

Qualcomm Interview Experience | Off-Campus 2020

- Difficulty Level : [Medium](#)
- Last Updated : 25 Feb, 2021

I applied through the Qualcomm careers website(workday portal). My resume was shortlisted for the role of RF embedded software engineer and I received a call from one of the team members.

Telephonic Round: After the formal introduction of myself, I walked him through my resume. I was questioned on pointers, structures, unions, data structures, and projects in the resume. Basics of electronics were also asked as I'm from an electronics background. It was a quick call of 30 mins. At the end of the call, I was informed that an interview will be scheduled at the convenience of the interviewers.

A full-day online interview(on Microsoft Teams platform) comprising 4 technical rounds was scheduled.

Round 1:

A brief explanation of the job role & responsibilities.

Code snippets were given and outputs should be determined. Codes were related to

- pointer referencing, dereferencing & incrementing
- size & memory allocations of structure & unions
- memset function
- constant pointer and pointer to a constant.
- executing volatile variables in functions

Questions other than code snippets were

- Structure padding & packing
- How are variables of a union stored? If the union contains an integer(4 bytes) and a single character(1 byte). Will the character points to the MSB or LSB? The question focuses on how the variables of the union are pointed in the memory block allotted to the union variable. [Concept of little-endian and big-endian systems]
- Implement your own **sizeof** function [Typecasting pointers]
- Write a program to find if the stack is increasing upwards or downwards. [Create two local variables. Local variables are stored on a stack.]
- What is the volatile keyword? How does a volatile variable change?
- There are two 32 bit HW timer registers `TIMER_L`, `TIMER_H`, which form a 64-bit timer(`TIMER_L` lower 32 bit, `TIMER_H` higher 32 bit). To design an API to determine a 64-bit timer, how many register reads are required at best case and worst case?

Round 2:

Questions related to process management & memory management concepts in operating systems.

- Suppose a program is running on 5 threads. If an extra thread should be added, what factors should be considered? [hardware related and thread management related requirements]
- Will the efficiency of the system increase for sure if add an extra thread. If yes/no, why?
- What is a segmentation fault & How to avoid it?

- What are memory leaks & how to recover from memory leakage in a program? [We can't recover within a program. Care should be taken to avoid memory leakage, i.e, by freeing unused variables.]
- If you are to allocate the memory block to a different variable, how will you do it? [allocation, deallocation, reallocation of the main memory. An exact answer was not required. The interviewer wants to know how I would come up with a solution.]
- If you are writing a program to allocate memory to variables, how will you avoid memory corruptions due to the allocation of pre-allocated memory?
- Suppose in a normal program if a variable or a memory location is corrupted how to identify what portion is corrupted?

Round 3:

- Double pointers
- Write code snippet for dynamic memory allocation to 1-dimensional, 2-dimensional arrays.
- You are given a sorted array A of m integers and another array B of the size of (m+n) and initially filled with n elements in sorted order. How will you insert the elements of A in B and finally make the array B sorted? [Time complexity : $O(m+n)$, space complexity : $O(1)$]
- Find the K-th node from the end in a linked list.
- A modified version of Floyd's cycle detection algorithm. Is it possible to find the loop in a linked list if the fast pointer is moved three steps ahead instead of moving 2 steps? If yes or no, why?
- **Puzzle:** 10 machines are manufacturing balls. The first machine produces balls of weight 1 unit, the second machine produces balls of weight 2 units, and so on incrementally up to the tenth machine which produces balls of weight 10 units. One out of the 10 machines fails and starts producing faulty balls of different sizes which differs x units from the initial weight. Given an electronic balance that can measure the weight of any number of balls in a single instance how many times should the balance be used to find out the faulty machine?
- Write a function that takes the root of a binary tree as an argument, creates a mirror of the binary tree, and returns the root of the mirrored tree.

Round 4:

- What is a RTOS (real-time operating system)? Where are they used?
- How is a RTOS different from an ordinary OS?
- Is Android or windows an RTOS. If no, why?
- What are function pointers & why do we need them as we can call a function from any other function?
- Inline function vs macro. Why do we need inline when we can use a macro? Exclude the scenarios where inline will be executed as a normal function and explain.
- Why is debugging faster when the inline function is used?
- If you are given a choice to choose between binary semaphore & mutex what will you choose and why?
- Why do we need a union if memory is not a constraint?
- What makes Interrupt Service Routine (ISR) different from a normal function?
- How is the value of volatile variables altered?
- Few casual questions like how was the interaction with the interviewers till that round, career goals, interest in higher education, etc were also asked.

Few more questions related to RTOS were asked.

I received a telephonic call from HR after 6 days. Just a casual talk regarding job role, work location, package, and all.

My Personal Notes

Add your personal notes here

Save