

The screenshot shows a Jupyter Notebook interface running on a Mac OS X system. The window title is "Lab1b_iSci_fall2022.ipynb". The left sidebar displays a file tree with several notebooks and files, including "R", "Untitled Fo...", "Untitled Fo...", "Lab1a_iSci...", "Lab1b_iSci...", "math1mp3...", "math1mp3...", "Untitled.ip...", "Untitled1.i...", "Untitled2.i...", "Untitled3.i...", "Untitled4.i...", "Untitled5.i...", "Untitled6.i...", and "Untitled7.i...". The main code editor area contains the following Python code:

```
[7]: import numpy as np
import math
def f(x):
    return x**2
c=1
E=0.1
d=0.00001
x1=c
x2=c
deltax= 10**(-2)
def fprime(x,deltax,f):
    return (f(x+deltax)-f(x-deltax))/2*deltax
def Lapprox (x,c,deltax,f):
    return f(c)+fprime(c,deltax,f)*(x-c)
for i in range (0;100):
    x1=x1-d
    if abs (f(x1)-Lapprox(x1,c,deltax,f)) <=E:
        ansx1=x1
    else:
        ansx1="X1 is not found within reasonable range"
for i in range (0,100):
    x2=x2+d
    if abs (f(x2)-Lapprox(x2,c,deltax,f)) <= E:
        ansx2= x2
    else:
        ansx2= "X2 is not found within reasonable range"
print (ansx1,ansx2)
```

The output cell [7] shows the results of the execution:

```
0.999000000000046 1.001000000000066
```

A screenshot of a Jupyter Notebook interface. The top menu bar includes View, Run, Kernel, Tabs, Settings, and Help. The tabs at the top show 'Lab1b_iSci_fall2022.ipynb X', 'Launcher X', 'math1mp3_part2_2022 (2) X', and 'Lab1a_iSci_fall2022.ipynb X'. On the left, a sidebar lists files with their names and last modified dates. The main area shows a code cell [8] containing Python code for numerical methods. The code defines functions for sine approximation and a root-finding algorithm using the bisection method. It includes error handling for cases where roots are not found within a reasonable range. The output of the cell shows the numerical values 0.7843981633974528 and 0.7863981633974437.

```
[8]: import numpy as np
import math
def f(x):
    return np.sin(x)
c=np.pi/4
E=0.05
d=0.0001
x1=c
x2=c
deltax= 10**(-2)
def fprime(x,deltax):
    return (f(x+deltax)-f(x-deltax))/2*deltax
def Lapprox (x,c,deltax,f):
    return f(c)+fprime(c,deltax,f)*(x-c)
for i in range (0;100):
    x1=x1-d
    if abs (f(x1)-Lapprox(x1,c,deltax,f)) <=E:
        ansx1=x1
    else:
        ansx1="X1 is not found within reasonable range"
for i in range (0,100):
    x2=x2+d
    if abs (f(x2)-Lapprox(x2,c,deltax,f)) <= E:
        ansx2= x2
    else:
        ansx2= "X2 is not found within reasonable range"
print (ansx1,ansx2)
```

0.7843981633974528 0.7863981633974437

mcmaster.syzygy.ca/jupyter/user/dhillj46/lab/tree/Untitled7.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name Q

[]:

Name	Last Modified
R	4 days ago
Untitled Fo...	9 days ago
Untitled Fo...	5 days ago
Lab1a_iSci...	9 10/31/23, 1:34 PM
Lab1b_iSci...	5 days ago
math1mp3...	28 days ago
math1mp3...	28 days ago
Untitled.ip...	28 days ago
Untitled1.i...	28 days ago
Untitled2.i...	28 days ago
Untitled3.i...	28 days ago
Untitled4.i...	23 days ago
Untitled5.i...	20 days ago
Untitled6.i...	9 days ago
Untitled7.i...	29 seconds ago

```
[9]: import numpy as np
import math
def f(x):
    return np.exp(x)
c=0
E=0.5
d=0.00001
x1=c
x2=c
deltax= 10**(-2)
def fprime(x,deltax,f):
    return (f(x+deltax)-f(x-deltax))/2*deltax
def Lapprox (x,c,deltax,f):
    return f(c)+fprime(c,deltax,f)*(x-c)
for i in range (0,100):
    x1=x1-d
    if abs (f(x1)-Lapprox(x1,c,deltax,f)) <= E:
        ansx1=x1
    else:
        ansx1="X1 is not found within reasonable range"
for i in range (0,100):
    x2=x2+d
    if abs (f(x2)-Lapprox(x2,c,deltax,f)) <= E:
        ansx2= x2
    else:
        ansx2= "X2 is not found within reasonable range"
print (ansx1,ansx2)
```

-0.0010000000000002 0.0010000000000002

Mode: Command