

DATA STRUCTURES & ALGORITHMS 1

BATCH – B

[FRIDAY FEBRUARY 01, 2019: 3:30 PM – 6:30 PM]

LAB ASSIGNMENT – 3

CODE:assign03

NOTES:

1. Please carefully read all assignments and there is no choice.
 2. **Use the template for this assignment**
 3. Each problem in this assignment has to be answered in the same c file.
 4. Create a .c file following the file name convention:
 - a. If your roll number is 'abc' and assignment code is 'assignXX'. Then use the following file name convention as follows: 'abc-assignXX.c'
 - b. For example, if the roll number is 92 and assignment code is assign01, then the file name should be 092-assign01.c
 - c. Strictly follow the file name convention. When you are ready, submit the solution via google classroom.
 5. Follow naming conventions
 - a. except for variables in for-loop, none of the other variables should be a single character.
 - b. The variable names and function names should indicate what they are storing/computing.
-

PROBLEM INSTRUCTIONS:

For the following problems write functions which satisfy the following:

1. The functions **should not have a return statement** (hence its return type should be void).
2. **All the arguments** to the functions should be **pointers**
3. Do **not use global or static** variables.

PROBLEMS [Total Marks: 20]:

1. [Marks: 6] Write a function which takes in an empty array of size 30 and fills it with the first 30 values of the function $T(n) = T(n-1) + T(n-3)$ for $n > 2$. Where $T(0)=0$, $T(1)=1$, $T(2)=1$.
 - a. The function should only populate, you should print the result in main() after the function call.
2. [Marks: 6] In main, use dynamic allocation (malloc) to store 'n' characters (taken as input from the user) and write a function which takes as input
 - a. a pointer to the first memory location
 - b. Reverses the list of characters in place (i.e same memory location)

- c. The function should only reverse, print the result in main()
3. [Marks: 4] Write a simple function to swap ('int') values stored in two variables. The function should not use a auto-variable (allocated in the stack) as temp variables instead use malloc.
4. [Marks: 4] Create a Faculty struct. It should have 'name' and 'salary'. Write a function increment_salary, which takes as argument a faculty struct and increments the salary by 20%. In main() create an instance of 'facuilty' and pass it to this function.
-