1, False
Counter example is: (Lx.z (Ly.(y y) Ly.(y y))).
Explanation: In this example, leftmost-outermost strategy will terminate, but leftmost-innermost strategy will not terminate; This is because the function will output some constant with respect to the arguments, we can just ignore the redexes inside the arguments.

2, True
Example: (Lx.((Ly.x y) x) Lx.(x x))
Explanation: leftmost-outermost will not terminate and leftmost-innermost also won't terminate. The reason is that leftmost-outermost strategy is like taking argument into functions, it will not change the whole function itself, if non-termination happens, this means that the function itself can not terminate.

3, False
Counter example is: (Lx.y (Lx.x x)).
Leftmost-innermost:
(Lx.y (Lx.x x))
=>(Lx.y x)
=>y
Leftmost-outermost:
(Lx.y (Lx.x x))
=>y
Explanation: This is because in leftmost-outermost strategy, if the function will output some constant with respect to the arguments, we can just ignore the redex inside the arguments.

4, False
Counter example is: (Lf.(f (f x)) (Lx.x y)).
Leftmost-innermost:
(Lf.(f (f x)) (Lx.x y))
=>(Lf.(f (f x)) y)
=>(y (y x))
Leftmost-outermost:
(Lf.(f (f x)) (Lx.x y))
=>((Lx.x y) ((Lx.x y) x))
=>(y ((Lx.x y) x))
=>(y (y x))
Explanation: This is because if the leftmost-outermost require us to substitute some variables with complex functions, it will lead to more steps to reduce these functions than if we can reduce the complex term at the beginning.