

# Birla Institute of Technology and Science, Pilani

Department of Computer Science and Information Systems

Semester I (2020-21)

Principles of Programming Languages (CS F301)

Tutorial sheet (#5)

Date: October 9, 2020

---

1. Consider the following nested function definitions written in a C-like language. The return statement returns the execution control to the calling function.

```
function_1(){
    int x, y, z, u;
    x=20; y=35; z=40; u=23;
    z=x+y*2-z+u;
    printf("in function_1: %d %d %d %d\n", x,y,z, u);
    function_2(){
        int x=76;
        z=x*3-y+z+u;
        printf("in function_2: %d %d %d %d\n", x,y,z, u);
        return;
    }
    function_3(){
        int y, z;
        y=90; z=11;
        z=x+y-z*4+u;
        function_2();
        printf("in function_3: %d %d %d %d\n", x,y,z,u);
        return;
    }
    function_3();
    function_2();
    return;
}
main()
{
    function_1();
    return;
}
```

- (a) Draw the activation tree for the above code.
- (b) Draw the snapshots of the call stack displaying the growth at each function call and shrinking at the completion of each function call.
- (c) List out the variable name occurrence bindings to their declarations and identify appropriate locations for accessing the non-local data.
- (d) Compute the output of the above program, if the language supports static scoping. Identify the static parents first for each variable occurrence.
- (e) Compute the output of the above program, if the language supports dynamic scoping. Identify the dynamic parents first for each variable occurrence.