

The execution status includes everything needed to restore the execution of the caller.

Includes- registers, CPU status bits, Environment Pointer (EP)

The EP is used to access parameters and local variables during the execution of a program.

The linkage action for a subprogram occurs at the beginning of its execution is called the *prologue*.

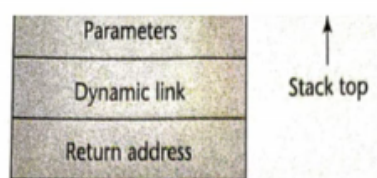
And the actions that occur at the end is called as *epilogue* of subprogram linkage.

Simple subprograms – actual code, and local variables and parameters.

The format or layout of the non-code part of a subprograms is called an *Activation Record* (AR), because the data it describes are relevant only during the execution of subprogram. The concrete example of AR is called *Activation Record Instance* (ARI).

For simple subprograms it is of fixed size. Hence can be statically allocated.

Activation record for simple subprograms.



Implementing subprograms with stack-dynamic variables

The stack dynamic variables can support recursion. Linkage of subprograms with stack dynamic variables is complex when compared to simple subprograms, because:

1. Compiler must generate code to cause implicit allocation and deallocation of local variables.
2. Recursion adds the possibility of multiple simultaneous activations of the same subprogram, means that there can be more than one instance of a subprogram at a given time. The number of activations is limited by memory size of the machine. Each activation requires its activation record instance.

functional value \rightarrow value returned by the callee to caller

```
void sub(float total, int part) {  
    int list[5];  
    float sum;  
    ...  
}
```

Can be done @ compile time

to base \leftarrow ptn

Local	sum
Local	list [4]
Local	list [3]
Local	list [2]
Local	list [1]
Local	list [0]
Parameter	part
Parameter	total
Dynamic link	
Return address	

Activation record for sub()

Dynamic link in the stack is the pointer to the base of the activation record instance of the caller.

The format of the activation record is fixed at compile time.

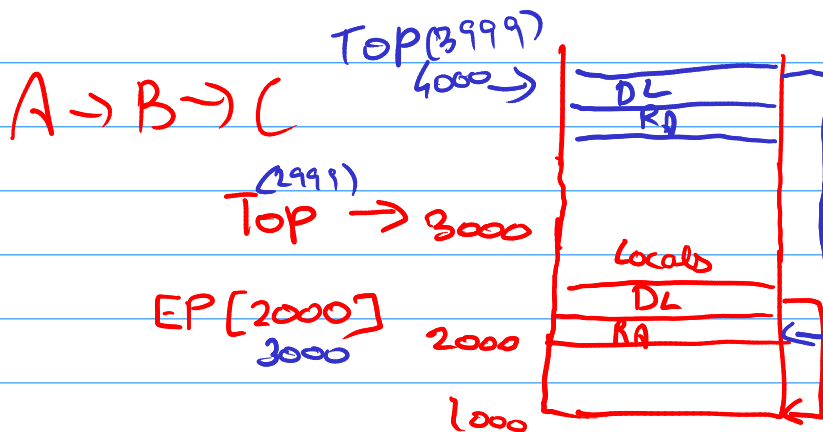
This stack is the part of the runtime system, hence called the **runtime stack**.

Upon return from the subprogram, the stack top is set to the value of current EP minus one.

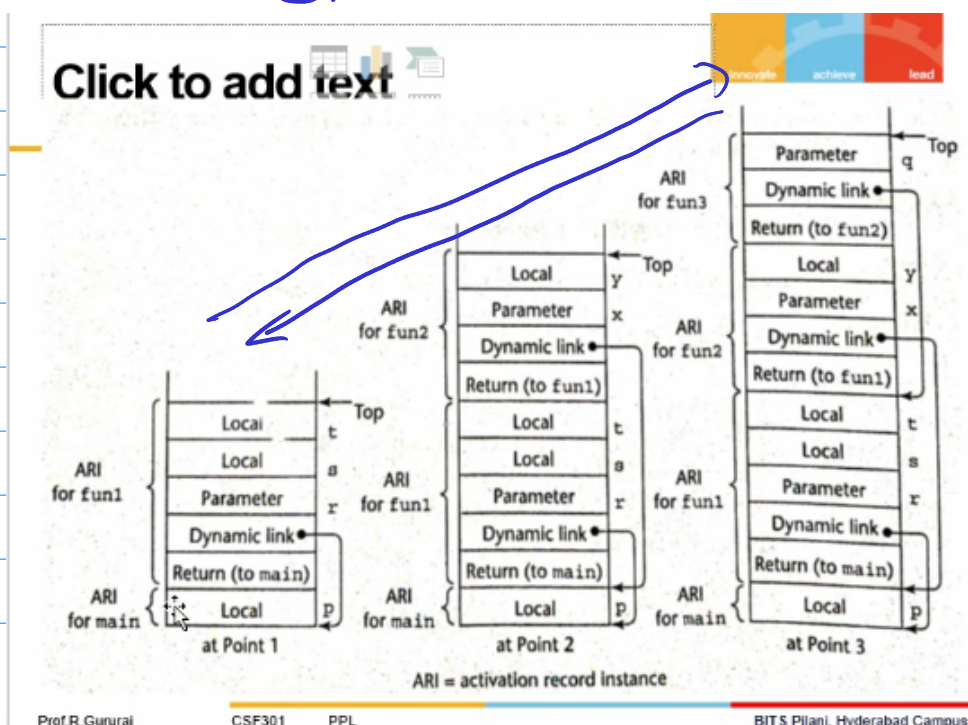
And the EP is set to the dynamic link for the activation record instance of the subprogram that has completed the execution.

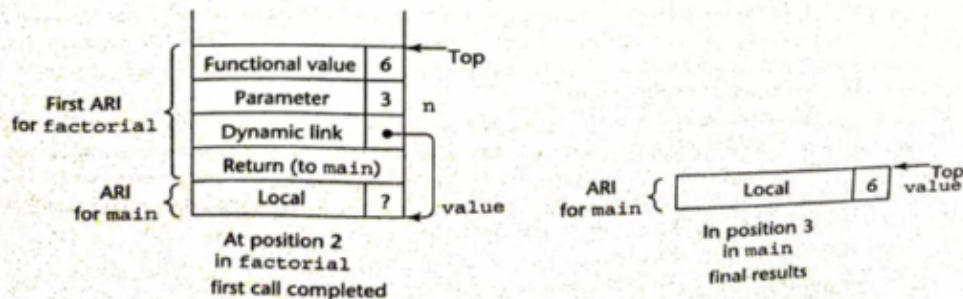
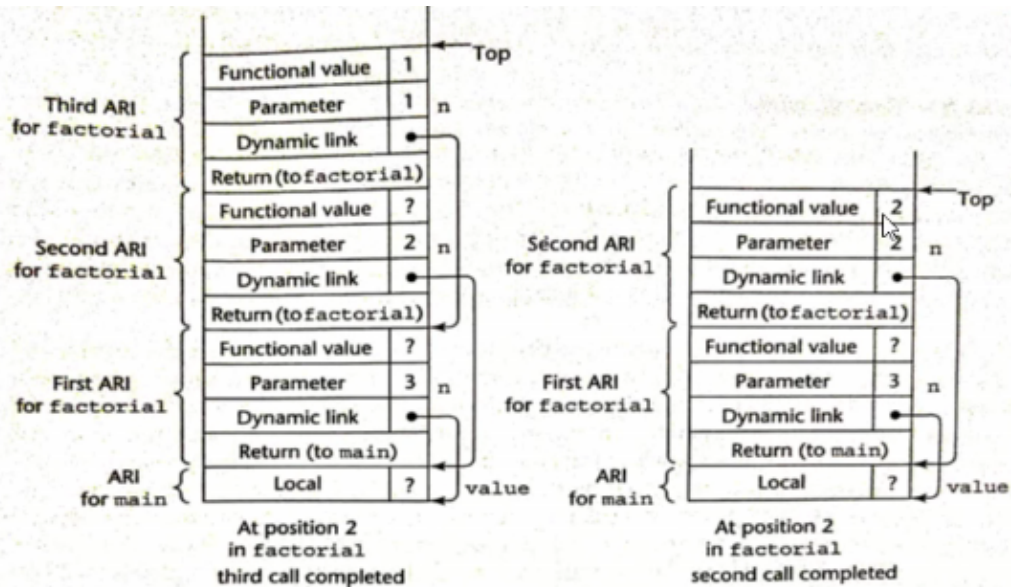
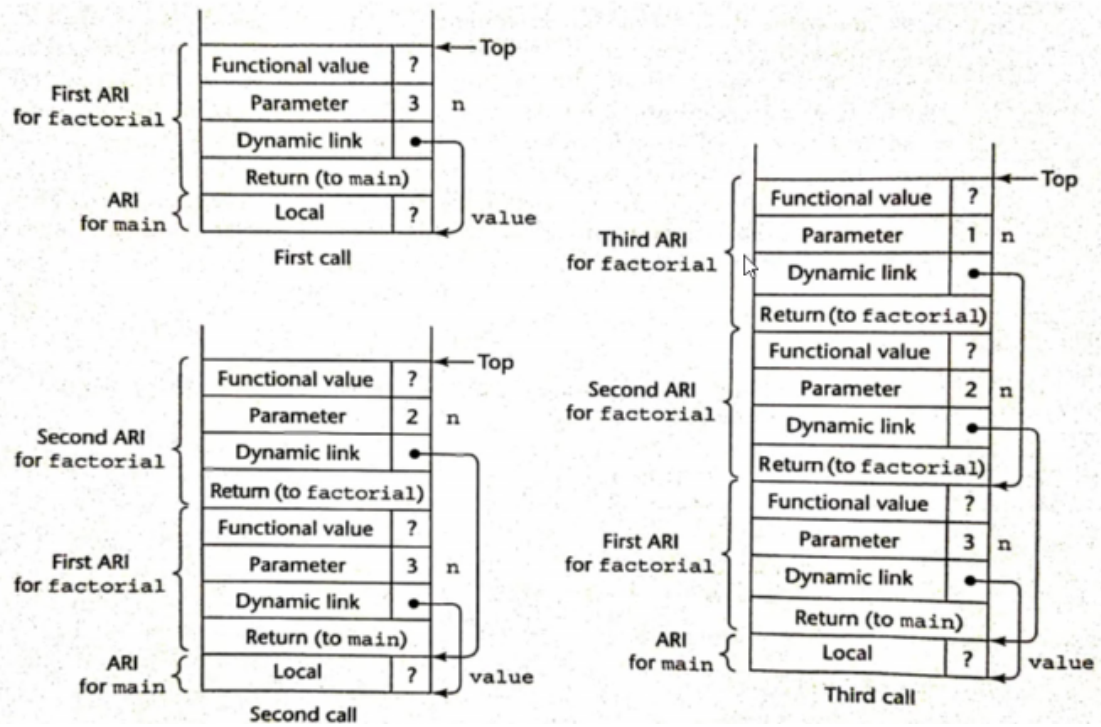
Resetting the top of the stack effectively removes the top activation record instance.

EP is only 1 variable, points to the base address of Active ART (Dynamic link)



Upon return top = 2999
EP = 2000





ARI = activation record instance

Summary of Ch.10

(covered Sec.1 to Sec. 3)



no bin T3
Check notice!

1. Introduction
2. Subprogram linkage
3. Implementing simple subprograms
4. Activation record instance
5. Implementing subprograms with stack-dynamic variables. I
6. Call prologue/epilogue
7. Implementing recursive subprograms.

- [illegible]

[illegible]