**DAY 7**

**1.Write a program to convert a data stream from Little Endian to Big Endian (you can take any 8 bit value or use 0b110101110**)?

Algorithm:

Step1: Start

Step2: Declaring the list of variable

Step3: To converting little endian to big endian

Step4: Using swapping converting little endian to big endian, print swap value.

Step5: Stop.

**2.  Write a code to swap even bits with odd bits in unsigned long variable(uint64).**

Algorithm:

Step1: Start

Step2: Declaring the variable as unsigned int

Step3: Converting decimal to binary value

Step4: Swap even bit to odd bit, and to right shift the even bit and left shift the odd bit

Step5: After the shifting of bits used OR operator to return the value.

Step6: Stop.

**3.Write program to clear (0) bits CIS, CM2, CM1, CM0 in register “CMCON” if bits CHS3, CHS2, CHS1, CHS0 bits are set to 0x07 in register “ADCON0”**

Algorithm:

Step1: Start

Step2: Declaring the variable.

Step3: Initialize register two register

Step4: To clear a bit from given register and to set bit from 2nd register from 0th bit to 7th bit.

Step5: convert the two registers bits in Hex format.

Step6: Stop.

**DAY 8**

**1.Realize a function which has 2 arguments and 1 return value. Argument 1 is Byte value, Argument 2 is bit position.**

**Function has to flip the bit position (as per argument 2) in argument 1 byte value and return complete byte value.**

Algorithm:

Step1: Start

Step2: Declaring two variable names as argument 1, argument 2.

Step3: If {condition as arg1 equal to first arg1 used XOR condition, if 1bit left shift by arg2.}

Step4: And the condition satisfied go to main print the A value.

Step5: Stop.

**2.Write a program in C to sort an array using Pointer**.

Algorithm:

Step1: Start

Step2: Initialization array value, using pointer to access the array address.

Step 3: for array index start from zero and pointer I and will be incremented.

Step4: if (pointer i > pointer j) and the swapping started.

Step5: finally go to print statement and print pointer i.

Step6: Stop.

**3. Set the register T0CON value such a way that bit TMR0ON and PSA bit are set (1), without disturbing the other bits.**

Algorithm:

Step1: Start

Step2: Initialization two position as pos1, pos2

Step 3: Using conditional operator n value is equal to n and position of the bit have to left shifted one bit after the condition of given is satisfied goes next condition.

Step4: print the n.

Step5: Stop

**II). Find out the value of T0PS2, T0PS1, T0PS0 bits in T0CON register if the value of the register is 0xF3**

Algorithm:

Step1: Start

Step2: Declaring variable name

Step3: for i is equal to 2, i greater than are equal to the zero it goes next condition.

Step4: one bit right shifted from i condition is true

Step5: print the x.

Step6: stop.

**DAY 9**

**1. I). Set the register SSPSTAT value such a way that bit CKE, P and S bit are cleared (0), without disturbing the other bits.**

Algorithm:

Step1: Start

Step2: Initialization two different register as a and b.

Step3: Using XOR operator compare the register a and b.

Step4: print the c.

Step 5: stop.

**II). Find out the value of UA, BF, SMP bits in SSPSTAT register if the value of the register is 0x55**

Algorithm:

Step1: Start

Step2: Declaring four variable.

Step3: To find the binary of hex value.

Step4: check the condition set 1 to 0th bit, set 1 to 1st bit, set 1 to 7th bit and print the value after bit set.

Step5: print the b, c, d.

Step 6: stop.

**2. Write program to set (1) bits CIS, C2OUT, C1OUT in register “CMCON” if bits CHS3, CHS2, CHS1 bits are set to 0x06 in register “ADCON0”**

Algorithm:

Step1: Start

Step2: Initialization and declaring the variable.

Step3: enter the value

Step4: scan the value

Step5: if condition of the given variable is equal to number to go next statement.

Step6: position left shift 1 position

Step7: if the condition is false print the else part.

Step8: stop.

**3. Assume that we have a car with 8 seats, and we are monitoring 8 seat belt status in a variable “G\_Msg\_switchstatus\_Byte[]” as shown in pic1. Each switch status is combination of 2 bits as shown in pic2.**

Algorithm:

Step1: Start

Step2: Initialization and declaring the variable.

Step3: enter the fault

Step4: scan the number.

Step5: using switch case to find case 1, case 2, case 3, case 4 condition.

Step6: stop.

**DAY 10**

**1. Write a function to store "G\_Eng\_EngineTemperature\_uchar" and "G\_Eng\_EngineRpm\_uint" into " G\_Msg\_EngineInformation\_Byte[5u]" as shown below.**

Algorithm:

Step1: Start

Step2: Initialization and declaring the variable name.

Step3: Using loop check the condition every request.

Step4: And every condition is satisfied it will print the Number.

Step5: stop.

**2.Write a function to store "G\_Str\_SteeringAngle\_sint" to "G\_Msg\_SteeringInformation\_Byte" as shown below.**

Algorithm:

Step1: Start

Step2: Initialization and declaring unsigned char, signed variable name

Step3: Using loop check the condition

Step4: Each variable can check the condition and it go to print statement

Step5: stop.

**3.Write a program to Implement ”Selection Sort” using Recursion for given elements (Number elements will vary, so make it as configurable item)**

Algorithm:

Step1: Start

Step2: Declaring array index value

Step3: Using loop condition to compare each bit from the hole array .

Step4: From the given value small to big number will be printing using selection sort

Step5: stop.

**DAY 11**

**1.Write a program to do binary search in a sorted array. Program should take input of sorted values and search value.**

Algorithm:

Step1: Start

Step2: Declaring and initialize var name  
Step3: In the given value to find the middle value from given array of element.  
Step4: Middle value equal to add the low index and high index to divide by 2.  
Step5: After the getting the middle value to search the value.  
Step6: if the value is present then print the statement, else condition not equal print the statement.

Step 7: Stop.

**DAY 12**

**1.Find common elements in three sorted arrays, 3 array values are given as input to program**

Algorithm:

Step1: Start

Step2: Initialize 3 array element  
Step3: array of size and array of each bit to find the common element to print.

Step4: Using the loop condition if the satisfied then go to next steps to increment the bits.

Step5: if the condition is false to print the else part.

Step6: Stop

2.Find empty strings in an array of strings (Array of strings will be given as input to program) and print the position of array where empty string is present. And replace the empty string array element as ‘CYIENT’.

Algorithm:

Step1: Start

Step2: Declaring variable and initialize char   
Step3: Using loop condition

Step4: if the condition is satisfied print the empty string.

Step5: found empty string that will be replaced with the given string.

Step6: Stop