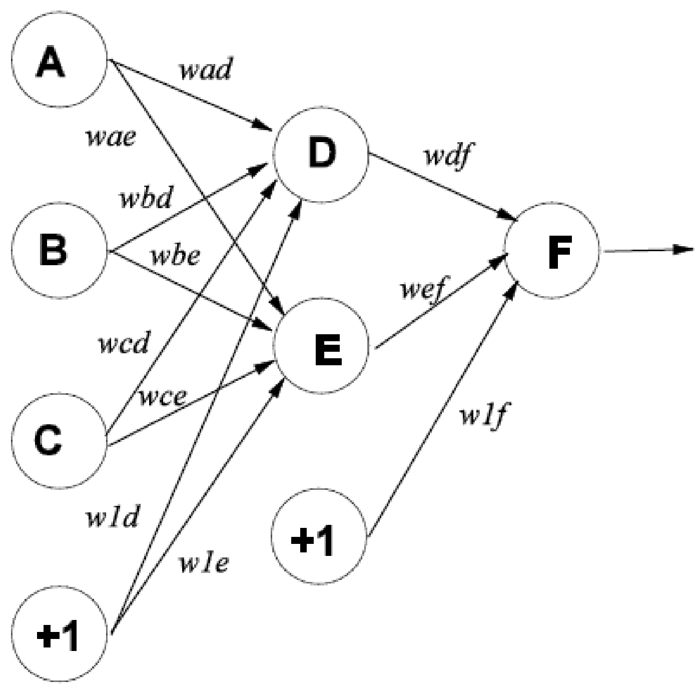
1) Define a neural network with two input units and one output unit that computes the Boolean function (A ∨ ¬B) ∧ (¬A ∨ B). If it is possible to construct a Perceptron that implements this function, define a set of weights that will work. If it is not possible to implement this function using a Perceptron, define a two-layer feed-forward network (i.e., containing one hidden layer) that implements it, using as few hidden units as possible. Let 1 correspond to True and 0 False. Use a step function as the activation function. Draw a figure that shows your network topology and the weights and bias values used.

2) Consider a learning task where there are three real-valued input units, A, B and C, and one output unit, F. You decide to use a 2-layer feed-forward neural network with two hidden units, D and E, defining the hidden layer, as shown in the figure below. The activation function used at nodes D, E and F is the sigmoid function. Each input unit is connected to every hidden unit, and each hidden unit is connected to the output unit. The initial weights and biases are given as: wad = 0.3, wae = -0.1, wbd = 0.3, wbe = -0.1, wcd = 0.3, wce = -0.1, w1d = 0.2, w1e = 0.2, wdf = 0.3, wef = -0.1, w1f = 0.2.



i) What is the output of nodes D, E and F given a training example with values A = 0.3, B = 0.8, and C = 0.1? Assume that D, E, and F all output real values as computed by their associated sigmoid function.

ii) Compute one (1) step of the Back-propagation algorithm using the same inputs given in (i) and teacher output F = 1. The error should be computed as the difference between the integer-valued teacher output and the real valued output of unit F. Using a learning rate of α = 0.2. Give your answer as the 11 new weights and biases. Show your work.