Assignments Embedded Training

1. WACP to print ASCII value of all characters with equivalent characters
2. WACP to print size of all Data Types with diffrent qualifiers
3. WACP to print 1-10 excluding 7 & 9 using while loop
4. WACP to print factorial of a number using while loop
5. WACP to setbit (function) - set the bit at a given positon as 1 (use bitwise operator)
6. WACP to resetbit (function) - set the bit at a given positon as 0 (use bitwise operator)
7. WACP to togglebit (function) - toggle the bit at given positon (use bitwise operator)
8. WACP to readbit (function) - read a bit at a given position in binary (use bitwise operator)
9. WACP to displaybit (function) - display number in binary (use bitwise operator)
10. WACP to countbit (function) - count no of once in a number in binary (use bitwise operator)
11. WACP to setbit (macros) - set the bit at a given positon as 1 (use bitwise operator)
12. WACP to resetbit (macros) - set the bit at a given positon as 0 (use bitwise operator)
13. WACP to togglebit (macros) - toggle the bit at given positon (use bitwise operator)
14. WACP to readbit (macros) - read a bit at a given position in binary (use bitwise operator)
15. WACP to displaybit (macros) - display number in binary (use bitwise operator)
16. WACP to countbit (macros) - count no of once in a number in binary (use bitwise operator)
17. program to check whether the number is even or odd(with and without using arithmetic ops)
18. program to swap two numbers (without and without third variable)
19. program to check whether two numbers are equal (with and without == operator)
20. program to find the sum of digits of a number (with and without recursion)
21. program to reverse the digits of a number
22. program to find the sum of even and odd numbers
23. program to print a identity matrix in simplest logic
24. WACP to multiply 2 numbers without using * operator
25. WACP to find the greatest/smallest of 3 numbers
26. WACP to find the greatest/smallest of 3 numbers using ternary operator
27. WACP to find the area of a triangle, square, circle, volume of a sphere
28. WACP to find the consonants and vowels in a sentence.
29. WACP to find whether a number is a prime or not.
30. WACP to print all the factors of a number
31. WACP to print convert lower case characters to upper case and vice versa in a string.
32. WACP to print all visible ASCII values and ASCII characters using a C program.
33. WACP to copy one string to another.
34. WACP to reverse a string
35. WACP to concatenate one string to another
36. WACP to find the factorial of a number
37. WACP to find whether a number is strong number or not
38. WACP to find whether a number a amstrong number or not
39. WACP to find whether a processor is endian or not using typecast
40. WACP to use preprocessor constants in GCC

41. WACP to check whether a number is pallindrome or not.
42. WACP to check whether the entered year is a leap year or not .
43. WACP read a bit of a byte at a particular position.
44. WACP to set a bit of a byte at a particular position.
45. WACP to reset a bit of a byte at a particular position.
46. WACP to toggle a bit of a byte at a particular position.
47. WACP to perform all 4 operations using a macro.
48. WACP function to print string length with **array** of chars (Do not use string.h library)
49. WACP function to reverse a string with **array** of chars (Do not use string.h library)
50. WACP function to compare two string with **array** of chars (Do not use string.h library)
51. WACP function to convert string to lower case with **array** of chars (Do not use string.h library)
52. WACP function to convert string to UPPER case with **array** of chars (Do not use string.h library)
53. WACP function to toggle Upper case letters with lower case letters and vise versa with **array** of chars (Do not use string.h library)
54. WACP function to print string length with **array** of chars (Do not use string.h library)
55. WACP function to concatinate strings with **array** of chars (Do not use string.h library)
56. WACP function to copy string with **array** of chars (Do not use string.h library)
57. WACP function to print string length with **pointers** (Do not use string.h library)
58. WACP function to reverse a string with **pointers** (Do not use string.h library)
59. WACP function to compare two string with **pointers** (Do not use string.h library)
60. WACP function to convert string to lower case with **pointers** (Do not use string.h library)
61. WACP function to convert string to UPPER case with **pointers** (Do not use string.h library)
62. WACP function to toggle Upper case letters with lower case letters and vise versa with **pointers** (Do not use string.h library)
63. WACP function to print string length with **pointers** (Do not use string.h library)
64. WACP function to concatinate strings with **pointers** (Do not use string.h library)
65. WACP function to copy string with **pointers** (Do not use string.h library)
66. Implement pre-processor directives
    1. #define
    2. #ifdef
    3. #undef
    4. #if
    5. #else
    6. #elseif
    7. #endif
67. Use pre-defined macros
    1. *_FILE_*
    2. *_DATE_*
    3. *_TIME_*
    4. *_LINE_*
68. recursivly print a string pattern

```
1
1   2
1   2   3
1   2
1
```

69. WACP to preform basic file operaions
    1. read
    2. write
    3. append
70. WACP to read a file and return a count of
    1. Characters
    2. Words
    3. Numbers
    4. Spaces
    5. Next line
    6. Tabs
71. WACP to use contant pointers
72. WACP to use contant pointers to constant
73. WACP to print matrix
74. WACP to print identity matrix in simplest way
75. WACP to check diffrence b/w strlen and sizeof
76. WACP to use pointer as a parameter of a function
77. WACP to use function array
78. WACP to use function with help of pointers
79. WACP to use structure normally and find its size (padding)
80. WACP to use packed structure then find its size (packing)
81. WACP to implement array of structure
82. WACP to write a program to take a date from user and return yesterday's date and tommorow's date
    - (the `next_date(struct Date *curr_date)` should take a pointer to a Date Structure as an argument and put next date relative to the date in the `curr_date` structure.)

```
// The Date Structure
struct {
 int dd:5;
 int mm:4;
 int yy;
} Date;
```

83. WACP to take user input in a structure and store in a file and do the following operations
    1. Read
    2. Add date

    3. Delete
    4. Display
84. WACP to get factorial in 3 ways
    1. loop
    2. recurtion
    3. macros
85. Print diffrent data types using single pointer (pointer typecasting)
86. shallowCopy vs deepCopy
87. linked List
    1. Insert front
    2. Insert rear
    3. Delete front
    4. Delete rear
    5. Display
    6. Display reverse
    7. Insert at pos
    8. Delete at pos
    9. Display using recursion
    10. Display Reverse using recursion
    11. Make Linked List circular
    12. Display Circular Linked List
    13. Reverse the Linked List
    14. Remove Duplicates from sorted Linked List