[Date]

By: Tarun Bharani

The Four P for Placement

**//Index//**

1. Interview questions on C programming. 3-7
2. Interview questions on Operating system. 8-10
3. Interview questions on cpp. 11-12
4. Interview questions on Data Structures. 13
5. Interview questions on Real Time Operating System 14
6. Interview questions on Microcontroller. 15-17
7. Interview questions on Internet of Things. 18
8. Interview questions on Linux Device Drivers. 19
9. Interview questions on miscellaneous Topics. 20-22

**INTERVIEW QUESTIONS ON**

**//C LANGUAGE//**

1. Storage classes. Explain each with example.
2. Function setting, clearing, toggling nth bit.
3. Function for counting set bits.
4. Array vs pointer
5. Structure vs union
6. Setting alternate bits.
7. Logic to separate even and odd bits
8. Zero Padding in structure.
9. String functions
10. Dangling pointer
11. Memory map (do from Ashok Pathak test 3 solution 1)
12. Complex scenarios of pointers like size of pointers, finding size from address of pointers, function pointers, array of function pointers
13. String related program (WAP to implement these without using strcpy, strlen, strcmp) Storage class, where each type of variable is stored
14. Storage class, where each type of variable is stored
15. Data types size -+, typedef, preprocessor directives, keywords like extern, static, volatile, Global static, local static.
16. Steps for c file compilation, what are the intermediate files generated.
17. Void pointers and their application
18. Macros and Inline function and argument passing in macros
19. Volatile memory
20. Bitwise operator (**Highly focussed topic in interview coding-Nxp, maxLinear, Honeywell, FEV, Intel**…)
21. Odd even using bitwise operator
22. Pattern Printing, Famous programs like palindrome, armstrong no, using recursion etc.
23. Static vs Global variable and their significance?
24. In which segment array, character is stored?
25. Difference between structures and classes?
26. Difference between extern and inline?
27. Difference between array and linked list?
28. Difference between C and embedded C?
29. 5 main stages in embedded C?
30. Dynamic memory allocation? Difference between malloc and calloc?
31. C program for a recursive function?
32. C program for reversing a string?
33. Difference between C and C++
34. Difference between C and python
35. Programming for swapping two variables without using third variable
36. Fibonacci series, Power of 2 series, Print reverse string/Array/number
37. Swap two number (with 3rd variable/ without third variable) use only operator addition sub possibility/multiplication div possibility
38. WAP to swap two structures.
39. Difference between macros and inline function.
40. Difference between normal function and inline function.
41. Macros vs Constants
42. memcpy vs strcpy?
43. Setting on and clearing a pin using bitwise
44. Function pointers and its registration (and where it is used)
45. How interrupt vector table can be created using c program
46. <https://www.geeksforgeeks.org/dangling-void-null-wild-pointers/>
47. <https://www.geeksforgeeks.org/swap-two-numbers-without-using-temporary-variable/>
48. <https://www.edureka.co/blog/swapping-of-two-numbers-in-c/>

**2022**

(Note-> C++ language becomes mandatory; importance is equal to C language. Practice on Hacker rank and Hacker Earth. Must do problem)

1. Write structure with 3 Element name roll no and marks, write two functions
2. One to display its data and other to increment its marks.
3. Pass pointer to structure variable while passing argument to function
4. Pani- Puri problem, Floor management Problem using oops concept (c++) and c, both.
5. Swapping bits at time -> like swapping the set bits with the reset bits and printing the result in decimal no.
6. Reversing the string and searching the string and replacing it with the first char of the string.
7. Freeing thee the memory without using free function. (ans>realloc(ptr,0));
8. Bottle filling problem, using the C and (C++) oops concept.
9. How negative numbers are stored.
10. Use of Linked List
11. How #define works
12. Can you do 'a'='b' in c.
13. What is lvalue and rvalue
14. What is Sizeof()
15. What is the difference between struct and union. What is padding in structure.
16. Make a user defined function for strchr()
17. Storage classes in C?
18. Difference between static and global
19. Can you redeclare a static variable x inside a function again
20. Compilation flow of C?
21. Detail about storage classes.
22. sizeof(), starlen functions.
23. Padding and Packing. How 👉🏻 #define MIN (A, B) ((A)<=(B)? (A):(B)) will work in compilation flow
24. Sizeof(Void), ans->=1. (In UNIX & LINUX)
25. Designing a fountain Problem’s solution. (pls Google it)
26. Passing Void Pointers as an argument to function pointers, program (eg -> int (\*fun) ( int, void \*ptr))--Write a program.
27. Scenarios in which a program can crash.
28. Write a few statements to intentionally crash the program.
29. If a code is being stuck at any point, how will you identify the point at which it is stuck?
30. Dangling pointer and dynamic memory allocation.
31. Write declaration of a function pointer. What are function pointers and why/where do we use them?
32. Game logic:

A king is trying to capture a kingdom. Provide a kingdom as a matrix of MxN, each state is equal in size and square. The territories guarded by Knights are at the position provided by an array. If the Knight is ith position then king will get caught if he tries to capture a state present in ith column and ith row. Write a program to find the maximum number of states king can capture without getting caught.

1. All Storage classes in C with examples, How does extern work,
2. Difference between Static and global, Auto keyword Volatile keyword, why and when is it used
3. Register Variable, Use case of it
4. Reverse a String in C with recursion
5. Write a prog to set ‘n’ consecutive bits from a given position ‘k’ in a number ‘b’
6. Write a macro to set a bit
7. Write a program to count number of set bits in a number
8. Write a program to shift all 1’s to left and all 0’s to right in a given binary number
9. What does Stack overflow error mean?
10. Explain how free(ptr) works, think about how you can implement it on your own.
11. Memory allocation in struct v/s union
12. Write a Function pointer declaration that takes 2 arguments and call it.
13. How is program counter affected during recursion
14. Write a MACRO to calculate no. Of seconds in a year?
15. String Palindrome, String Reversal, String copy [without String functions]
16. Storage classes: Implement Extern practically.
17. What is segmentation fault? How can u initialise this fault?
18. What is the difference between Embedded C and C?
19. What is the entry level function in C code? (main)
20. What is the significance of writing Void before main? What does it mean?
21. Why do we Write return 0; at the end of the program? What if we don't Write it?
22. How many arguments can main take? (2- int argc, char \*argv[ ])
23. What is argc and \*argv[ ] ?
24. What is the difference between various standards of C? Which standard you are currently using?
25. Scenarios in which a program can crash.
26. write a few statements to intentionally crash the program.
27. If a code is being stuck at any point, how will you identify the point at which it is stuck?
28. Error log files.
29. Dangling pointer and dynamic memory allocation.
30. Write declaration of a function pointer. What are function pointers and why/where do we use them?

**INTERVIEW QUESTIONS ON**

**Operating Systems**

**OS:**

1. Semaphore vs mutex
2. Binary semaphore vs mutex
3. Ipc mechanisms. System calls used in each.
4. Thread vs process vs task
5. Named unnamed semaphore use
6. Linux kernel module loading command and usage
7. Benchmark test for Linux based OS
8. Testing and Debugging of Linux OS
9. If you want to build a OS FROM sketch then, how you will do it?
10. IPC Mechanism in OS.
11. Cross compilation,native compilation.study well question can be ask based on scenarios.
12. 5-main function of OS?
13. About JTAG and real-time system and core debugging.
14. How shell script works..?
15. In laptop the OS is GPOS or Embedded OS?
16. Fork vs exec
17. Does bootloader laod the complete OS ?
18. Have you heard about Secure OS ?
19. Critical secton and semaphore mutex.?
20. Can we have a cryptographic module in OS ?
21. Why do u use fork(); ?
22. What is lseek() ?
23. Difference between Linux OS and OS?
24. What is an OS? [Definition should be more technical]
25. Is Linux an OS or a Kernel? Why Yes, Why No?

**INTERVIEW QUESTIONS ON**

**CPP**

**CPP:**

1. Polymorphism
2. Inheritance
3. Encapsulation
4. Abstraction
5. Virtual function
6. Friend
7. Constructors
8. Destructors
9. Class vs structure
10. C vs cpp
11. **Interview Questions :**
    1. Producer Consumer Problem
    2. Write A Program for creating thread its arguments and their synchronisation using Semaphore and Mutex
    3. Types of scheduler in RTOS, GPOS
    4. Semaphore
    5. Spinlock
    6. Mutex
    7. Shared Memory
    8. Difference b/w process and thread
    9. Data shared in same data segment or different in threads
    10. What is IPC?
    11. Why does segmentation fault occurs (how to debug it)
    12. Process v/s Thread v/s Task
    13. Basic Linux commands
    14. Difference between Semaphore and Mutex
    15. What is context switching
    16. Contents of PCB.

**INTERVIEW QUESTIONS ON**

**//Data Structures//**

* 1. Linked list insertion deletion from anywhere, size of link list and other operations as well.
  2. Implement push operation of stack using linklist?
  3. Queue & Circular queue
  4. Doubly linked list
  5. Write delete function of singly linked list of a given user data
  6. Difference between C and DS
  7. Reversing of single linked list
  8. Searching and Sorting at least one algo. Each
  9. Inorder, preorder, postorder traversal of tree.
  10. Insert a node in BST using preorder traversal.
  11. Reverse a singly linked list.
  12. Implement push operation of stack using linklist?
  13. Write a program to find the 3rd largest element from an array in O(n) complexity.
  14. Write an optimised program to find out if in a sorted array sum of two numbers equals a given number. (Hint: Make use of the fact that it's a sorted array) Evaluate its time complexity.
  15. Declare a singly and doubly linked list and Add a node.
  16. DS for singly and Doubly LL
  17. Doubly LL insertion - write function
  18. Traversing of LL - write function
  19. Delete node from particular node position - write function
  20. Stack - push posh operations
  21. Priority queue

**INTERVIEW QUESTIONS ON**

**//Real Time Operating System//**

1. RTOS v Task delay HAL delay
2. Hard Real Time OS
3. What is RTOS how differs from OS
4. Soft Real Time OS
5. Name some RTOS? Vxworks.
6. Types of scheduling policies in RTOS?
7. Multitasking Programming
8. Priority Inheritance
9. Priority Inversion, Ceiling
10. Apart from time-constraint what are features of RTOS
11. API for Task creation, Semaphore, Mutex.
12. Types of scheduling
13. Schedule the following tasks for a Car
14. Airbags, Display, Brakes.
15. Which scheduling will you use and why?
16. How will you schedule two task: Reads & process the Sensor data
17. Priority inversion and priority inheritance, with examples
18. Can you make a function call in ISR
19. What is Priority Ceiling protocol
20. How can you initialise init function without HAL Library?
21. Policy scheduler
22. Time frame between fcfs and rtos .
23. How are Time slicing in rtos?
24. Kernel vs user space.

**INTERVIEW QUESTIONS ON**

**//MICROCONTROLLER//**

1. CAN, SPI, I2C, UART bus in detail with pins, addressing a slave, diagram, baud rates (Highly Focussed)
2. Difference between Microprocessor and Microcontroller
3. Why STM32 is used?
4. What is ARM? Explain
5. Explain RISC and CISC
6. ISR-Interrupt service routine explain?
7. Explain watchdog timer?
8. Instruction pipeline explain?

3 stage:

F etch--->>Decode--->>execute

5 stage:

fetch--->>Decode--->>execute---->>Memory ---->>write back

1. Von Neuman vs Harvard Architecture
2. What is the use of 3V & 5V in electronics ckt?
3. various bit manipulation logics - normal set reset, masking, set reset a range of bits
4. Half duplex full duplex meaning
5. Gated clocking, effect of clock on power consumption
6. Flip flops, use of flip flops in micro cont.
7. What are the features of STM32 & Beaglebone black.?
8. What are the protocols supported by STM32. & Explain.
9. What are the steps required to blink LED in 8051.
10. Steps for generating hex file and why tool is required to generate hex file?
11. What is Prescaler?

*A* ***prescaler*** *is an electronic* [*counting circuit*](https://en.wikipedia.org/wiki/Counter_(digital)) *used to reduce a high* [*frequency*](https://en.wikipedia.org/wiki/Frequency) *electrical signal to a lower frequency by* [*integer division*](https://en.wikipedia.org/wiki/Integer_division)*. The prescaler takes the basic timer clock frequency (which may be the CPU clock frequency or may be some higher or lower frequency) and divides it by some value before feeding it to the timer, according to how the prescaler register(s) are configured.*

* 1. What happens when we power on a micro-cont. How hex file is burnt?
  2. Interrupt while interrupt is executing and how does stack changes in this situation
  3. Due to pipelining PC is one instruction ahead, justify and due to this what effect happens on stack execution
  4. I2C, SPI & CAN start and stop signal waveform setup.
  5. How will you measure 10v on ADC of a microcontroller which supports only 5v
  6. What is the difference between synchronous and asynchronous communication.
  7. Explain CAN Data Frame (detailed study)
  8. What is the use of resistor in CAN (remember the value->120ohm)
  9. Explain voltage levels in CAN.
  10. Explain the process of write a code on computer to burning it to flash of microcontroller.
  11. What are the points that should be considered during writing an ISR.
  12. Why is it not suggested to write a lengthy ISR.
  13. What is Program counter and Link Register in ARM
  14. What is the job of NVIC and how does it do that?
  15. Number of interrupts in ARM and priority levels.
  16. Difference between I2C and SPI. I2C and SPI in detail. Which is preferred when?
  17. Why ARM follows Thumb-2?
  18. ISR FLOW?
  19. SPI/I2C - Data frames + Clock cycle
  20. CAN - Data frame
  21. What is NVIC?
  22. How the processor came to know that certain task has been resumed after an interrupt occurred, how it will check? (By checking PC)?
  23. What is a breakpoint? Why do you use it?
  24. What do u mean by parsing?
  25. What is an arm-cc-compiler? It's significance?
  26. CAN protocol-->Explain working, framing, error detection, synchronization, arbitration mechanism, tranceiver, terminating resistance, TTL logic....
  27. How SPI data fetching works? Explain in depth with proper signalling diagram.
  28. Explain the working of I2C using proper diagram.

**INTERVIEW QUESTIONS ON**

**//Internet Of Things//**

1) Why AWS is used in projects?

2) About MQTT?

3) Is the use of MQTT a single process or many processes?

4) In IoT which protocol use most?

5) What is cloud? You know how and give example?

6) Which is the best cloud?

7) What is socket communication?

8) What the different cloud platforms.?

9) Difference between MQTT and HTTP and why MQTT is used??

10) Which IoT platform u choose for project and why?

11) ESP32 characteristics and it's PIN Diagram?

**INTERVIEW QUESTIONS ON**

**// LINUX DEVICE DRIVER//**

1. Definition-Char device drivers, Block device drivers (generally (512bytes= 1block(generally))– Asked In FEV India, Network Device drivers with example (be very careful in answering and using words in definition).
2. Write a program for Char Device Driver to load “Hello world Module”→ Asked in intel—> very simple (google it)
3. What is a Kernel Module?
4. How would you go about writing your own Low level character driver (File operations)

**INTERVIEW QUESTIONS ON**

**// Miscellaneous Topics//**

**BASIC ELECTRONICS:**

1) What is diode? operating voltage of diode.

Ans) A diode is *a two-terminal electronic component that conducts electricity primarily in one direction*.

**Silicon diodes have a forward voltage of approximately 0.7 volts.** **Germanium diodes have a forward voltage of approximately 0.3 volts**.

2) Basic logic gates and its truth table.

3) What is cmos and pmos? Circuit diagram

*A* ***complementary metal-oxide semiconductor*** *(CMOS) is the semiconductor technology used in most of today's integrated circuits (ICs), also known as chips or microchips. CMOS transistors are based on metal-oxide semiconductor field-effect transistor (MOSFET) technology.*

4) What is photodiode? how it works??

5) No. of pins reduced in the DRAM (CAS and RAS pins) (google it)

**DIGITAL COMMUNICATION:**

1. Aliasing effect.
2. Sampling Theorem.
3. Find Nyquist interval if Sampling frequency (for ex: 20kHz)
4. What is Mixer (frequency)
5. What is the communication system? How is the message signal sent?. From transmission to reception, Explain?
6. If I send a bits stream (eg. 010010101001 ...something like this), then how do I know if I received the correct bits stream at the receiver end?
7. What is parity check, explain even and odd parity?
8. What is hamming code?

**PROJECT AND OTHER TOPICS**

1. Asked about all three projects ie. CDAC, College Major, College Minor with proper explanations with proper circuit diagram.
2. Asked some questions on MATLAB related to my project on college.
3. What is LDR and explain its working with the help of circuit diagram?
4. Explain the bootloader and how it works? Explain the flow of bootloading.
5. How to debug kernel if it is not booted successfully.(explain the procedure in the terms of hardware as well as software perspective)
6. Different b/w insmod and modprobe
7. If you system hangs how to find which process responsible for it.
8. Explain the booting procedure of kernel in the 100 core cpu.
9. What is difference b/w timer and watchdog timer.
10. What is Rush in cloud explain.
11. What is trust in ARM.
12. Suppose I given you one uc board but it is not working properly. What will be your debug procedure for that board.
13. Networking - 1.TCPIP 2.OSI layers.