

Consider the following skeletal C program:

```
void fun1(void); /* prototype */
void fun2(void); /* prototype */
void fun3(void); /* prototype */
void main() {
    int a, b, c;
    ...
}
void fun1(void) {
    int b, c, d;
    ...
}
void fun2(void) {
    int c, d, e;
    ...
}

void fun3(void) {
    int d, e, f;
    ...
}
```

Given the following calling sequences and assuming that dynamic scoping is used, what variables are visible during execution of the last function called? Include with each visible variable the name of the function in which it was defined.

- a. main calls fun1; fun1 calls fun2; fun2 calls fun3.
- b. main calls fun1; fun1 calls fun3.
- c. main calls fun2; fun2 calls fun3; fun3 calls fun1.
- d. main calls fun3; fun3 calls fun1.
- e. main calls fun1; fun1 calls fun3; fun3 calls fun2.
- f. main calls fun3; fun3 calls fun2; fun2 calls fun1.

Consider the following Pascal skeletal program:

```
program main;  
  var x : integer;  
  procedure sub3; forward;  
  procedure sub1;  
    var x : integer;  
    procedure sub2;  
      begin { sub2 }  
        ...  
      end; { sub2 }  
    begin { sub1 }  
      ...  
    end; { sub1 }  
  procedure sub3;  
    begin { sub3 }  
      ...  
    end; { sub3 }  
  begin { main }  
    ...  
  end. { main }
```

Assume that the execution of this program is in the following unit order:

```
main calls sub1  
sub1 calls sub2  
sub2 calls sub3
```

- a. Assuming static scoping, which declaration of **x** is the correct one for a reference to **x** in the following:
 - i. sub1
 - ii. sub2
 - iii. sub3
- b. Repeat part a, but assume dynamic scoping.