Question 1 of 6

Given the following code in a C-like language,

```c
int a[2] = {0, 1, 2};
void func() {
    a[0] = 3;
}

void main() {
    func();
    printf("%d\n", a[0]);
}
```

What is the printing output to the parameter passing mechanism applied to the function `func`?

A. 1.0
B. 1.2
C. 0.13
D. 0.12
E. No output

Q. 1) By value return
Q. 2) By reference
Q. 3) By value
Q. 4) Not catch
Q. 5) By constant reference

---

Control Abstraction

Table of Contents

Part 1 of 1

Question 2 of 6

What kind of parameter passing mechanism exploits "lazy evaluation"?

A. Pass by name
B. Pass by reference
C. Pass by result
D. Pass by value
E. Pass by value result

Reset Selection
Part 1 of 1

Question 3 of 6

Given the following code in C++:

```cpp
void f(int a, int b, int c) {
    if (a > 0) x = y; else y = x;
}
```

```cpp
void main()
int a = 5, b = 7, c = 9;
p = 6, q = 8;
scanf("%d", &x);
}
```

If the statement `y = a` in `if` statement is executed, the variable is modified?

- A. a
- B. b
- C. q
- D. p
- E. y

Reset Selection
Control Abstraction

Part 1 of 1:

Question 5 of 6
Which property of simple call-tern subprogram does scheduled use overcome?

A. Explores call site
B. Single Execution
C. No recursion
D. Single Entry
E. Immediate control transfer

1.0 Points

Question 6 of 6
Given the following code in a C-like language:

```c
int f(int x) { return x; }
int g(int x) { return f(f(x)); }
int h(int x) { return x; }
```

Match the value of z after the last statement is executed to the mechanism applied?

A. 11
B. Other value
C. 13
D. 19

1.0 Points