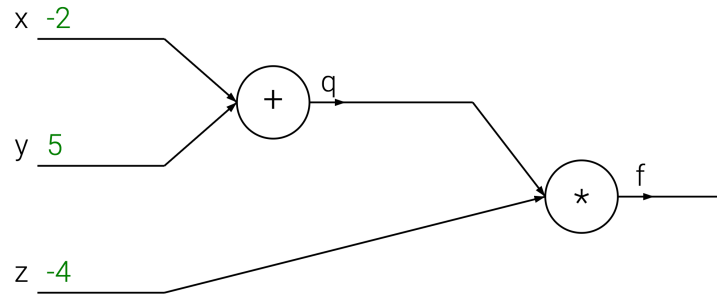


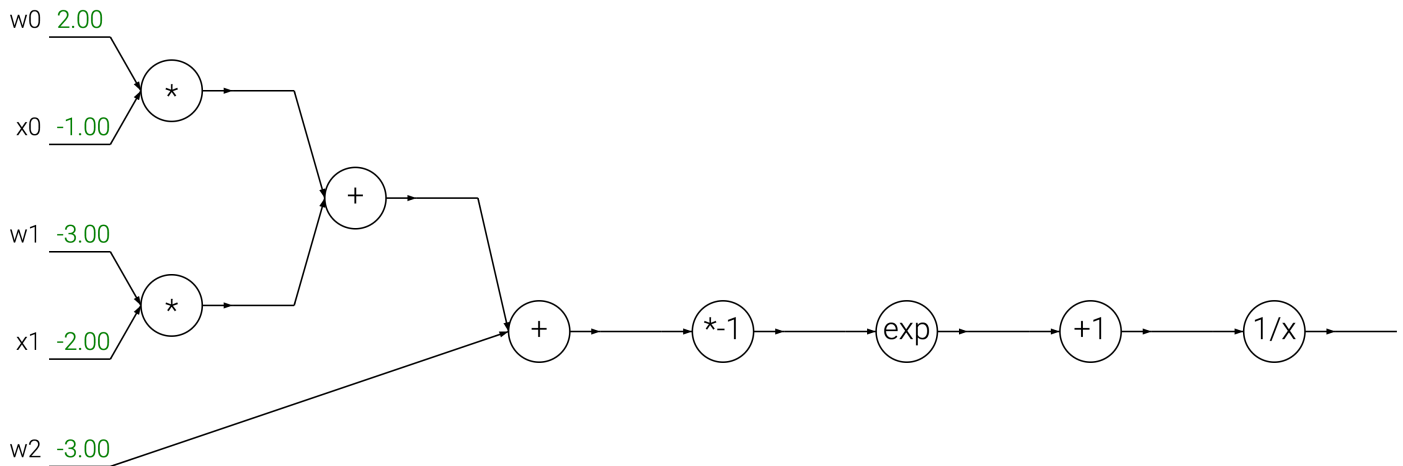
**Backpropagation**

*(find and work with a partner)*

1. *Example 1.* Let  $f(x, y, z) = (x + y)z$ , a function of three inputs. Let  $q = x + y$ , so we can rewrite this function as  $f = qz$ . To determine how  $f$  changes as each input changes, we will use backpropagation through this neural network. First run the “forward pass” to compute the output value of each node (write above the lines). Then use the idea of the chain rule to compute the derivative of each node with respect to the inputs (write below the lines).



2. *Example 2.* Let  $f(w_0, w_1, w_2, x_0, x_1) = \frac{1}{1 + e^{-(w_0 x_0 + w_1 x_1 + w_2)}}$ . Again compute the forward pass (can use a calculator) to determine the output value of each node, then use backpropagation to determine the gradients.



*Acknowledgements: examples from Stanford course CS231n: <http://cs231n.github.io/optimization-2/>*