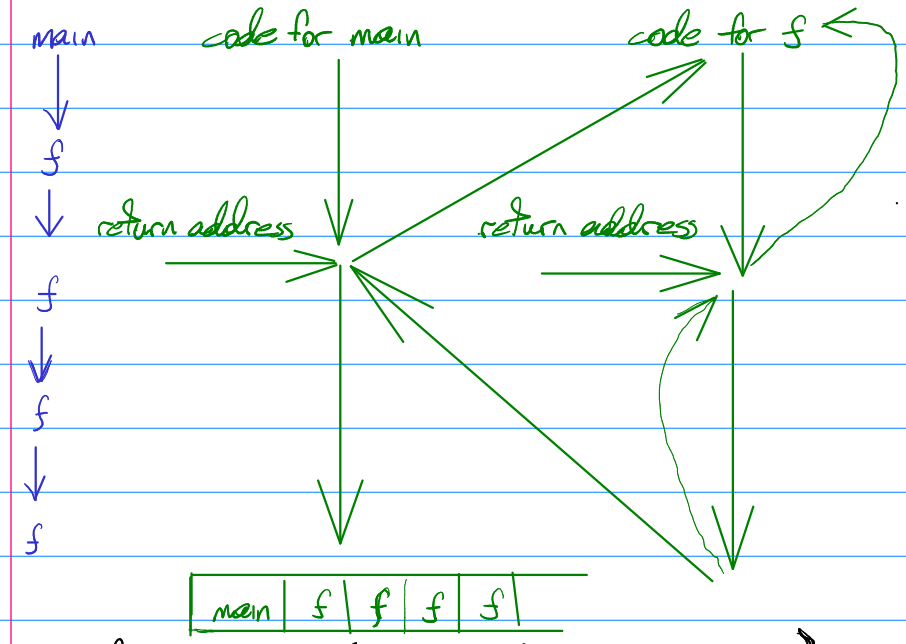
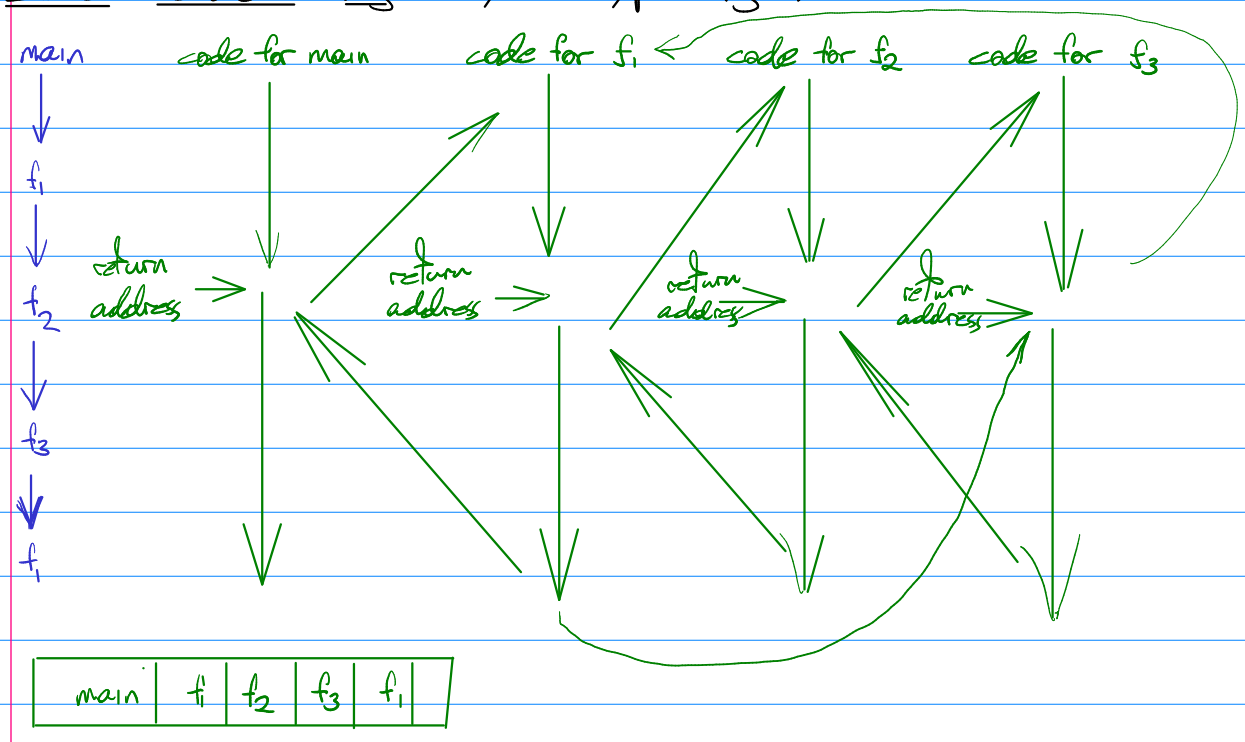
316 10-27-09

Direct Recursion



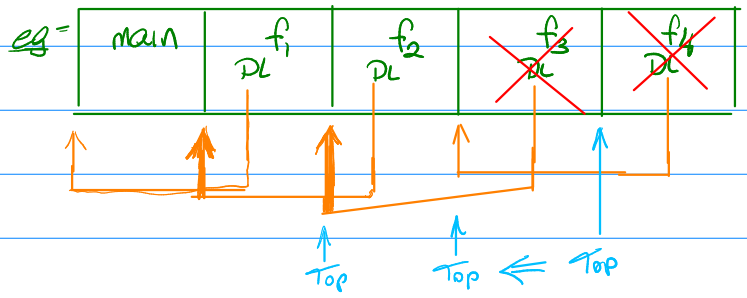
Indirect Recursion (eg = E(), term(), primary())



Stack frame called Activation Record (AR)

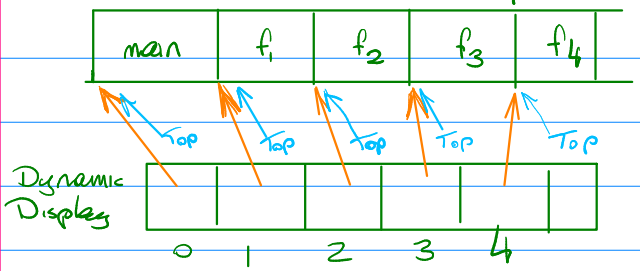
Data collected in ARs

- Return Address = The address of the suspended instruction of the caller. At the callee's return, the control jumps to the return address.
- Dynamic Link = Keeps track of the bottom (base) of ARs. Dynamic Link points to the bottom of the caller's AR.



Data in f_4 are accessed by offsets on Top

Alternative to DLs is a separate set of registers called dynamic display.

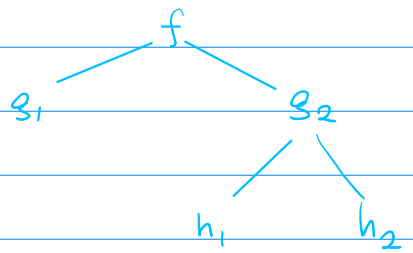


- Static Link = Used only for languages that allow nested declarations of functions. (eg= ADA, PASCAL, LISP)

```
void f(....)
{
   void g1(...)
   {..... body. .. }

   int g2(...)
   {
      void h1(...)
      {... body...}
      void h2(...)
      {.... body...}
      body for g2...
   }
}
```

Static Hierarchy



Which functions can call which functions?

- Static links keep track of the variables / functions that are accessible from each function call.

- **Memory Cells for Formal Parameters**
  - Every function call gets its own memory cells for its formal parameters.
  - Includes "this" or "self" pointer/reference in OOP languages.
    "this" always denotes the reference to the "target" object combined in x
    x can be considered the 0th parameter of f
- **Memory Cells for Local Variables**
  - explicitly declared local variables
  - "hidden" local variables
    - temporary variables generated by compilers for expression evaluations

      $x = a + b + c / (d * e)$

      $t_1 = a + b$

      $t_2 = d * e$

      $t_2 = c / t_2$

      $t_1 = t_1 + t_2$
    - memory space for operand (evaluation) stack
      - operand stack can be grown within AR
      - AR contains a pointer to the operand stack which itself lies outside of the AR
    - memory cells to save register values.
      - the saved register values will be restored at function return
  - Function Return Value = Only needed for non-void functions
    - Alternatively, the address of the memory cell to which the function return value will be assigned.

      $a = f(E_1, \ldots, E_n)$

example =

```
int fact(int n)
{
  if(n <= 1) return 1;
  else return n * fact(n-1);
}
int main()
{
  i = fact(3);
}
```

AR for main
local var i
AR for fact
return address
DL
parameter n
temporary var t