

# ASSIGNMENT1: Menu Driven Program Generation

## Using General Purpose Trees

DUE DATE: 11:59 PM , 17th September (Thursday night)

### Problem Statement:

Create a code generation program that will take as an input from stdin a file containing the specifications of a menu and output a C code to stdout.

The input would look like the following:

```
one
|
|   oneone
|   onetwo
|   |   onetwothree
|   onethree
two
|
|   aaa
|   |   xxx
|   |   |   zzz
|   |   |   |   yyy
|   bbb
|   ccc
three
|
|   abc
|   pqr
|   |   ppp
|   |   |   qqq
|   rrr
four
five
```

The levels are indented using a tab space (\t).

The output should be a single C file which when compiled and run, waits for input from stdin like in a menu driven program.

The input to this program now will be a sequence of integers.

0 means you would go back a level (going up a function call sequence). If we're at the outermost level, then 0 would exit the program.

So, if the input now is:

1  
1  
2  
1  
2  
0  
3  
0  
4  
0

The output should be:

one  
oneone  
onetwo  
onetwothree  
onethree  
four

### **A couple of things to note:-**

Every line will contain only one word.

While inside a menu, entering a number that's more than the range of possible inputs should not print anything. Notice how there's nothing printed after onetwothree even though I gave 2 as an input.

You cannot technically *enter* the levels of leaves. You can only print in those cases. Like oneone. Note the level we'd be in at the of the following two inputs:-

Input 1:-

1  
1  
0

We'd be at the outermost level (depth 0) accepting inputs for that level, that is 1-5.

Input 2:-

1  
2  
0

We'd be at the level inside 'one' (depth 1) accepting inputs for that level, that is 1-3.

You can assume the limit of each line to be max 1024 characters.

You are allowed to use string.h

**Files to be submitted:-**

A1\_<SRN>.c

A1\_readme\_<SRN>.c

Make sure you make your code readable with appropriate comments.

**How we'll be compiling and testing your code :-**

(Based on the experience from last time we're adding this section to help you submit a proper working version of your code)

All your code will be evaluated on an Ubuntu machine. Make sure you test your code on an ubuntu machine (take care of line endings, etc).

Suppose the SRN is PES1201701142, input file is menu.dat, and input sequence file is inputsequence.txt

```
gcc A1_PES1201701142.c -o codegen  
./codegen <menu.dat >outputprog.c
```

```
gcc outputprog.c -o outputprog  
./outputprog <inputsequence.txt >output.txt
```

You can ask doubts at: [Doubt Clarifications](#)

Submission link will be shared later.

**DO NOT COPY/SHARE CODE.**