Lista01Q02e

June 19, 2025

Lista 01

Victor Paiva Paulo Neto

Questões : 1, 2(e), 2(a), 2(g), 2(c), 2(i)

Escreva um programa e envie ao arquivo fonte ou um link do mesmo, o qual tenha como ponto de partida uma funcao a(n) que define o termo geral de uma sequˆencia numerica an = a(n) (use n como a variavel do programa), que sejam fornecidos dois n´umeros naturais nmin e nmax que definem um intervalo [nmim, nmax] para os ındices “plotados”e que tenha duas opcoes exclusivas de execucao:

[2]:
```
import matplotlib.pyplot as plt
```

[3]:
```
#funcao que define a sequencia numerica de termo geral e(n) 2(e)
def e(n):
    return ((n**2) / (n + 1)) - ((n**2) / (n + 2))
```

[4]:
```
#difinição de nmin e nmax
nmin = int(input("Digite o nmin: "))
nmax = int(input("Digite nmax: "))
print(f"nmin: {nmin}, nmax: {nmax}")
```

```
nmin: 10, nmax: 400
```

[5]:
```
#separa nos casos em que o limite é e não é conhecido
existeLimite = (input("eh conhecido se a sequencia converge para um limite L?␣
 ↪(a se nao, b se sim): "))
print(f"opcao escolhida: {existeLimite}")
```

```
opcao escolhida: b
```

[6]:
```
valoresN = []
valoresAn = []
print(f"\n{'n':>5} {'a(n)':>10}")
print("-" * 15)

for n in range( nmin, nmax +1):
    en = e(n)
    print(f"{n:>5} {en:>10.6f}")
```

```
    valoresN.append(n)
    valoresAn.append(en)
```

```
    n       a(n)
---------------
   10   0.757576
   11   0.775641
   12   0.791209
   13   0.804762
   14   0.816667
   15   0.827206
   16   0.836601
   17   0.845029
   18   0.852632
   19   0.859524
   20   0.865801
   21   0.871542
   22   0.876812
   23   0.881667
   24   0.886154
   25   0.890313
   26   0.894180
   27   0.897783
   28   0.901149
   29   0.904301
   30   0.907258
   31   0.910038
   32   0.912656
   33   0.915126
   34   0.917460
   35   0.919670
   36   0.921764
   37   0.923752
   38   0.925641
   39   0.927439
   40   0.929152
   41   0.930786
   42   0.932347
   43   0.933838
   44   0.935266
   45   0.936633
   46   0.937943
   47   0.939201
   48   0.940408
   49   0.941569
   50   0.942685
   51   0.943759
```

52	0.944794
53	0.945791
54	0.946753
55	0.947682
56	0.948578
57	0.949445
58	0.950282
59	0.951093
60	0.951877
61	0.952637
62	0.953373
63	0.954087
64	0.954779
65	0.955450
66	0.956102
67	0.956735
68	0.957350
69	0.957948
70	0.958529
71	0.959094
72	0.959645
73	0.960180
74	0.960702
75	0.961210
76	0.961705
77	0.962188
78	0.962658
79	0.963117
80	0.963565
81	0.964002
82	0.964429
83	0.964846
84	0.965253
85	0.965651
86	0.966040
87	0.966420
88	0.966792
89	0.967155
90	0.967511
91	0.967859
92	0.968199
93	0.968533
94	0.968860
95	0.969180
96	0.969493
97	0.969800
98	0.970101
99	0.970396

100	0.970685
101	0.970969
102	0.971247
103	0.971520
104	0.971788
105	0.972051
106	0.972309
107	0.972562
108	0.972811
109	0.973055
110	0.973295
111	0.973530
112	0.973762
113	0.973989
114	0.974213
115	0.974433
116	0.974649
117	0.974861
118	0.975070
119	0.975275
120	0.975478
121	0.975676
122	0.975872
123	0.976065
124	0.976254
125	0.976440
126	0.976624
127	0.976805
128	0.976983
129	0.977158
130	0.977331
131	0.977501
132	0.977668
133	0.977833
134	0.977996
135	0.978156
136	0.978314
137	0.978469
138	0.978623
139	0.978774
140	0.978923
141	0.979070
142	0.979215
143	0.979358
144	0.979499
145	0.979638
146	0.979776
147	0.979911

148 0.980045
149 0.980177
150 0.980307
151 0.980435
152 0.980562
153 0.980687
154 0.980811
155 0.980933
156 0.981053
157 0.981172
158 0.981289
159 0.981405
160 0.981520
161 0.981633
162 0.981745
163 0.981855
164 0.981964
165 0.982072
166 0.982179
167 0.982284
168 0.982388
169 0.982491
170 0.982592
171 0.982693
172 0.982792
173 0.982890
174 0.982987
175 0.983083
176 0.983178
177 0.983272
178 0.983364
179 0.983456
180 0.983547
181 0.983637
182 0.983725
183 0.983813
184 0.983900
185 0.983986
186 0.984071
187 0.984155
188 0.984238
189 0.984321
190 0.984402
191 0.984483
192 0.984563
193 0.984642
194 0.984720
195 0.984797

196 0.984874
197 0.984950
198 0.985025
199 0.985100
200 0.985173
201 0.985246
202 0.985318
203 0.985390
204 0.985461
205 0.985531
206 0.985600
207 0.985669
208 0.985737
209 0.985805
210 0.985871
211 0.985938
212 0.986003
213 0.986068
214 0.986133
215 0.986196
216 0.986260
217 0.986322
218 0.986384
219 0.986446
220 0.986507
221 0.986567
222 0.986627
223 0.986687
224 0.986745
225 0.986804
226 0.986861
227 0.986919
228 0.986976
229 0.987032
230 0.987088
231 0.987143
232 0.987198
233 0.987252
234 0.987306
235 0.987360
236 0.987413
237 0.987465
238 0.987517
239 0.987569
240 0.987620
241 0.987671
242 0.987722
243 0.987772

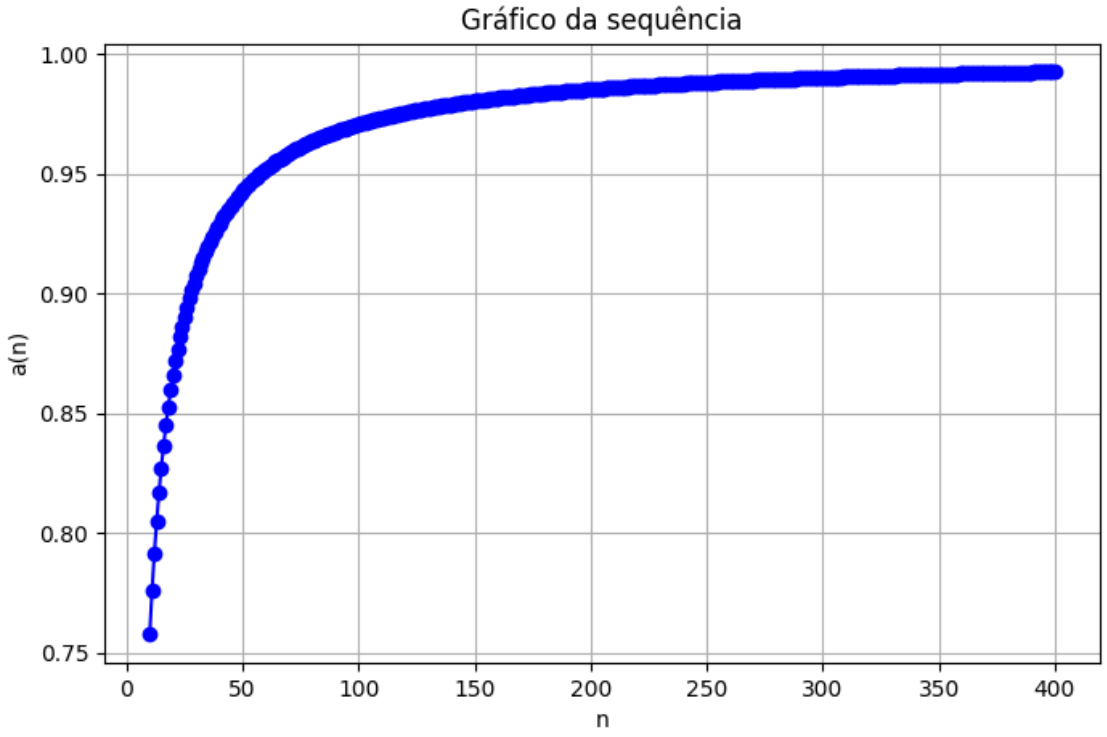
244	0.987821
245	0.987871
246	0.987920
247	0.987968
248	0.988016
249	0.988064
250	0.988111
251	0.988158
252	0.988205
253	0.988251
254	0.988297
255	0.988342
256	0.988387
257	0.988432
258	0.988476
259	0.988520
260	0.988564
261	0.988608
262	0.988651
263	0.988694
264	0.988736
265	0.988778
266	0.988820
267	0.988861
268	0.988903
269	0.988944
270	0.988984
271	0.989024
272	0.989064
273	0.989104
274	0.989144
275	0.989183
276	0.989222
277	0.989260
278	0.989299
279	0.989337
280	0.989374
281	0.989412
282	0.989449
283	0.989486
284	0.989523
285	0.989559
286	0.989595
287	0.989631
288	0.989667
289	0.989703
290	0.989738
291	0.989773

292 0.989808
293 0.989842
294 0.989876
295 0.989910
296 0.989944
297 0.989978
298 0.990011
299 0.990044
300 0.990077
301 0.990110
302 0.990142
303 0.990175
304 0.990207
305 0.990239
306 0.990270
307 0.990302
308 0.990333
309 0.990364
310 0.990395
311 0.990426
312 0.990456
313 0.990486
314 0.990516
315 0.990546
316 0.990576
317 0.990605
318 0.990635
319 0.990664
320 0.990693
321 0.990722
322 0.990750
323 0.990779
324 0.990807
325 0.990835
326 0.990863
327 0.990891
328 0.990918
329 0.990946
330 0.990973
331 0.991000
332 0.991027
333 0.991054
334 0.991080
335 0.991107
336 0.991133
337 0.991159
338 0.991185
339 0.991211

340	0.991237
341	0.991262
342	0.991288
343	0.991313
344	0.991338
345	0.991363
346	0.991388
347	0.991412
348	0.991437
349	0.991461
350	0.991485
351	0.991509
352	0.991533
353	0.991557
354	0.991581
355	0.991605
356	0.991628
357	0.991651
358	0.991674
359	0.991697
360	0.991720
361	0.991743
362	0.991766
363	0.991788
364	0.991811
365	0.991833
366	0.991855
367	0.991877
368	0.991899
369	0.991921
370	0.991943
371	0.991964
372	0.991986
373	0.992007
374	0.992028
375	0.992049
376	0.992071
377	0.992091
378	0.992112
379	0.992133
380	0.992153
381	0.992174
382	0.992194
383	0.992215
384	0.992235
385	0.992255
386	0.992275
387	0.992295

```
388   0.992314
389   0.992334
390   0.992353
391   0.992373
392   0.992392
393   0.992411
394   0.992431
395   0.992450
396   0.992469
397   0.992488
398   0.992506
399   0.992525
400   0.992544
```

[7]:
```
plt.figure(figsize=(8, 5))
plt.plot(valoresN, valoresAn, 'bo-', label='a(n)')
plt.xlabel('n')
plt.ylabel('a(n)')
plt.title('Gráfico da sequência')
plt.grid(True)
```



Limite:

pelo gráfico e por contas realizadas, é possível verificar que a sequencia possui limite igual a 1.

[8]:
```
#funcao que verifica se o modulo da diferença entre o termo geral e o limite é␣
 ↪menor ou igual a epsilon
def N_epsilon_is_true(x, L, epsilon):
    if abs(x - L) <= epsilon:
        return True
    else:
        return False
```

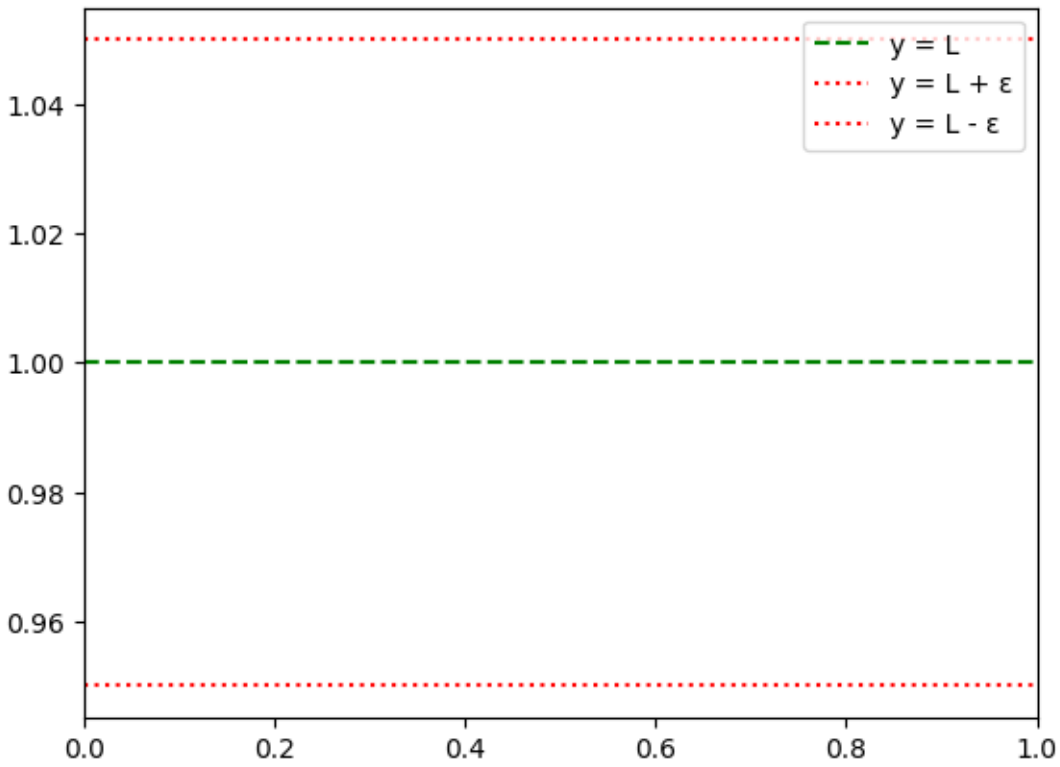
[ ]:
```
if existeLimite == "b":
    L = float(input("Digite o valor do limite L: "))
    epsilon = float(input("Digite a tolerância epsilon: "))
    N_epsilon = float(input("Digite o valor de N(epsilon): "))

    Ne = N_epsilon_is_true(e(n), L, epsilon)

    while Ne == False:
        L = float(input("Digite o valor do limite L: "))
        epsilon = float(input("Digite a tolerância epsilon: "))
        N_epsilon = int(input("Digite o valor de N(epsilon): "))
        Ne = N_epsilon_is_true(e(n), L, epsilon)

    plt.axhline(y=L, color='green', linestyle='--', label='y = L')
    plt.axhline(y=L+epsilon, color='red', linestyle=':', label='y = L +  ')
    plt.axhline(y=L-epsilon, color='red', linestyle=':', label='y = L -  ')

    plt.legend()
    plt.show()
```




```
The Kernel crashed while executing code in the current cell or a previous cell.

Please review the code in the cell(s) to identify a possible cause of the␣
↪failure.

Click <a href='https://aka.ms/vscodeJupyterKernelCrash'>here</a> for more info.

View Jupyter <a href='command:jupyter.viewOutput'>log</a> for further details.
```