

```

1  from numpy import linspace, zeros, log, exp
2
3  def u(x) :
4      return 1/(1 + x**2)
5
6  def Integration(u,a,b,N) :
7      h = (b - a)/N
8      x = linspace(a,b,N+1)
9      integral = 0.
10     for n in range(1,N+1) :
11         integral = integral + (u(x[n-1]) + u(x[n]))/2*h
12     return integral
13
14  a = -1.; b = 1.
15
16  N = 1
17  r = 2; S = 3
18  p = 2; q = 2
19
20  U = zeros((S,S))
21  R = zeros((S,S))
22
23  for s in range(S) :
24      U[s,0] = Integration(u,a,b,r**s*N)
25
26  for s in range(1,S) :
27      for l in range(s) :
28          R[s,l] = (U[s,l] - U[s-1,l])/(r**(p + l*q) - 1)
29          U[s,l+1] = U[s,l] + R[s,l]
30
31  # Функция выводит форматированную таблицу
32  def PrintTriangular(A,i) :
33      print(' ',end=' ')
34      for l in range(len(A)) :
35          print(' p={0:<4d}'.format(p + l*q),end=' ')
36      print()
37      for m in range(len(A)) :
38          print(' s={0:<2d}'.format(m),end=' ')
39          for l in range(m + 1 - i) :
40              print('{0:7.4f}'.format(A[m,l]),end=' ')
41          print()
42      print()
43
44  print('Таблица приближённых значений интеграла:')
45  PrintTriangular(U,0)
46  print('Таблица оценок ошибок:')
47  PrintTriangular(R,1)
48
49  # Листинг программы, реализующей приближённое вычисление интеграла
50  # с помощью рекуррентного сгущения сеток и многократного повышения
51  # точности по Ричардсону

```