

Day 11

- ① Write a program to do binary search in a sorted array. Program should take input of sorted values and search value. (make number of elements as configurable value). How binary search works is explained below.

- ① Accept the sorted data.
- ② Accept search value.
- ③ Middle value is equal to addition of start and end index and divided by 2.
- ④ If x is greater, ignore left half and if x is smaller, ignore right half.
- ⑤ If condition is false repeatedly check from the second point until the value is found.

Day 12

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

- Initialize the arrays.
- Use If condition.
- If $i < j$, then increment i .
- If $j < k$, then increment j .
- If $i = j = k$, then print that equal value otherwise it will check next equality.

- Initializing the main method.
- If common value are found it prints that values otherwise it will print no common values found.
- Stop.

③ 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 "CLIENT".

→ start

→ Initialize the strings with empty strings.

→ size will be stored in one variable.

→ Use for loop to find the empty strings.

→ when you found empty string that will be replaced with the given strings.

→ print all the strings along with replaced string.

→ stop.

Day 8

④ Realize a function which has 2 arguments and 1 return value. Argument 1 is byte value, Argument 2 is bit position. Function has to shift the bit position (as per argument 2) in argument 1 byte value and return complete byte value.

→ start

→ Declaring the variables naming as a argument 1 and argument 2.

→ If Condition as arg1 equal to first arg1, i.e 1 bit left shift by argument 2) using XOR condition.

→ And the Condition satisfied go to main point of the A value.

→ stop.

⑤ Write a program in C to sort an array using pointers. The sample output must be as below.

- start
- Initializing the main method.
- Declaring the variable of arrays.
- And array starts from 0 and pointer starts from 1.
- pointer of variable is taking temporarily for array.
- Declaring the condition, if the condition is pointer [i] greater than pointer [j].
- And swapping starts.
- Then, print giving print statement and print pointer [i].
- stop.

⑥ Set the registers TCON value such a way that bit TMROON and PSA Bit are set(1) without disturbing the other bits.

- start
- Initializing the main method.
- Declaring the variable of (n) to derive the values.
- To change the position of bit without disturb the other bits.
- Then, change the particular bit of position to make (1).~~(0)~~
- Giving print statement and print n.
- stop.

① Find out the value of TOPS₂, TOPS₁,
TOPS₀ bits in TOCON register if the
value of the register is 0xF3.

→ start

→ Initializing the main method.

→ We are Declaring the variable of n,

→ Take 0xf3 to save the value
of n.

→ If the condition is true($i=2 ; i>=0 ; i--$).

→ Making x equal to n greater

than i.

→ Then, moving the position of bits.

→ statement, & print x.

→ Givens print

→ stop.

Day 9

② Set the register SEPSTAT value.

③ Set the register SEPSTAT value.

→ start

→ Initialize the main method.

→ Declare the variables a and b.

→ And giving the hex value of 0x58

for a and hex value of 0xFF

for b.

→ We can do (XOR) for 0xFF with

0x58 value.

→ Storing this values in C variable.

$$c = a \wedge b.$$

→ Givens print statement and print c.

→ stop.

⑦ Find out the value of VA, BF, SMP bits in CSPSTAT register if the value of the register is 0x55.

→ start

→ Initialize the main method.

→ Declare the variables a, b, c, d.

→ They given the value of 0x55 we are taken as a.

→ Then, Do left shift to move the position of bits for ~~b,c,d~~ b, c, d.

→ And, we can do (&) for b, c, d with the a.

→ And give print statement and point b, c, d values.

→ stop.

⑧ ② Write program to set C1 bits C1S, C2OUT, C1OUT in register "CMCON" if bit CHS3, CHS2, CHS1 bits are set to 0x06 in register 'ADCON0'.

→ start

→ Initialize the main method.

→ Declaring and Initializing the variables.

→ And point the values and scan the numbers of num1 and num2.

→ Giving If Condition, Both the variables are equal to the given numbers, it goes to next statements.

→ If the condition is true, print the statement, If the condition is false point the else part.

→ stop.

⑨ G-Meg - switch status :-

- start
- Initializing the main method.
- Initialization and declaring the variable.
- Enter the faulty types.
- Scan the number.
- Use switch cases to find case 1, Case 2, Case 3, Case 4 Condition.
- stop.

day 10

⑩ Write a function to store "G_Eng EngineTemperature ucha" and "G_Eng EngineRpmUnit" into G_Meg EngineInformation byte [suJ].

- start
- Initializing the main method.
- Declares the variable name.
- Use for loop to check the condition.
- And check the condition is satisfied or not.
- Then, the condition is satisfied it will print the number.
- stop.

⑪ Write a function to store "G-ctr-steering
Angle sint" to "G-meg-steering-information-Byt8".

→ start

→ Initialize the main method.

→ we are declaring the unsigned char,
and signed variable names.

→ Given signed int G-ctr-steeringAngle-sint=-60.
and unsigned char is G-meg-steeringInformation-ByteB[7]

→ Using loop to check the condition.

→ And every variable can check the
condition and it go to point statement.

→ stop.

⑫ Write a program to Implement "Selection Sort"
using Recursion for given elements (Number
elements will vary, so make it as
Configurable stem).

→ start

→ Initialize the main method.

→ Declare the array values.

→ Use loop condition.

→ It has compare each bit from
the hole array.

→ They are given value small to
big number will be printing
using selection sort.

→ stop.

Day 7

step 1:

Initializing the method called swap-Endians.

step 2:

Declaring the variables.

step 3:

Swapping the individual binary two digits.

step 4:

Return the value to the main function.

step 5:

Initializing main method.

step 6:

Declaring the variables.

step 7:

Calling the swap method for big endians.
and little endians.

step 8:

Print the result from swap endians.

step 9:

End.

- ① Write a program to convert
from little Indian
to BIG
a data
value
or use
stream
or use
 $0b110101110$?

2) Write a code to swap even bits with odd bits in unsigned long variable (const64).

step 1:-

Initializing the method swap bits.

step 2:-

Declare the unsigned integer variables.

step 3:-

And to converting decimal value into binary value.

step 4:-

Return the even bits, and odd bits to the main function.

step 5:-

Initializing the main method.

step 6:-

Declaring the unsigned integer variable with the value.

step 7:-

Print the return value from the swap bit method.

step 8:-

End.

③ Write program to clear bits CM1, CM2, CM3, CM4 in register "CMCON". If bits CHS₁, CHS₂, CHR₁, CHR₀ are set to 0x07 in register "ADCON0".

Step 1:

First to declare a datatype.

Step 2

And to initialize register name.

Step 3

To clear a bit (or) given register, and to get the bit of another register bit 0 to bit 4.

Step 4

And after the set a bit in register to change the position of bit.

Step 5

To convert into register bit in hex formate.

Step 6

Step (or) end.