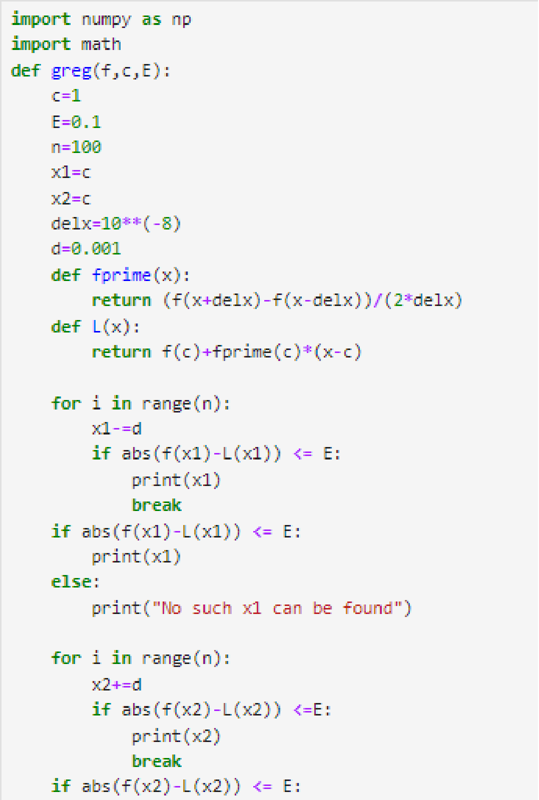
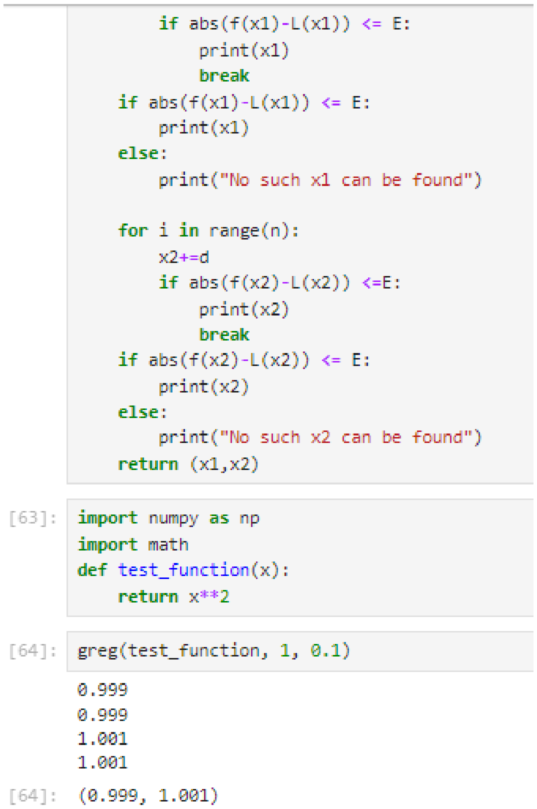


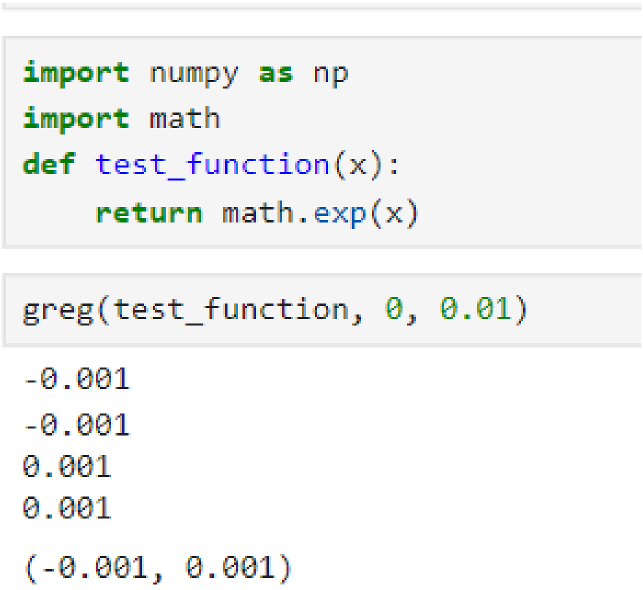
```
import numpy as np
import math
def greg(f,c,E):
    c=1
    E=0.1
    n=100
    x1=c
    x2=c
    delx=10**(-8)
    d=0.001
    def fprime(x):
        return (f(x+delx)-f(x-delx))/(2*delx)
    def L(x):
        return f(c)+fprime(c)*(x-c)

    for i in range(n):
        x1-=d
        if abs(f(x1)-L(x1)) <= E:
            print(x1)
            break
    if abs(f(x1)-L(x1)) <= E:
        print(x1)
    else:
        print("No such x1 can be found")

    for i in range(n):
        x2+=d
        if abs(f(x2)-L(x2)) <=E:
            print(x2)
            break
    if abs(f(x2)-L(x2)) <= E:
```



```
        if abs(f(x1)-L(x1)) <= E:
            print(x1)
            break
    if abs(f(x1)-L(x1)) <= E:
        print(x1)
    else:
        print("No such x1 can be found")

    for i in range(n):
        x2+=d
        if abs(f(x2)-L(x2)) <=E:
            print(x2)
            break
    if abs(f(x2)-L(x2)) <= E:
        print(x2)
    else:
        print("No such x2 can be found")
    return (x1,x2)
```

```
[63]: import numpy as np
      import math
      def test_function(x):
          return x**2
```

```
[64]: greg(test_function, 1, 0.1)

      0.999
      0.999
      1.001
      1.001

[64]: (0.999, 1.001)
```

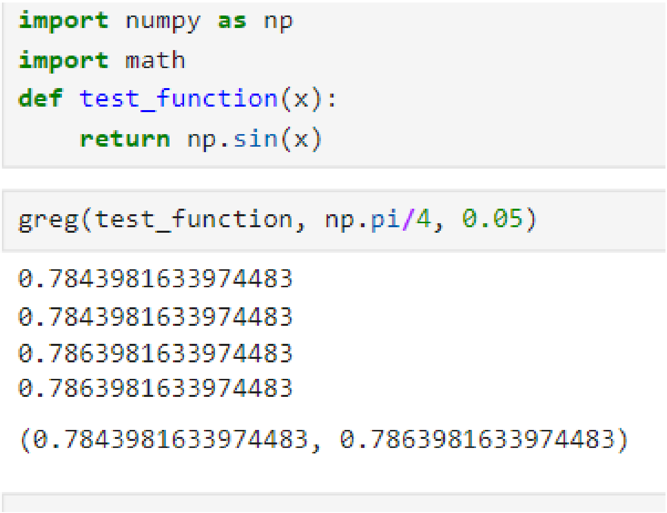


```
import numpy as np
import math
def test_function(x):
    return math.exp(x)
```

```
greg(test_function, 0, 0.01)
```

```
-0.001
-0.001
0.001
0.001
```

```
(-0.001, 0.001)
```



```
import numpy as np
import math
def test_function(x):
    return np.sin(x)
```

```
greg(test_function, np.pi/4, 0.05)
```

```
0.7843981633974483
0.7843981633974483
0.7863981633974483
0.7863981633974483
```

```
(0.7843981633974483, 0.7863981633974483)
```