Adv C Module

1. MCQ

1)
   ```
   Int a[]={1,2,3,4,5};
   Int *p;
   p=a;
   printf(“%d\n” ,*p);
   printf(“%d\n” ,*++p);
   printf(“%d\n” ,*(p+1));
   p=&a[2];
   *p=-10;
   printf(“%d” ,a[2]);
   ```
   What is the output for this cod?

2. Subjective

2.1 Basic Refreshers

1. Explain data types and their sizes.

2.
   ```
   int c[32];         // Array declaration
   int i;              // Loop variable (not initialized)
    for(i = 0; i < sizeof(c); i++)  // Loop
   {
       c[i] = 0;                   // Assignment
   }
   ```
   This piece of C code is supposed to initialize every element of the array c to 0. Does it perform this task correctly
3. How do you shift a number by a certain position?
4. What is the difference between a++ and a=a+1? Which one will execute fast?
5. What is the purpose of using void main() in a program?
6. Explain data types in detail
7. What is C?
8. What is the difference between int and float
9. Explain what is meant by an array and how to define an array.

2.2 1D Pointers and Functions

1. What is a Segmentation fault? Explain with an example.
2. What is the difference between pass by value and pass by reference?
3. Explain pointers with an example
4. Explain the NULL pointer
5. Explain what is meant by function, with advantages and an example program.
6. What is a function?

2.3 String

1. Explain the strcat in detail.

2.5 Storage classes and memory segments

1. Explain Storage Classes in detail.
2. Explain the Static and Volatile keywords with examples.
3. Explain the Compilation stages in detail.
4. Explain the Static function with examples.

2.6 2D Pointers and DMA

1. What is a Memory leak? Explain with an example.
2. How will you free the memory without using free()?
3. Explain the function pointers with examples.
4. Explain the dynamic memory allocation.
5. Explain DMA with examples.

2.7 Preprocessing

1. Explain the compilation stages with relevant commands.

2.8 UDT

1. Explain the differences between Structure and Union with examples.
2. Explain the bitfields.
3. What is structure? Explain the examples

2.9 Miscellaneous

1. Explain the volatile keyword
2. Explain the difference between const and volatile.

3. Programming

1. WAP to reverse the hexadecimal value. Ex: i/p = 0x72 o/p = 0x4E.
2. WAP to swap the nibbles.
3. WAP to reverse each word in a given string.
4. WAP to convert a hex value to binary.
5. WAP to find the 3rd largest number in a given array.
6. WAP to find HCF and LCM using recursion.
7. Write a C program to swap the left 8 bits and the right 8 bits of a 16-bit unsigned integer.
8. WAP to swap the two numbers by using pointers.
9. WAP to change the number from little endian to big endian
10. WAP to reverse a number using bitwise operators?
11. Write a program to swap the bytes of the given hexadecimal number
    ex:
        I/p = 0xABCD
        O/p = 0xDCBA
12. WAP to swap the two numbers without a third variable.

MC Module

1. Basic electronics

1. What code is executed in the microcontroller before the execution of the main function?
2. Explain the different software methods to reset an MC.
3. The microcontroller stores all the memory in RAM itself. Why do we have registers?
4. List the differences between a microcontroller & microprocessor
5. How pull-up and pull-down work.
6. Explain the EEPROM.
7. How many pins are there in pic18f4580?
8. What is the use of the Watchdog timer?
9. Explain the volatile keyword.

2. Basics

1. What is a voltage conversion device?
2. What is a MOSFET? How can it be switched, and what are its terminals? Is the gate terminal similar to a regular logic gate?

3. Interrupts

1. Explain the interruption in detail.
2. How much is the clock frequency in 1 second?
3. List the differences between polling and interrupting. List the advantages and disadvantages of using interrupts instead of polling.
4. Explain how the timer is working. How to calculate 1 second using an 8-bit timer at 20MHz frequency.
5. What is going to be the count of the timer for generating a delay of 10 milliseconds in a controller running at a speed of 4 MHz.

4. Projects

1. Explain car black box project in detail
2. Explain the pick-to-light project in detail.
3. Explain the microwave oven project in detail. Ask for the code to start the microwave, then display "Microwave Started" on the CLCD and prompt the user to set the time. Then, write the program to set the time.

5. ADC

1. Explain the ADC in detail.

6. Embedded Systems

1. Explain about an embedded system.

7. Protocols

1. Explain the UART protocol with a frame format. Write a simple code.
2. Explain the SPI protocol with a frame format.
3. Explain the I2C protocol with a frame format.
4. Explain what is meant by the baud rate and where it is used.

5. Why should we communicate at a very high baud rate?
6. Describe 2 to 3 debugging steps if the UART protocols are not working.
7. What is the difference between UART and USART? What are the disadvantages of UART communication?
8. What is the difference between SPI and I^{2}C protocols?

8. PWM

1. Explain how the PWM is working.

CPP Module

1. How will you create a user-defined template in C++? Give an example.
2. Explain the Pillars of OOPS.
3. What is STL in C++? Why do we use it?
4. What are the main components of STL?
5. Name a few commonly used STL containers.
6. What is the difference between a vector, a list, and a deque?
7. How does a map differ from an unordered_map?
8. What is the difference between a set and a multiset?
9. What is the difference between a struct and a class in C++?
10. What is the difference between new and malloc?
11. What are access specifiers in C++? Name them.
12. What is function overloading? Give an example

Ds Module

1. Basics

1. Explain linear and non-linear data structures.

2. Linked lists

1. Explain the types of linked lists

2. WAP to reverse an SLL.
3. WAP to insert a node at the end in SLL
4. WAP to remove duplicates in SLL.
5. WAP to delete the nth last element in an SLL.
6. What is the difference between algorithms and data structures?

3. Stack

1. What is a Stack? Explain with an example
2. Explain the complete code on the stack implementation and applications
3. WAP to demonstrate push and pop operations in the stack.

4. Searching and Sorting Techniques

1. WAP to sort the array by using the bubble sort method
2. WAP to sort the given array by using merge sort
3. Explain quick sort algorithms.
4. Explain searching algorithms.

5. Queue

1. What is a Queue? Explain with an Example.
2. WAP to Implement a Queue using an array
3. What is a circular queue? WAP to implement a Circular queue.

6. Hashing

1. What is a hash table, and what is its use? Give an example of a hash table briefly about with an example. What is the time complexity of the hash program?

7. Trees

1. WAP to print the pre-order output of a given BST
2. WAP to find the height of a tree.
3. WAP to print in-order output of a given BST.

8. Project

1. Explain the arbitrary precision calculator project.
2. Explain the inverted search project.

LI Module

1. Basics

1. Explain about kernel.
2. Explain the use of OS in detail.

2. System call

1. Explain the difference between a Library call and a system call.

3. Networking

1. Explain about TCP and UDP.
2. Which is reliable between TCP and UDP?
3. What are the TCP and UDP protocols? What is the difference between the TCP and UDP protocols?

4. Process

1. Explain how to create the process.
2. Explain the use of the exec system and its variation in detail with examples.

5. IPC

1. What is a pipe?
2. What is shared memory?
3. Why? (with real-time Example) IPC mechanism brief Explanation (PIPE, FIFO, SHM).
4. Explain the IPC mechanism.

6. Signal

1. Explain how to handle the SIGKILL signal.
2. Explain when the SIGCHLD signal will occur.

7. Socket

1. Explain socket programming.

8. Threads

1. What is a semaphore?
2. Race condition? Critical section Implementation with mutex method and Semaphore (Using real-time example)

3. When we are using the mutex method and when we are using the semaphore. (with real-time example)
4. Instead of a mutex, if you use a semaphore? Instead of a semaphore, if you use a mutex, what happens? Explain in detail.
5. What is the difference between a binary mutex and a mutex with an example?
6. What is the difference between thread and pthread?
7. Explain about process and thread. Can we create threads or processes in Kernel Space?
8. What is the difference between a mutex and a semaphore?
9. What is the difference between a process and a thread?

9.Process and Memory Management

1. Explain the Process and memory management