(b) What is the running time complexity of your function? Justify.

The time complexity of my particular solution is O(Log N) this is because on each recursive call the size of the problem gets cut in half, as an example given with starting at 63 on the second pass, this becomes 31 + 1 extra multiplication, on the second pass its only 15 + 1 extra multiplication, and then its just 7 + 1 extra multiplication ect… On each consecutive recursive the problem has been cut in roughly half. Thus we can conclude that the time complexity is O(log n).

(c) Give a number of multiplications used by your function to calculate x 63.
Starting at 63 we have
63 → 31*2 + 1 extra multiplication of the base → 15*2 + 1 → 7*2 + 1 → 3*2 + 1 → 1*2 + 1 → 0*2 + 1.
adding all the odd 1 gets us 6
each multiplication is repeated on the higher step so we get that we have 12 multiplications.