

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «САНКТ-ПЕТЕРБУРГСКИЙ
ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО»

ИНСТИТУТ КОМПЬЮТЕРНЫХ НАУК И КИБЕРБЕЗОПАСНОСТИ
ВЫСШАЯ ШКОЛА ПРОГРАММНОЙ ИНЖЕНЕРИИ

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1

**На тему: «Исследование различных методов аппроксимации
функций» по дисциплине: «Вычислительная математика»**

Вариант 6

Выполнила студентка:

Назарова К.А.

группа: в5130904/30030

Проверил:

д.т.н, профессор

Устинов С.М.

Санкт-Петербург, 2024 г.

Задание

По заданной таблице

x	-1.000	-0.960	-0.860	-0.790	0.220	0.500	0.930
f(x)	-1.000	-0.151	0.894	0.986	0.895	0.500	-0.306

построить полином Лагранжа и сплайн-функцию. Вычислить значения обеих функций в точках $x_k = -1 + 0.1k$ ($k = 1, 2, \dots, 19$). Построить графики.

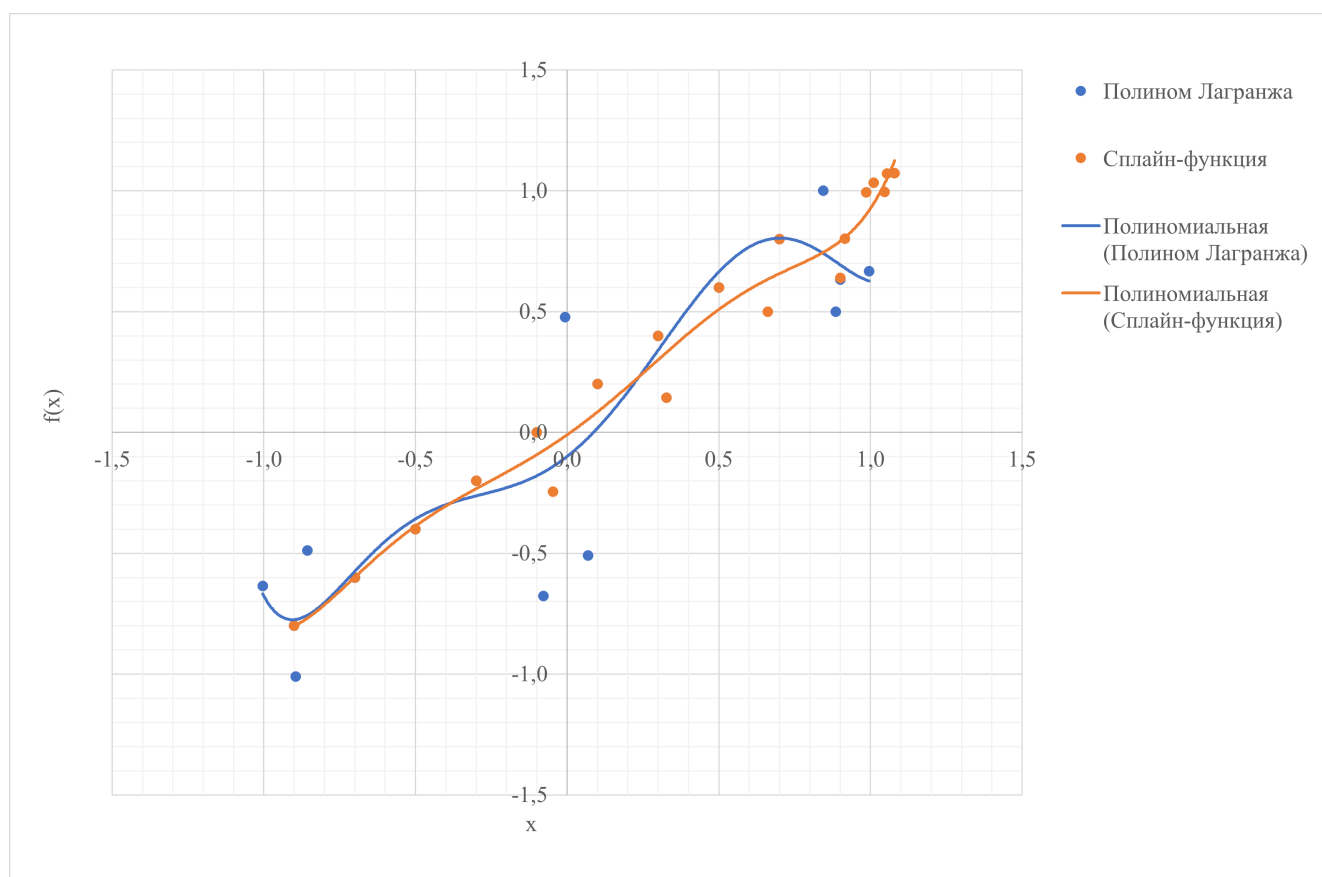
Используя программу **QUANC8**, вычислить интеграл:

$$\int_1^2 \frac{\tan(x)}{x} dx$$

Результаты вычисления полинома Лагранжа и Сплайн-функции

Полином Лагранжа		Сплайн-функция	
х	f(x)	х	f(x)
-0.8999999762	-0.8000000119	-0.8999999762	-0.8000000119
-0.6999999881	-0.6000000238	-0.6999999881	-0.6000000238
-0.5000000000	-0.3999999762	-0.5000000000	-0.3999999762
-0.3000000119	-0.1999999881	-0.3000000119	-0.1999999881
-0.0999999642	0.0000000000	-0.0999999642	0.0000000000
0.1000000238	0.2000000477	0.1000000238	0.2000000477
0.3000000715	0.3999999762	0.3000000715	0.3999999762
0.5000000000	0.6000000238	0.5000000000	0.6000000238
0.7000000477	0.8000000715	0.7000000477	0.8000000715
0.8999999762	0.6322703958	0.8999999762	0.6400555372
0.9958360791	0.6671667099	0.9855651259	0.9932332039
0.0692915618	-0.5091525316	1.0104051828	1.0324817896
-0.8948062658	-1.0105026960	1.0545891523	1.0718533993
-0.8560926318	-0.4893001914	1.0794007778	1.0723575354
-0.0064600259	0.4767180681	1.0458498001	0.9950038791
0.8443872333	0.9999086857	0.9149457812	0.8019503951
0.8851773143	0.5000000000	0.6607797742	0.4999999702
-0.0785126984	-0.6765033007	0.3269802034	0.1442285180
-1.0039086342	-0.6349481344	-0.0469621420	-0.2452987432

Графики полинома Лагранжа и Сплайн-функции



Выводы по результатам вычисления

В результате вычислений были получены узлы интерполяции полинома Лагранжа и сплайн-функции. Согласно данным, приведенным в таблице результатов вычисления, можно прийти к выводу, что точность вычисления сплайн-функции падает после вычисления половины узлов функции. Это связано, в первую очередь, с тем, что сплайн-функция имеет кубическую степень полинома. У полинома Лагранжа, наоборот вычисления наиболее точные из-за 6-ой степени полинома.

Вычисление интеграла с применением программы QUANC8

Результаты вычисления интеграла представлены ниже:

$f = 6.332953930;$

$errest = 0.004659366;$

$nofun = 3905;$

$flag = 76.429$

Выводы по результатам вычисления

Проведенный анализ результатов показывает, что нельзя сделать определенные выводы о надежности полученных данных из-за наличия ненулевого значения индикатора **flag** и значительного отклонения. Большое количество промежутков (76), принятых с нарушением условия вычисления подынтегральной функции, может свидетельствовать о разрывах в функции подынтегральной или возможных шумах, вызванных вычислительной погрешностью.

Исходный код

```
1 program lab1
2
3   integer , parameter      :: M = 7, N = 19
4   integer                  :: nofun = 0, k = 0, i = 0
5   real                     :: lower = 0, upper = 0, abserr = 0, relerr
   = 0, res = 0, errest = 0, flag = 0
6   real                     :: x_k(N), B(N), C(N), D(N), xNodes(M),
fNodes(M)
7   real                     :: result(N), spline_result(N)
8
9   data xNodes /-1.000, -0.960, -0.860, -0.790, 0.220, 0.500, 0.930/
10  data fNodes /-1.000, -0.151, 0.894, 0.986, 0.895, 0.500, -0.306/
11
12  lower = 1
13  upper = 2
14
15  abserr = 0.000001
16  relerr = 0.000001
17  x_k = [(-1 + 0.1 * k, k = 1, 19)]
18
19
20  call SPLINE(M, xNodes, fNodes, B, C, D)
21
22  do i = 1, N
23     result(i) = Lagrange(M, xNodes, fNodes, x_k(i))
24     spline_result(i) = SEVAL(M, x_k(i), xNodes, fNodes, B, C, D)
25  end do
26
27  write (*, "('absError = ', f11.6, '; relError = ', f11.6)") abserr,
relerr
28  write (*, *)
29  do k = 1, N
30     write(*, "('Polynomial at = ', f22.10, '; Result = ', f22.10)")
x_k, result
31     write(*, "('Spline Polynomial at = ', f15.10, '; Spline Result =
', f15.10)") x_k, spline_result
32     write (*, *)
33  end do
34
35  call QUANC8(f, lower, upper, abserr, relerr, res, errest, nofun,
flag)
36  write (*, *)
37  write(*, "('f(tan(x) / x) = ', f11.9, '; Errest = ', f11.9, ';
Nofun = ', i4, '; Flag = ', f15.9)") res, errest, nofun, flag
38
39
40 contains
41   include 'QUANC8.FOR'
```

```

42  include 'SPLINE.FOR'
43  include 'SEVAL.FOR'
44
45  real pure function f(x)
46      real, intent(in) :: x
47      f = tan(x) / x
48  end function f
49
50  real function Lagrange(nodesAmount, xNodes, fNodes, x_ki) result(
value)
51      integer, intent(in) :: nodesAmount
52      real, intent(in) :: xNodes(nodesAmount), fNodes(
nodesAmount), x_ki
53      integer :: i, j
54      real :: term
55
56      value = 0
57      do i = 1, nodesAmount
58          term = 1
59          do j = 1, nodesAmount
60              if (i /= j) then
61                  term = term * (x_ki - xNodes(j)) / (xNodes(i) - xNodes
(j))
62              end if
63          end do
64          value = value + fNodes(i) * term
65      end do
66  end function Lagrange
67
68  end program lab1

```