## **Data Structures and Algorithms (2017)**

You are logged in as 111603043 Chetan Shriram Paralikar (Log out)

Search/Create Courses COEP Website COEP Wiki COEP FOSS Server COEP IT Handbook Learn Moodle
Change Password Preferences

Home ▶ Computer Engineering and IT ▶ DSA-2017 ▶ 25 September - 1 October ▶ (Assignment) Tree Functions



- Problem

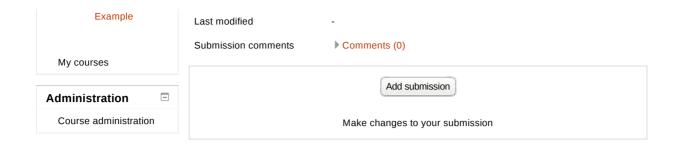
## (Assignment) Tree Functions

```
Using the code of the tree given on moodle for compilation,
Just add the following functions in a file named as 'extra.c' and submit here.
Your code will be compiled with the tree.c and try.c files. 'tree.c' is the same as
given under code for tree. The 'try.'c is attached here.
//[1 mark]
char *max(tree t) { /* returns pointer to malloced memory */
/* Searches the maximum value in the tree. Do not assume to be a binary search
tree. The code should work for any binary tree. You may think of writing a
recursive function. */
}
//[2 marks]
void preorder_norecur(tree t) {
/* Does a preorder traversal without using recursion. Hint: requires a stack: stack
of what? */
//[1 mark]
char *search(tree t, char *str) {
/* Searches the str in the tree and returns a pointer to the str , NULL if not
found*/
//[3 marks]
void printtree(tree t) {
/* prints the binary tree so that it looks like a tree. Hint: May use an already
known data type, which one? */
Hints: Finding the max is easy, as the code is very small. To write Non-recursive
preorder travresal, you can focus on "unfolding" the recursion which happens in
preorder traversal, to get a sense of sequence of function calls, and how
arguments are "remembered" and "remembered-back" in LIFO fashion as the
function calls proceed. You need to incorporate the same sequence of
remembering function-calls and their return sequence in your code.
For printing the tree - You need to to think of a way to figure out the exact
"position" of a node in the tree,i. .e. it's depth and it's sequential position at that
depth. The printing code, needs to take care of exact-positioning on the screen of
each element, by calculating spaces (for separation) for each element. This will
require some mathematical formulas to be devised.
```

## **Submission status**

Submission status	No attempt
Grading status	Not graded
Due date	Sunday, 15 October 2017, 9:00 PM
Time remaining	10 days 1 hour

1 of 2 10/5/17, 7:19 PM



You are logged in as 111603043 Chetan Shriram Paralikar (Log out)

DSA-2017

2 of 2 10/5/17, 7:19 PM