I modified the code by Goel Grus to write my fizzbuzz assignment . I tried the code with 1 ,2, 3 Hidden layers and found that the with the 2rd hidden layer its giving the maximum accuracy go 98% in the test data set. When the third hidden layer is added the accuracy reduced .I think its due to over fitting and be avoided using regularization . Also found out that the program reaches maximum accuracy with less no 0f epochs when we use deeper networks .

```
[193] def init_weights(shape):
          return tf.Variable(tf.random_normal(shape, stddev=0.01))

      w_h = init_weights([NUM_DIGITS, NUM_HIDDEN])
      w_o = init_weights([NUM_HIDDEN_2, 4])
      w_h1 = init_weights([NUM_HIDDEN, NUM_HIDDEN_2])
      w_h2 = init_weights([NUM_HIDDEN_2, NUM_HIDDEN_3])
```

```
    def model(X, w_h, w_o):
        h = tf.nn.relu(tf.matmul(X, w_h))
        h_1 = tf.nn.relu(tf.matmul(h, w_h1))
        h_2 = tf.nn.relu(tf.matmul(h_1, w_h2))
        return tf.matmul(h_2, w_o)
```

Instead of using RELU for activation I tries using sigmoid and tanh .But the accuracy in the test set was not good compare to RELU and it took the gradient decent more epochs for maximum accuracy in the text data set (This can be avoided if we use momentum based gradient decent) .

Each time when we train the model the accuracy changes due to the random weight initialisation .

Also used different optimisation algorithms (Different version of gradient decent like stochastic gradient decent , Adam etc..) We can get the same performance by just using gradient decent with proper learning rate . Just increment the learning rate logarithmically

Results:

```
['1' '2' 'fizz' '4' 'buzz' 'fizz' '7' '8' 'fizz' 'buzz' '11' 'fizz' '13'
 '14' 'fizzbuzz' '16' '17' 'fizz' '19' 'buzz' '21' '22' '23' 'fizz' 'buzz'
 '26' 'fizz' '28' '29' 'fizzbuzz' '31' '32' 'fizz' '34' 'buzz' 'fizz' '37'
 '38' 'fizz' 'buzz' '41' 'fizz' '43' '44' 'fizzbuzz' '46' '47' 'fizz' '49'
 'buzz' 'fizz' '52' '53' 'fizz' 'buzz' '56' 'fizz' '58' '59' 'fizzbuzz'
 '61' '62' 'fizz' '64' 'buzz' 'fizz' '67' '68' 'fizz' 'buzz' '71' 'fizz'
 '73' '74' 'fizzbuzz' '76' 'fizz' 'fizz' '79' 'buzz' '81' '82' '83' '84'
 'buzz' '86' 'fizz' '88' '89' 'fizzbuzz' '91' '92' 'fizz' '94' 'buzz'
 'fizz' '97' '98' 'fizz' 'buzz']
```