P346 (Computational Physics Lab)
Assignment 6
Name: Ratul Das
Roll No.: 1911128

[1]:
```
import mylibrary
import math
from matplotlib import pyplot as plt
```

Question 1

[2]:
```
#function given in Q1
def func(x):
    y = (1+(1/x))**(0.5)
    return y
```

[3]:
```
a = 1
b = 4
iter_list = [8,16,24] #values of N for which we obtain the integrals

print('Solution using Midpoint method:')
print()
print('N    Value of integral')
for N in iter_list:
    print(N,'   ', mylibrary.Midpoint(a,b,func,N))
print()

print('Solution using Trapezoidal method:')
print()
print('N    Value of integral')
for N in iter_list:
    print(N,'   ', mylibrary.Trapezoid(a,b,func,N))
print()

print('Solution using Simpson method:')
print()
print('N    Value of integral')
for N in iter_list:
    print(N,'   ', mylibrary.Simpson(a,b,func,N))
print ()

print('The value of the integral using analytic method is: 3.620184280785752')
#i.e. the actual value of the integral
print ()
```

```
Solution using Midpoint method:
```

```
N     Value of integral
8     3.6239569493985613
16     3.621135404364218
24     3.620607687124767

Solution using Trapezoidal method:

N     Value of integral
8     3.6239569493985626
16     3.621135404364217
24     3.6206076871247665

Solution using Simpson method:

N     Value of integral
8     3.620330143440291
16     3.62019488935277
24     3.620186449815973

The value of the integral using analytic method is: 3.620184280785752
```

Question 2

[4]:
```python
#function given in Q2
def func(x):
    y = x*((1+x)**(0.5))
    return y
```

[5]:
```python
tolerance = 0.001
a = 0   #lower bound
b = 1   #upper bound
d2f = 1
d4f = 1.5      #calculated manually for the maximum value of f''(x) and f''''(x)

N_Midpoint = math.ceil(((b-a)**3*d2f/(24*tolerance))**0.5)        #manually␣
 ↪checked, got N = 7.
print("Value of integral using Midpoint method:", mylibrary.
 ↪Midpoint(a,b,func,N_Midpoint))
print()
N_Trapezoidal = math.ceil(((b-a)**3*d2f/(12*tolerance))**0.5)      #manually␣
 ↪checked, got N = 10.
print("Value of integral using Trapezoidal method:", mylibrary.
 ↪Trapezoid(a,b,func,N_Trapezoidal))
print()
N_Simpson = math.ceil(pow((b-a)**5*d4f/(180*tolerance), 1/4))     #manually␣
 ↪checked, got N = 2.
if N_Simpson%2 == 0:        #applying the condition for even value of N.
```

```
    print("Value of integral using Simpson method:", mylibrary.
 ↪Simpson(a,b,func,N_Simpson))
else:
    print("Value of integral using Simpson method:", mylibrary.
 ↪Simpson(a,b,func,N_Simpson+1))

print()
print('The value of the integral by analytically solving the integral is: 0.
 ↪643790283299492')
print ()
```

```
Value of integral using Midpoint method: 0.6450956918316096

Value of integral using Trapezoidal method: 0.6444300126935936

Value of integral using Simpson method: 0.6439505508593788

The value of the integral by analytically solving the integral is:
0.643790283299492
```

Question 3

[6]:
```
#function in Q3
def func(x):
    y = 4/(1+x**2)
    return y
```

[7]:
```
a = 0  #lower bound
b = 1  #upper bound

n_values = [] #list for storing number of samples
results = []  #corresponding values for no. of samples

print("The values of the integral for different numbers of samples are:")
print()
print('N         Value of integral')

#sampling from 10 and incrementing by multiples of 10
for N in range(10,10000,10):
    result, N_value = mylibrary.MonteCarlo(a,b,func,N) #calculating value of the␣
 ↪integral
    print(N,'            ', result)
    n_values.append(N)                 #recording number of samples
    results.append(result)             #recording results
print(N + 10,'            ', result)  #final result (last value of N)
print()
```

```
print('Value of pi calculated using Monte-Carlo method (with N = 10000) is: ',␣
 ↪result)
print()

#Plot for pi against N
plt.plot(n_values, results, 'r-')
plt.title("Value of $\pi$ vs number of samples, $N$")
plt.xlabel("Number of samples, $N$")
plt.ylabel("Value of $\pi$")
plt.show()
```

```
The values of the integral for different number of iterations are:

N          Value of integral
10               3.6
20               3.2
30               3.066666666666667
40               3.4
50               3.04
60               3.3333333333333335
70               3.0285714285714285
80               3.1
90               3.066666666666667
100               2.88
110               3.018181818181818
120               2.8666666666666667
130               3.2
140               3.0285714285714285
150               3.0933333333333333
160               3.2
170               3.176470588235294
180               3.0
190               3.0736842105263156
200               3.02
210               3.1619047619047618
220               3.1454545454545455
230               3.2695652173913046
240               3.0833333333333335
250               3.328
260               3.0153846153846153
270               3.037037037037037
280               3.242857142857143
290               3.1724137931034484
300               2.986666666666667
310               3.161290322580645
320               3.1375
330               3.272727272727273
```

340	3.1294117647058823
350	3.0742857142857143
360	3.1333333333333333
370	3.1135135135135137
380	3.1052631578947367
390	3.0153846153846153
400	3.07
410	3.053658536585366
420	3.1714285714285713
430	3.1534883720930234
440	3.0454545454545454
450	3.04
460	3.008695652173913
470	3.251063829787234
480	3.0416666666666665
490	3.2
500	3.216
510	3.168627450980392
520	3.1076923076923078
530	3.018867924528302
540	3.1555555555555554
550	3.12
560	3.107142857142857
570	3.087719298245614
580	3.0965517241379312
590	3.0847457627118646
600	3.2
610	3.1868852459016392
620	3.264516129032258
630	3.257142857142857
640	3.18125
650	3.1753846153846155
660	3.175757575757576
670	3.1343283582089554
680	3.052941176470588
690	3.055072463768116
700	3.1028571428571428
710	3.0704225352112675
720	3.061111111111111
730	3.1013698630136988
740	3.1405405405405404
750	3.184
760	3.236842105263158
770	3.116883116883117
780	3.18974358974359
790	3.2253164556962024
800	3.31
810	3.1160493827160494

820	3.0878048780487806
830	3.1180722891566264
840	3.119047619047619
850	3.1576470588235295
860	3.2418604651162792
870	3.232183908045977
880	3.109090909090909
890	3.1730337078651685
900	3.1733333333333333
910	3.1032967032967034
920	3.134782608695652
930	3.2301075268817203
940	3.072340425531915
950	3.094736842105263
960	3.129166666666667
970	3.1876288659793817
980	3.126530612244898
990	3.187878787878788
1000	3.184
1010	3.128712871287129
1020	3.164705882352941
1030	3.0834951456310677
1040	3.173076923076923
1050	3.1695238095238096
1060	3.267924528301887
1070	3.095327102803738
1080	3.151851851851852
1090	3.093577981651376
1100	3.16
1110	3.0846846846846847
1120	3.0142857142857142
1130	3.090265486725664
1140	3.1333333333333333
1150	3.1095652173913044
1160	3.103448275862069
1170	3.135042735042735
1180	3.135593220338983
1190	3.183193277310924
1200	3.1866666666666665
1210	3.133884297520661
1220	3.3278688524590163
1230	3.1642276422764226
1240	3.1193548387096772
1250	3.1488
1260	3.2063492063492065
1270	3.1653543307086616
1280	3.11875
1290	3.1286821705426355

1300	3.169230769230769
1310	3.151145038167939
1320	3.2181818181818183
1330	3.1398496240601506
1340	3.1522388059701494
1350	3.194074074074074
1360	3.088235294117647
1370	3.164963503649635
1380	3.1623188405797102
1390	3.2028776978417266
1400	3.177142857142857
1410	3.15177304964539
1420	3.1774647887323946
1430	3.1132867132867132
1440	3.136111111111111
1450	3.219310344827586
1460	3.0684931506849313
1470	3.1129251700680274
1480	3.1486486486486487
1490	3.1328859060402685
1500	3.1653333333333333
1510	3.11523178807947
1520	3.1526315789473682
1530	3.1790849673202612
1540	3.2
1550	3.153548387096774
1560	3.076923076923077
1570	3.1414012738853505
1580	3.1265822784810124
1590	3.167295597484277
1600	3.045
1610	3.120496894409938
1620	3.123456790123457
1630	3.2245398773006135
1640	3.15609756097561
1650	3.1563636363636363
1660	3.1108433734939758
1670	3.178443113772455
1680	3.1
1690	3.178698224852071
1700	3.216470588235294
1710	3.167251461988304
1720	3.0953488372093023
1730	3.2786127167630057
1740	3.135632183908046
1750	3.12
1760	3.102272727272727
1770	3.157062146892655

1780	3.1101123595505618
1790	3.177653631284916
1800	3.117777777777778
1810	3.1845303867403314
1820	3.1142857142857143
1830	3.1890710382513663
1840	3.1
1850	3.1135135135135137
1860	3.189247311827957
1870	3.202139037433155
1880	3.029787234042553
1890	3.168253968253968
1900	3.1389473684210527
1910	3.1790575916230366
1920	3.129166666666667
1930	3.0196891191709843
1940	3.1484536082474226
1950	3.1466666666666665
1960	3.1530612244897958
1970	3.1370558375634516
1980	3.096969696969697
1990	3.171859296482412
2000	3.136
2010	3.0865671641791046
2020	3.198019801980198
2030	3.091625615763547
2040	3.1941176470588237
2050	3.147317073170732
2060	3.0407766990291263
2070	3.101449275362319
2080	3.0923076923076924
2090	3.1157894736842104
2100	3.097142857142857
2110	3.0938388625592417
2120	3.1056603773584905
2130	3.175586854460094
2140	3.216822429906542
2150	3.1962790697674417
2160	3.161111111111111
2170	3.133640552995392
2180	3.108256880733945
2190	3.1835616438356165
2200	3.130909090909091
2210	3.190950226244344
2220	3.1531531531531534
2230	3.1515695067264575
2240	3.1464285714285714
2250	3.105777777777778

2260	3.0654867256637166
2270	3.170044052863436
2280	3.1649122807017545
2290	3.102183406113537
2300	3.1043478260869564
2310	3.1186147186147184
2320	3.160344827586207
2330	3.107296137339056
2340	3.1452991452991452
2350	3.116595744680851
2360	3.176271186440678
2370	3.1932489451476793
2380	3.1243697478991597
2390	3.123012552301255
2400	3.11
2410	3.1784232365145226
2420	3.1603305785123967
2430	3.106172839506173
2440	3.1114754098360655
2450	3.1363265306122448
2460	3.196747967479675
2470	3.097975708502024
2480	3.1516129032258067
2490	3.140562248995984
2500	3.136
2510	3.1107569721115538
2520	3.1126984126984127
2530	3.177865612648221
2540	3.0929133858267717
2550	3.1419607843137256
2560	3.1890625
2570	3.2
2580	3.13953488372093
2590	3.145945945945946
2600	3.126153846153846
2610	3.117241379310345
2620	3.1114503816793895
2630	3.149809885931559
2640	3.1333333333333333
2650	3.1743396226415093
2660	3.1278195488721803
2670	3.1250936329588015
2680	3.1402985074626866
2690	3.1776951672862452
2700	3.1866666666666665
2710	3.180811808118081
2720	3.1294117647058823
2730	3.1545787545787545

2740	3.181021897810219
2750	3.1563636363636363
2760	3.1405797101449275
2770	3.165342960288809
2780	3.1064748201438848
2790	3.1268817204301076
2800	3.1357142857142857
2810	3.1330960854092527
2820	3.0936170212765957
2830	3.157597173144876
2840	3.1380281690140843
2850	3.1452631578947368
2860	3.1468531468531467
2870	3.1442508710801396
2880	3.1708333333333334
2890	3.1778546712802767
2900	3.129655172413793
2910	3.1285223367697594
2920	3.136986301369863
2930	3.1235494880546075
2940	3.2
2950	3.1077966101694914
2960	3.135135135135135
2970	3.1205387205387205
2980	3.1543624161073827
2990	3.129096989966555
3000	3.1346666666666665
3010	3.1362126245847177
3020	3.189403973509934
3030	3.128712871287129
3040	3.1210526315789475
3050	3.159344262295082
3060	3.108496732026144
3070	3.13485342019544
3080	3.1714285714285713
3090	3.0809061488673137
3100	3.1225806451612903
3110	3.1485530546623792
3120	3.193589743589744
3130	3.1629392971246006
3140	3.086624203821656
3150	3.165714285714286
3160	3.187341772151899
3170	3.186119873817035
3180	3.128301886792453
3190	3.167398119122257
3200	3.18125
3210	3.1090342679127727

3220	3.0956521739130434
3230	3.1393188854489162
3240	3.1358024691358026
3250	3.1544615384615384
3260	3.1263803680981597
3270	3.1376146788990824
3280	3.1390243902439026
3290	3.0869300911854105
3300	3.113939393939394
3310	3.1649546827794564
3320	3.136144578313253
3330	3.171171171171171
3340	3.125748502994012
3350	3.1534328358208956
3360	3.136904761904762
3370	3.1548961424332345
3380	3.1159763313609465
3390	3.0997050147492624
3400	3.136470588235294
3410	3.160117302052786
3420	3.135672514619883
3430	3.147521865889213
3440	3.140697674418605
3450	3.152463768115942
3460	3.179190751445087
3470	3.1227665706051875
3480	3.093103448275862
3490	3.1140401146131804
3500	3.161142857142857
3510	3.14985754985755
3520	3.1772727272727272
3530	3.13314447592068
3540	3.136723163841808
3550	3.1335211267605634
3560	3.141573033707865
3570	3.138375350140056
3580	3.093854748603352
3590	3.1220055710306407
3600	3.137777777777778
3610	3.2033240997229915
3620	3.1171270718232043
3630	3.1404958677685952
3640	3.1263736263736264
3650	3.1013698630136988
3660	3.1595628415300547
3670	3.1607629427792916
3680	3.131521739130435
3690	3.1523035230352305

3700	3.151351351351351
3710	3.1223719676549866
3720	3.1870967741935483
3730	3.153887399463807
3740	3.1497326203208558
3750	3.152
3760	3.1319148936170214
3770	3.1310344827586207
3780	3.129100529100529
3790	3.1598944591029023
3800	3.1378947368421053
3810	3.159055118110236
3820	3.143455497382199
3830	3.147780678851175
3840	3.1635416666666667
3850	3.1605194805194805
3860	3.123316062176166
3870	3.145219638242894
3880	3.1649484536082473
3890	3.125964010282776
3900	3.194871794871795
3910	3.152941176470588
3920	3.163265306122449
3930	3.166412213740458
3940	3.1431472081218272
3950	3.15746835443038
3960	3.1545454545454548
3970	3.1123425692695212
3980	3.171859296482412
3990	3.1348370927318294
4000	3.125
4010	3.151122194513716
4020	3.1194029850746268
4030	3.18014888337469
4040	3.1455445544554457
4050	3.1901234567901233
4060	3.1261083743842364
4070	3.1105651105651106
4080	3.134313725490196
4090	3.150122249388753
4100	3.144390243902439
4110	3.2126520681265207
4120	3.207766990291262
4130	3.130266343825666
4140	3.1091787439613525
4150	3.1257831325301204
4160	3.1
4170	3.111750599520384

4180	3.172248803827751
4190	3.1675417661097853
4200	3.1523809523809523
4210	3.1657957244655583
4220	3.146919431279621
4230	3.120567375886525
4240	3.193396226415094
4250	3.1491764705882352
4260	3.1436619718309857
4270	3.159718969555035
4280	3.147663551401869
4290	3.1095571095571097
4300	3.1832558139534886
4310	3.1925754060324825
4320	3.136111111111111
4330	3.1371824480369517
4340	3.0894009216589864
4350	3.135632183908046
4360	3.173394495412844
4370	3.134096109839817
4380	3.1488584474885846
4390	3.123462414578588
4400	3.1472727272727274
4410	3.104761904761905
4420	3.130316742081448
4430	3.166591422121896
4440	3.118918918918919
4450	3.1595505617977526
4460	3.082511210762332
4470	3.1400447427293066
4480	3.1357142857142857
4490	3.1474387527839642
4500	3.1493333333333333
4510	3.1281596452328158
4520	3.1283185840707963
4530	3.1328918322295807
4540	3.143612334801762
4550	3.1384615384615384
4560	3.1447368421052633
4570	3.1413566739606127
4580	3.109170305676856
4590	3.1233115468409585
4600	3.1278260869565218
4610	3.129718004338395
4620	3.187012987012987
4630	3.172354211663067
4640	3.1206896551724137
4650	3.1501075268817202

4660	3.132188841201717
4670	3.147751605995717
4680	3.18034188034188
4690	3.0925373134328358
4700	3.126808510638298
4710	3.1439490445859875
4720	3.1254237288135593
4730	3.172938689217759
4740	3.140084388185654
4750	3.1562105263157894
4760	3.150420168067227
4770	3.1454926624737944
4780	3.1246861924686193
4790	3.1498956158663884
4800	3.1441666666666666
4810	3.140956340956341
4820	3.1477178423236514
4830	3.1006211180124224
4840	3.112396694214876
4850	3.110103092783505
4860	3.1358024691358026
4870	3.108829568788501
4880	3.1418032786885246
4890	3.128834355828221
4900	3.155918367346939
4910	3.1421588594704684
4920	3.1634146341463416
4930	3.1643002028397564
4940	3.146558704453441
4950	3.1547474747474746
4960	3.117741935483871
4970	3.132394366197183
4980	3.159839357429719
4990	3.08376753507014
5000	3.1344
5010	3.1265469061876248
5020	3.1816733067729084
5030	3.137972166998012
5040	3.138888888888889
5050	3.1302970297029704
5060	3.1280632411067195
5070	3.177120315581854
5080	3.089763779527559
5090	3.130058939096267
5100	3.136470588235294
5110	3.1632093933463796
5120	3.13828125
5130	3.1173489278752435

5140	3.1284046692607004
5150	3.1339805825242717
5160	3.1348837209302327
5170	3.1713733075435204
5180	3.1783783783783783
5190	3.1761078998073216
5200	3.163846153846154
5210	3.1570057581573896
5220	3.142528735632184
5230	3.1105162523900574
5240	3.119847328244275
5250	3.158095238095238
5260	3.1330798479087454
5270	3.171157495256167
5280	3.140151515151515
5290	3.1198487712665406
5300	3.169056603773585
5310	3.178154425612053
5320	3.1736842105263157
5330	3.178236397748593
5340	3.108614232209738
5350	3.1334579439252335
5360	3.1417910447761193
5370	3.1314711359404095
5380	3.1680297397769515
5390	3.1628942486085343
5400	3.136296296296296
5410	3.1792975970425137
5420	3.126937269372694
5430	3.1425414364640885
5440	3.1242647058823527
5450	3.1625688073394493
5460	3.136996336996337
5470	3.180255941499086
5480	3.1087591240875914
5490	3.142440801457195
5500	3.146181818181818
5510	3.132486388384755
5520	3.1355072463768114
5530	3.1544303797468354
5540	3.1335740072202167
5550	3.1877477477477476
5560	3.1589928057553958
5570	3.13967684021544
5580	3.1268817204301076
5590	3.1391771019677996
5600	3.1435714285714287
5610	3.1493761140819965

5620	3.1409252669039147
5630	3.1651865008880993
5640	3.1283687943262413
5650	3.142654867256637
5660	3.146996466431095
5670	3.1696649029982362
5680	3.1640845070422534
5690	3.1451669595782072
5700	3.110877192982456
5710	3.132749562171629
5720	3.1468531468531467
5730	3.15043630017452
5740	3.122648083623693
5750	3.1172173913043477
5760	3.115972222222222
5770	3.1285961871750434
5780	3.131487889273356
5790	3.1592400690846287
5800	3.1358620689655172
5810	3.113253012048193
5820	3.1243986254295533
5830	3.1361921097770153
5840	3.1417808219178083
5850	3.170598290598291
5860	3.131058020477816
5870	3.1243611584327087
5880	3.1258503401360542
5890	3.134804753820034
5900	3.1552542372881356
5910	3.16751269035533
5920	3.1635135135135135
5930	3.1595278246205734
5940	3.1616161616161618
5950	3.1361344537815126
5960	3.1402684563758387
5970	3.155108877721943
5980	3.117725752508361
5990	3.1626043405676127
6000	3.112
6010	3.150748752079867
6020	3.1116279069767443
6030	3.1601990049751243
6040	3.1543046357615894
6050	3.111404958677686
6060	3.1425742574257427
6070	3.1532125205930805
6080	3.138157894736842
6090	3.130377668308703

6100	3.156065573770492
6110	3.137152209492635
6120	3.1581699346405228
6130	3.171941272430669
6140	3.1550488599348534
6150	3.1726829268292684
6160	3.1733766233766234
6170	3.175364667747164
6180	3.130097087378641
6190	3.1302100161550888
6200	3.134193548387097
6210	3.139452495974235
6220	3.1517684887459807
6230	3.1640449438202247
6240	3.112820512820513
6250	3.1392
6260	3.136741214057508
6270	3.1661881977671453
6280	3.1515923566878983
6290	3.1860095389507155
6300	3.1707936507936507
6310	3.129001584786054
6320	3.1215189873417724
6330	3.160821484992101
6340	3.1697160883280757
6350	3.130708661417323
6360	3.1452830188679246
6370	3.135949764521193
6380	3.1404388714733544
6390	3.1455399061032865
6400	3.2125
6410	3.11201248049922
6420	3.1333333333333333
6430	3.190668740279938
6440	3.1236024844720496
6450	3.131782945736434
6460	3.141795665634675
6470	3.1208655332302935
6480	3.1333333333333333
6490	3.159938366718028
6500	3.176
6510	3.149615975422427
6520	3.152760736196319
6530	3.146094946401225
6540	3.151681957186544
6550	3.1389312977099237
6560	3.1097560975609757
6570	3.1263318112633183

6580	3.141033434650456
6590	3.1453717754172987
6600	3.150909090909091
6610	3.0886535552193646
6620	3.132930513595166
6630	3.119155354449472
6640	3.154819277108434
6650	3.1278195488721803
6660	3.1771771771771773
6670	3.20119940029985
6680	3.1580838323353295
6690	3.1635276532137517
6700	3.1438805970149253
6710	3.14575260804769
6720	3.1035714285714286
6730	3.122139673105498
6740	3.170326409495549
6750	3.1472592592592594
6760	3.1319526627218934
6770	3.1036927621861152
6780	3.1427728613569323
6790	3.139322533136966
6800	3.1394117647058826
6810	3.171806167400881
6820	3.1331378299120236
6830	3.1437774524158124
6840	3.1578947368421053
6850	3.1328467153284674
6860	3.1533527696793002
6870	3.169141193595342
6880	3.1476744186046512
6890	3.132656023222061
6900	3.1142028985507246
6910	3.123589001447178
6920	3.138728323699422
6930	3.1584415584415586
6940	3.136599423631124
6950	3.1079136690647484
6960	3.1166666666666667
6970	3.1437589670014345
6980	3.1512893982808023
6990	3.12618025751073
7000	3.1605714285714286
7010	3.1366619115549215
7020	3.1435897435897435
7030	3.145376955903272
7040	3.122159090909091
7050	3.1148936170212767

7060	3.1246458923512748
7070	3.1666195190947666
7080	3.1559322033898307
7090	3.1447108603667138
7100	3.148169014084507
7110	3.1448663853727146
7120	3.1269662921348313
7130	3.1360448807854135
7140	3.134453781512605
7150	3.158041958041958
7160	3.130167597765363
7170	3.1430962343096236
7180	3.1454038997214484
7190	3.149930458970793
7200	3.1155555555555554
7210	3.148404993065187
7220	3.168421052631579
7230	3.1330567081604426
7240	3.113259668508287
7250	3.137655172413793
7260	3.169146005509642
7270	3.1334250343878955
7280	3.1527472527472526
7290	3.184087791495199
7300	3.100821917808219
7310	3.1403556771545826
7320	3.153551912568306
7330	3.1121418826739426
7340	3.1291553133514984
7350	3.155918367346939
7360	3.127173913043478
7370	3.1544097693351425
7380	3.1577235772357723
7390	3.1415426251691474
7400	3.134054054054054
7410	3.1605937921727394
7420	3.1417789757412398
7430	3.1779273216689097
7440	3.162903225806452
7450	3.1591946308724834
7460	3.127077747989276
7470	3.1191432396251675
7480	3.121390374331551
7490	3.139652870493992
7500	3.1466666666666665
7510	3.1275632490013314
7520	3.1611702127659576
7530	3.151660026560425

7540	3.146949602122016
7550	3.1650331125827815
7560	3.1417989417989416
7570	3.1619550858652574
7580	3.150395778364116
7590	3.1652173913043478
7600	3.1936842105263157
7610	3.150591327201051
7620	3.1517060367454066
7630	3.146002621231979
7640	3.094764397905759
7650	3.1633986928104574
7660	3.133159268929504
7670	3.17496740547588
7680	3.152604166666667
7690	3.15370611183355
7700	3.131948051948052
7710	3.1263294422827497
7720	3.144559585492228
7730	3.1322121604139714
7740	3.152454780361757
7750	3.1592258064516128
7760	3.147422680412371
7770	3.133590733590734
7780	3.119280205655527
7790	3.1522464698331194
7800	3.182051282051282
7810	3.137516005121639
7820	3.151918158567775
7830	3.1340996168582373
7840	3.147448979591837
7850	3.1194904458598725
7860	3.1353689567430028
7870	3.154764930114358
7880	3.1390862944162436
7890	3.1574144486692015
7900	3.1331645569620252
7910	3.1671302149178255
7920	3.139393939393939
7930	3.1460277427490544
7940	3.1501259445843828
7950	3.1667924528301885
7960	3.1492462311557787
7970	3.1352572145545796
7980	3.1333333333333333
7990	3.139424280350438
8000	3.138
8010	3.1560549313358304

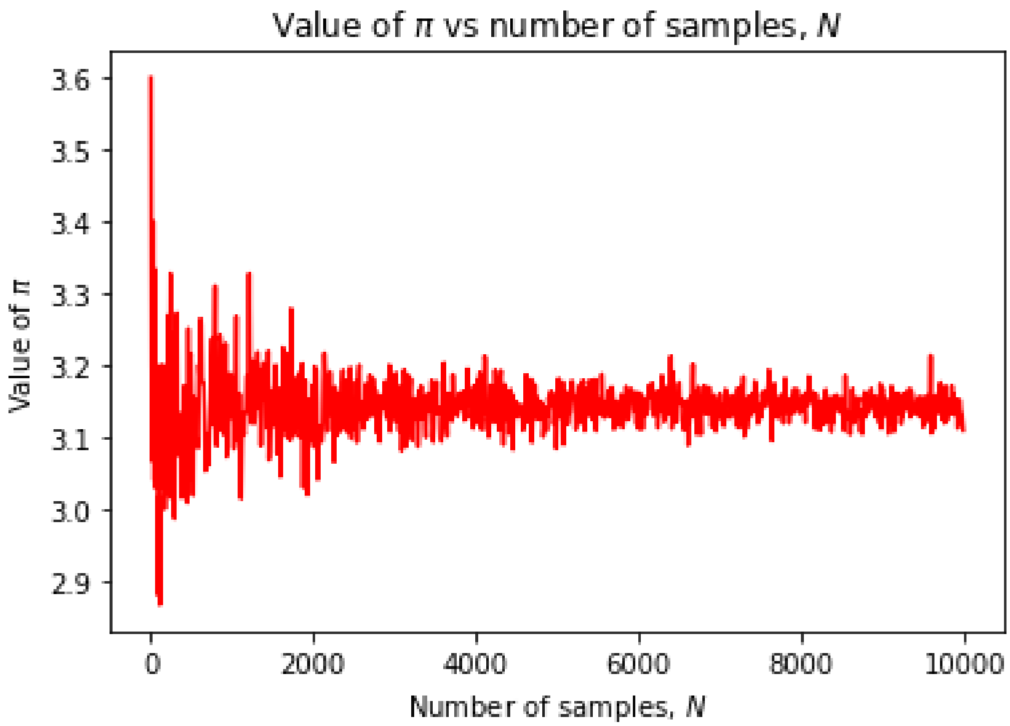
8020	3.1581047381546137
8030	3.1432129514321296
8040	3.1412935323383087
8050	3.12
8060	3.161786600496278
8070	3.1558859975216853
8080	3.1435643564356437
8090	3.184672435105068
8100	3.14962962962963
8110	3.1511713933415537
8120	3.1645320197044335
8130	3.1601476014760146
8140	3.153808353808354
8150	3.160245398773006
8160	3.1254901960784314
8170	3.1113831089351285
8180	3.16039119804401
8190	3.147252747252747
8200	3.131707317073171
8210	3.1395858708891597
8220	3.1304136253041364
8230	3.109599027946537
8240	3.1495145631067962
8250	3.1505454545454548
8260	3.1394673123486685
8270	3.1187424425634824
8280	3.130434782608696
8290	3.1184559710494573
8300	3.166265060240964
8310	3.135980746089049
8320	3.1221153846153844
8330	3.142857142857143
8340	3.126139088729017
8350	3.127185628742515
8360	3.155980861244019
8370	3.107287933094385
8380	3.1470167064439143
8390	3.1194278903456496
8400	3.1376190476190478
8410	3.1434007134363853
8420	3.153919239904988
8430	3.1587188612099646
8440	3.1412322274881515
8450	3.1555029585798815
8460	3.135697399527187
8470	3.13530106257379
8480	3.1528301886792454
8490	3.1552414605418138

8500	3.1336470588235295
8510	3.145475910693302
8520	3.139906103286385
8530	3.1085580304806566
8540	3.1859484777517566
8550	3.134502923976608
8560	3.1322429906542055
8570	3.1355892648774795
8580	3.124941724941725
8590	3.1264260768335275
8600	3.103720930232558
8610	3.127061556329849
8620	3.1327146171693734
8630	3.1675550405561994
8640	3.1259259259259258
8650	3.1195375722543353
8660	3.1496535796766745
8670	3.1257208765859286
8680	3.1557603686635947
8690	3.1429228998849252
8700	3.128735632183908
8710	3.1366245694603903
8720	3.136697247706422
8730	3.106987399770905
8740	3.1363844393592677
8750	3.104
8760	3.1561643835616437
8770	3.1475484606613455
8780	3.163097949886105
8790	3.127645051194539
8800	3.1559090909090908
8810	3.1582292849035185
8820	3.145124716553288
8830	3.1343148357870896
8840	3.1429864253393665
8850	3.1407909604519775
8860	3.1674943566591423
8870	3.152198421645998
8880	3.1463963963963963
8890	3.1559055118110235
8900	3.1447191011235955
8910	3.1281705948372616
8920	3.1560538116591927
8930	3.1552071668533035
8940	3.1507829977628634
8950	3.132067039106145
8960	3.1191964285714286
8970	3.174581939799331

8980	3.112249443207127
8990	3.153726362625139
9000	3.168
9010	3.1289678135405103
9020	3.1547671840354767
9030	3.1499446290143966
9040	3.1535398230088494
9050	3.1323756906077347
9060	3.1456953642384105
9070	3.1047409040793825
9080	3.139647577092511
9090	3.143234323432343
9100	3.1191208791208793
9110	3.171020856201976
9120	3.132456140350877
9130	3.1570646221248633
9140	3.1072210065645516
9150	3.125245901639344
9160	3.1397379912663754
9170	3.1611777535441656
9180	3.144226579520697
9190	3.154298150163221
9200	3.1334782608695653
9210	3.159174809989142
9220	3.1197396963123643
9230	3.118093174431203
9240	3.141991341991342
9250	3.1273513513513516
9260	3.1606911447084234
9270	3.1542610571736787
9280	3.1405172413793103
9290	3.1457481162540364
9300	3.1376344086021506
9310	3.147583243823845
9320	3.166952789699571
9330	3.147695605573419
9340	3.12762312633833
9350	3.1666310160427806
9360	3.161965811965812
9370	3.1568836712913555
9380	3.136460554371002
9390	3.142492012779553
9400	3.1323404255319147
9410	3.1349628055260363
9420	3.146072186836518
9430	3.1550371155885473
9440	3.147457627118644
9450	3.128888888888889

9460	3.1560253699788583
9470	3.1493136219640974
9480	3.1502109704641352
9490	3.1308746048472074
9500	3.114105263157895
9510	3.156256572029443
9520	3.142436974789916
9530	3.1202518363064007
9540	3.1350104821802933
9550	3.1380104712041885
9560	3.1581589958158998
9570	3.142737722048067
9580	3.1532359081419625
9590	3.2133472367049007
9600	3.1054166666666667
9610	3.1567117585848075
9620	3.1501039501039503
9630	3.1115264797507787
9640	3.129460580912863
9650	3.157720207253886
9660	3.1420289855072463
9670	3.157807652533609
9680	3.1380165289256197
9690	3.1331269349845203
9700	3.1595876288659794
9710	3.17610710607621
9720	3.1489711934156377
9730	3.1243576567317572
9740	3.166735112936345
9750	3.142153846153846
9760	3.1192622950819673
9770	3.171340839303992
9780	3.1537832310838447
9790	3.1231869254341165
9800	3.1448979591836737
9810	3.148623853211009
9820	3.153156822810591
9830	3.1340793489318415
9840	3.1329268292682926
9850	3.1715736040609137
9860	3.1342799188640975
9870	3.160688956433637
9880	3.1311740890688258
9890	3.151061678463094
9900	3.138181818181818
9910	3.15196770938446
9920	3.1125
9930	3.150453172205438

```
9940               3.1525150905432597
9950               3.151758793969849
9960               3.1353413654618474
9970               3.1334002006018054
9980               3.122645290581162
9990               3.1095095095095093
10000               3.1095095095095093

Value of pi calculated using Monte-Carlo method (with N = 10000) is:
3.1095095095095093
```

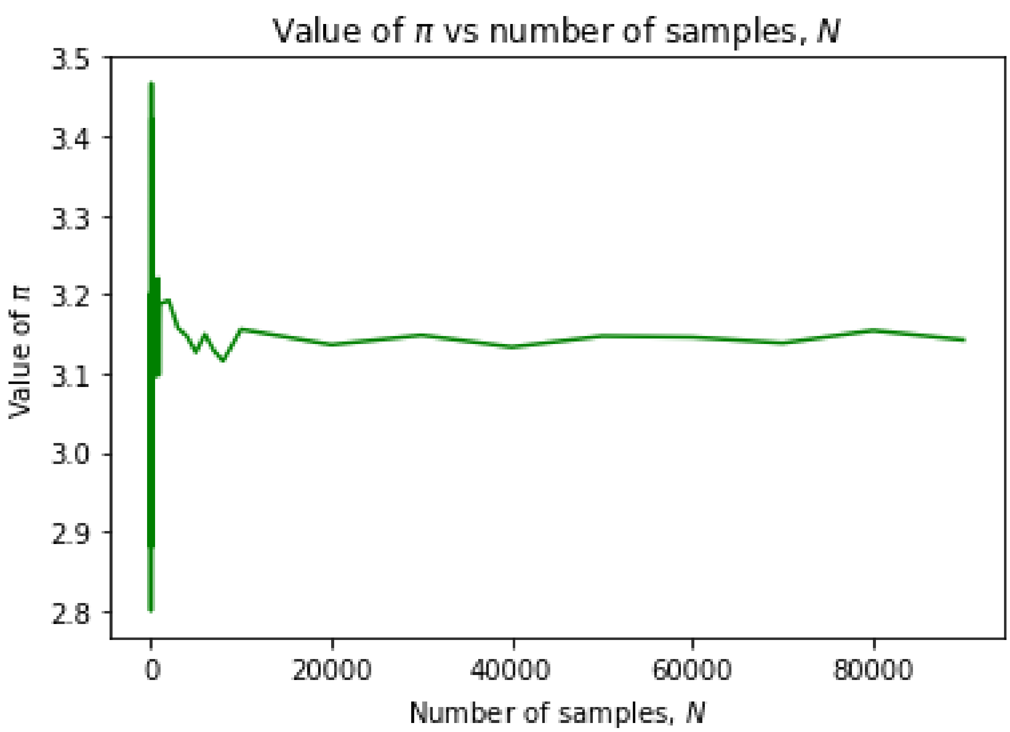


[11]:
```python
a = 0
b = 1
n_values = []
results = []

print("The values of the integral for different numbers of samples are:")
print()
print(' N          Value of integral')

#finding results for multiples of exponents of 10
for exponent in range(1,5):
```

```
    for digit in range(1,10):
        result, N_value = mylibrary.MonteCarlo(a,b,func,(10**exponent)*digit)
        print((10**exponent)*digit,'            ', result)
        n_values.append((10**exponent)*digit)         #recording number of␣
 ↪iterations
        results.append(result)                        #recording results
print(10**(exponent+1),'            ', result)        #final result (last value␣
 ↪of N)
print()
print('Value of pi calculated using Monte-Carlo method (with N = 100000) is: ',␣
 ↪result)

#Plot for pi against N
plt.plot(n_values, results, 'g-')
plt.title("Value of $\pi$ vs number of samples, $N$")
plt.xlabel("Number of samples, $N$")
plt.ylabel("Value of $\pi$")
plt.show()
```

```
The values of the integral for different numbers of samples are:

 N         Value of integral
10              3.2
20              2.8
30              3.2
40              3.2
50              2.88
60              3.466666666666667
70              3.2
80              3.2
90              3.422222222222222
100              3.0
200              2.96
300              3.1866666666666665
400              3.12
500              3.096
600              3.1533333333333333
700              3.182857142857143
800              3.22
900              3.097777777777778
1000              3.188
2000              3.192
3000              3.1573333333333333
4000              3.147
5000              3.1264
6000              3.1493333333333333
7000              3.1285714285714286
```

```
8000                 3.1155
9000                 3.135111111111111
10000                 3.1556
20000                 3.136
30000                 3.1481333333333335
40000                 3.1331
50000                 3.14696
60000                 3.1456
70000                 3.138114285714286
80000                 3.1541
90000                 3.142177777777778
100000                 3.142177777777778

Value of pi calculated using Monte-Carlo method (with N = 100000) is:
3.142177777777778
```



Question 4

```
[12]: #defining functions for Q4
def linear_density(x):
    #linear mass density for given beam
    lambda_x = x**2
    return lambda_x
```

```
def mass_moment(x):
    #element for first moment of mass
    dm = linear_density(x)*x
    return dm
```

[13]:
```
#the rod is placed along a single spatial dimension (here, we take it to be the␣
 ↪x-axis)
#the ends of the rod lie on x = 0 and x = 2 respectively
a = 0
b = 2

#since the mass distribution in the system is continuous,
#we divide the integral of the first moment of mass by the total mass of the␣
 ↪system
#the integrals are calculated using the various numerical methods we have

print("Finding center of mass of the described beam:")
#calculating the centre of mass through Midpoint, Trapezoid and Simpson methods
com_midpoint = mylibrary.Midpoint(a,b,mass_moment,100)/mylibrary.
 ↪Midpoint(a,b,linear_density,100)
com_trapezoid = mylibrary.Trapezoid(a,b,mass_moment,100)/mylibrary.
 ↪Trapezoid(a,b,linear_density,100)
com_simpson = mylibrary.Simpson(a,b,mass_moment,100)/mylibrary.
 ↪Simpson(a,b,linear_density,100)

#showing the centre of mass locations on the x-axis
print("Using Midpoint method - " + str(com_midpoint))
print("Using Trapezoidal method is - " + str(com_trapezoid))
print("Using Simpson method - " + str(com_simpson))
```

```
Finding center of mass of the described beam:
Using Midpoint method - 1.5000749962501878
Using Trapezoidal method is - 1.5000749962501865
Using Simpson method - 1.5000000000000002
```