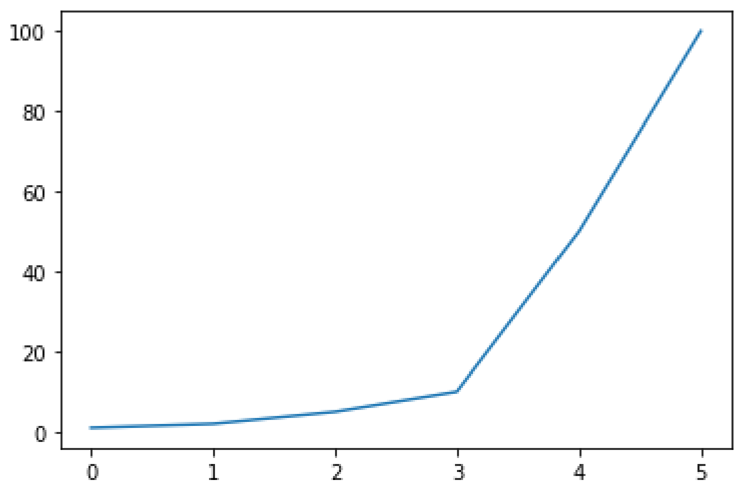
```
# !/usr/bin/env python
# (c) hughes
# ----------------------------------------------------
#

import sys
import math
import random
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
from math import pi,sin,cos


class integration():

  # object attribute
  def __init__(self,a,b,n):
    self.a = a
    self.b = b
    self.n = n

  # object method (function)
  def simpson(self, a, b, n):
    sum = 0
    x = (b-a)/n
    for i in range(1, n):
      if i % 2 == 1:
        sum += 4 * sin(x * i + a)
      elif i % 2 == 0:
        sum += 2 * sin(x * i + a)

    sum += sin(a)
    sum += sin(b)
    integral = x * sum / 3

    print("When n =",n,"the approximate integral of sin(x) is", integral)
    return integral

# create objects
a = 0
b = 7*math.pi
nlist = [1,2,5,10,50,100]

# call objects
for n in nlist:
  result = simpson(a, b, n)
  y = (cos(a)-cos(b))
  print("This interval has a rate of improvement of: ", "%3.2e" % abs(resu
  print("")
```

```
# plot convergence
x = range(6)
y = [n for n in nlist]
plt.figure()
plt.plot(x,y)
plt.show()
```

```
When n = 1 the approximate integral of sin(x) is 6.283990932824513e-15
This interval has a rate of improvement of:  3.18e+14 difference from the true va

When n = 2 the approximate integral of sin(x) is -14.660765716752366
This interval has a rate of improvement of:  1.45e+01 difference from the true va

When n = 5 the approximate integral of sin(x) is -3.1955009359030786
This interval has a rate of improvement of:  2.57e+00 difference from the true va

When n = 10 the approximate integral of sin(x) is 2.5591736269865084
This interval has a rate of improvement of:  1.78e+00 difference from the true va

When n = 50 the approximate integral of sin(x) is 2.000425558854594
This interval has a rate of improvement of:  1.00e+00 difference from the true va

When n = 100 the approximate integral of sin(x) is 2.000026136956072
This interval has a rate of improvement of:  1.00e+00 difference from the true va
```



0s completed at 4:51 PM