In [18]:

```
# algorithm starts at x = 5

x_0 = 5

# learning rate

rate = 0.001

# max number of iteration
max_iters = 10000

# stopping criteria
s = 0.0001

# diff between current x and new x

diff = 1

# initiate number of iteration

iters = 0
```

In [19]:

```
# input function

d = lambda x : 4*(x**3)-(20*x)
```

In [20]:

```python
# while loop for iteration

while diff>s and iters<max_iters:
    # old value of x
    prev_x = x_0
    # new value of x
    x_0 = x_0 - rate*d(prev_x)

    # diff
    diff = abs(x_0 - prev_x)

    # increase iertions by 1

    iters = iters + 1

    # condition for steps

    if iters%10==0:
        print("iteration {} : x = {}".format(iters,x_0))

print("===================================")
print("global minima is at x = {}".format(x_0))
```

```
iteration 10 : x = 3.216578885474962
iteration 20 : x = 2.752765519933938
iteration 30 : x = 2.540026227015159
iteration 40 : x = 2.424157649151624
iteration 50 : x = 2.355698674758306
iteration 60 : x = 2.3133970293306554
iteration 70 : x = 2.286553922614749
iteration 80 : x = 2.2692375130660856
iteration 90 : x = 2.257949336033131
iteration 100 : x = 2.2505410046607888
iteration 110 : x = 2.2456575422507323
iteration 120 : x = 2.2424291258463547
iteration 130 : x = 2.2402907793151225
iteration 140 : x = 2.238872658364325
===================================
global minima is at x = 2.2384494200813188
```

In [ ]:

```python

```