Plan for Today

Calling convention example problem

Block scopes

Dynamic scoping versus static scoping

Nested procedure scopes

Other interesting language features
  – functions as first class objects
  – variable number of parameters
  – structure return values

Suggested Exercises
  – how could we handle named parameters?

Static versus Dynamic Scope

Static Scope
  – also called lexical scope because can determine scoping by analyzing the program
  – each use of a variable is bound to a location statically

Dynamic Scope
  – each use of a variable is bound to the most recently visible defined value for that same variable name

int x = 0;
int f () { return x; }
int g () { int x = 1; return f(); }
Nested Procedures Example

```c
int E(int x)
{
    int G(int z) { int x = 7;
        return F(z); }
    int F(int y) {
        if (y<=0) {
            return 1;
        } else {
            return x + F(y-1);
        }
    }
    return G(2);
}
int main() {
    printf("%d\n", E(4));
}
```

Nested Procedures Suggested Exercise

```c
int foo(int x)
{
    int baz(int y)
    {
        return x+y;
    }
    int bar() {
        return baz(2);
    }
}
int main() {
    printf("%f\n", bar());
}
```

What is the output of the above program?

Draw the stack frame using access links.