

Question (Concepts of Programming Languages).

Consider the following pseudo-code in an imperative language:

```
int x = 1, n = 10;
```

```
function f () {  
    int x = 3;
```

```
    function g (int c,d,e) {  
        c = c-1;  
        print(c,d,e,x,n);  
    } /* end of g */
```

```
    /* body of f */  
    g(x,x,x*x);  
} /* end of f */
```

```
function h () {  
    int n = 2;  
  
    f();  
} /* end of h */
```

```
main() { h(); }
```

a) inside h()

n = 2

inside f()

x = 3

g(3,3,9)

inside g

c = 2

2 3 9 3 10

- Suppose that the language uses lexical scoping and pass by value. What is printed?
- Suppose that the language uses dynamic scoping and pass by reference. What is printed?
- Suppose that the language uses dynamic scoping and pass by name. What is printed?