact list in a mobile phone storing Name and other details. Your task is to give the data structure and algorithm for efficient searching among the contacts. As the user types the letters in the text field, the screen should show all the names beginning with those letters (of either first or last name or any other word in the name).

Interview_590 : none

Interview_591 : Evaluate postfix expression

Interview_592 : Given an int array and an int X, find out if two elements sum up to X

Interview_593 : Given an int array, find the sequence of elements that has the largest sum

Interview_594 : What do you do when there is no Answer?

Interview_595 : design an elevator

Interview_596 : Write a program to find palindromes

Interview_597 : find a recurring element in a linked list of unicode characters. If a unicode character/s is found to have a duplicate then just delete that repeating node and adjust the list. The constraint was to not to use any extra memory.

Interview_598 : using recursion to traverse a binary tree

Interview_599 : How is a context switch handled by the operating system?

Interview_600 : Writing code on the white board

Interview_601 : Write the kernel of the OS of a stopwatch

Interview_602 : Write a function that determines whether a given number is a prime. Discuss perf improvements.

Interview_603 : Write a function that takes a linked list and removes every other element in it. Discuss test cases.

Interview_604 : Describe a bug which are proud of. Why are you proud of this bug?

Interview_605 : how to find all the palindromes inside a string?

Interview_606 : why the lids of the gutter are circular?

Interview_607 : What is the structured class program in ASP.NET

Interview_608 : which ms product do you use a lot (hotmail). how would you test it? what would you change about it?

Interview_609 : Explain virtual table concept

Interview_610 : Given a file that has a C++ program in it, how would you verify that the code in it is syntactically correct?

Interview_611 : Write a function to turn a string into an integer and test it

Microsoft interview-SDET

Interview_612 : How do you create and reserve a linked list? After you have written the function to create the functions, write a function to sort the linked list.

Interview_613 : Given a sequence

AEF

BCD

Complete it.

Interview_614 : Given a numeric expression, evaluate the expression and give the result.

For instance:

$4-3*12+300$

Interview_615 : How to detect loops on a single linked list?

Interview_616 : I was asked to write an algorithm in C++ to sort an array. The function I was to write was to take in a string of letters "b" and "r". b = a red ball and r = a blue ball. The function was to return an integer with the least number of swaps needed to separate the balls to be next to their own colors...

So for example, RBRR would return 1, since you only need to swap one of the letters to be RRBB or BBRR.

Interview_617 : How would you add two linked lists which represent a number, they have one of the digits among 0-9 in there data field

Interview_618 : Remove a node from Linked list when address of same node is given

Interview_619 : Design Memory Management System

Interview_620 : I was asked to write a method which takes an integer as an input parameter and returned a string as the output variable. The string's value was to be the written format of the integer. For example, if the input was, "1,550,225" the output was, "One million, five hundred and fifty thousand, two hundred and twenty five"

Interview_621 : how do you test a stapler?

Interview_622 : you have 100 doors in a row that are all initially closed. you make 100 passes by the doors starting with the first door every time. the first time through you visit every door and toggle the door (if the door is closed, you open it, if its open, you close it). the second time you only visit every 2nd door (door #2, #4, #6). the third time, every 3rd door (door #3, #6, #9), etc, until you only visit the 100th door.

Interview_623 : How to rotate strings in a multi-dimensional matrix.

Interview_624 : Linked list questions

Interview_625 : sorting question

Microsoft interview-SDET

Interview_626 : One of the difficult question was about solving a puzzle with good algorithm and coding it up as well. I don't recall the puzzle's name now, but the solution involved using linked list (circular), and writing some classes and methods. In the end, testing was also required for this solution.

Interview_627 : Code a function in C to get the largest consecutive addition of integer numbers from an array.

Interview_628 : How would you test a toothpaste package product?

Interview_629 : Design and test a system to figure out if a shape is closed or not

Interview_630 : How many gas stations are there in California?

Interview_631 : How would you test (a physical object in the room or something else)? Totally unexpected when asked to tell them how you would proceed to test a spoon. This is to test your testing aptitude.

Interview_632 : None of the questions stood out as most difficult or unexpected, but after a few hours the last interview was the hardest just from fatigue and being short on sleep from the traveling.

Interview_633 : Maximum common Subsequence

Interview_634 : How would you test a vending machine ?

Interview_635 : BFS

Interview_636 : Reverse a link list. Try to do it recursively as well.

Interview_637 : Write a procedure to shuffle cards, and also mention strengths and shortcomings of your approach.

Interview_638 : Given a document and a set of words keywords. Find the shortest text from the document that contains all the keywords

Interview_639 : how would you move mount fuji?

Interview_640 : The most difficult questions were the technical ones. They consisted of programming questions to "how would you test this" questions.

Interview_641 : How many ballons would fit in the room you are currently in?