

Restore Caller's Frame Pointer from the Stack Frame

```
LDR R7,R5,#2
LDR R5,R5,#1
```

Next, restore caller's frame pointer (R5).

Is there an LC-3 instruction for that?

Note: always the same offset from R5.

What are the base register and offset?

Pop Down to Return Value

```
LDR R7,R5,#2
LDR R5,R5,#1
ADD R6,R6,#3
```

Finally, pop the stack down to the return value.

Is there an LC-3 instruction for that?

Note: amount added depends on space for local variables.

How much do we add?

One More Step... Return!

Time for Step 4: **return to caller.**

A function's code consists of four parts:

1. set up the stack frame,
2. execute the statements,
3. tear down the stack frame (leaving the return value on the stack with LC-3),
4. and return (RET).

We're Done! It's Time to Return to the Caller

```
LDR R7,R5,#2
LDR R5,R5,#1
ADD R6,R6,#3
RET
```

Return to the caller.

Is there an LC-3 instruction for that?