Федеральное государственное автономное

образовательное учреждение высшего образования

“Национальный исследовательский университет ИТМО”

Факультет программной инженерии и компьютерной техники

Лабораторная работа №5

по вычислительной математике

Вариант №3

Выполнил:

Голиков Андрей

Группа: P32092

Принял: Рыбаков

Степан Дмитриевич

???

?

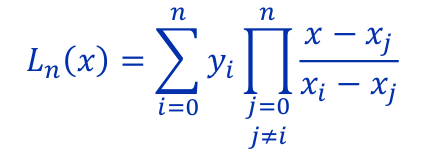


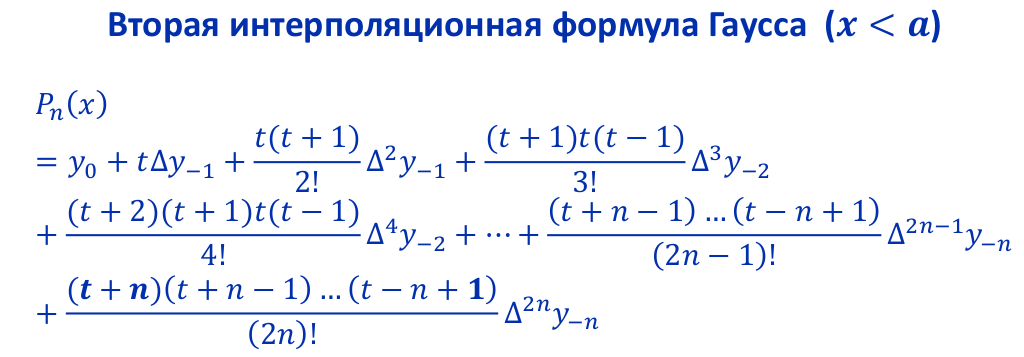
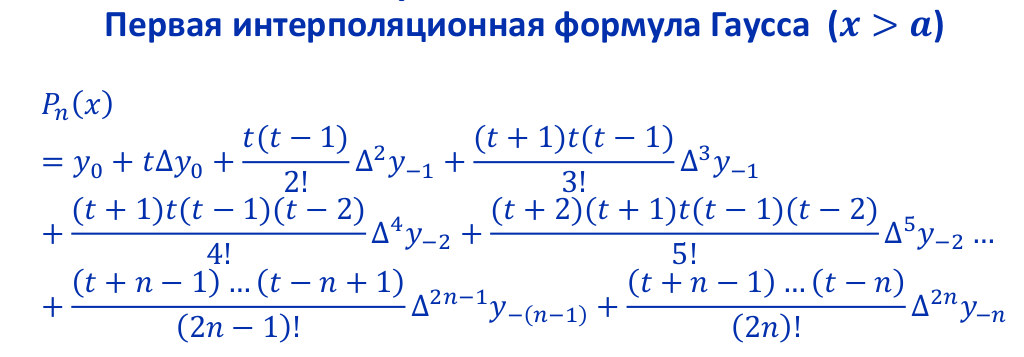
Санкт-Петербург, 2023

Цель работы: решить задачу интерполяции, найти значения функции при заданных значениях аргумента, отличных от узловых точек.

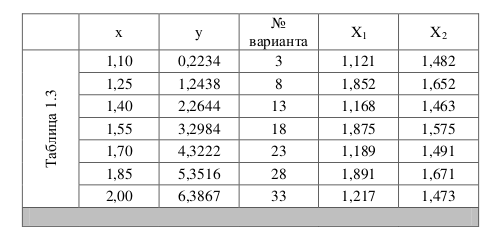
Рабочие формулы:

Многочлен Лагранжа





Вычислительная часть:

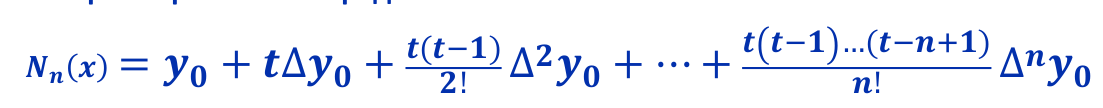


|  | y | ∆y | ∆^2y | ∆^3y | ∆^4y | ∆^5y | ∆^6y | ∆^7y |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0.2234 | 1.0204 | 0.0002 | -0.1634 | 0.3956 | 0.3071 | -3.9649 | 13.5535 |
| 1 | 1.2438 | 1.0206 | -0.1632 | 0.2322 | 0.7027 | -3.6578 | 9.5886 |  |
| 2 | 2.2644 | 0.8574 | 0.0690 | 0.9349 | -2.9551 | 5.9308 |  |  |
| 3 | 3.1218 | 0.9264 | 1.0129 | -2.0202 | 2.9759 |  |  |  |
| 4 | 4.0482 | 1.9393 | -1.0073 | 0.9557 |  |  |  |  |
| 5 | 5.9875 | 0.9320 | -0.0516 |  |  |  |  |  |
| 6 | 6.9195 | 0.9164 |  |  |  |  |  |  |
| 7 | 7.8359 |  |  |  |  |  |  |  |

X1 = 1.121 - в левой половине

t = (x-x0)/0.15 = 0.14

Ньютон



N(1.121) = 0.2234 + 0.14\*1.0204 - 0.0602\*0.0002 - 0.0373\*0.1634 - 0.0267\*0.3956 + 0.0206\*0.3071 + 0.0167\*3.9649 + 0.0140\*13.5535 = 0.4227

Гаусс

X2 = 1.482

a = 1.4

G(1.482) = 0.2644 + 0.14\*0.8574 + 0.0602\*0.1632 - 0.0229\*0.2322 + 0.0106\*0.3956 - 0.0045\*0.3071 = 2.7707

Листинг программы:

module Evaluator where

lagrange :: Points -> Func

lagrange points = weighted\_summ ys ls where

(xs, ys) = unzip points

ls = map terrible\_shit xs

terrible\_shit x = (1 / denominator) \*. numerator where

numerator = prodd $ map (\k -> \t -> if k == x then 1 else t - k) xs

denominator = product $ map (\xj -> if x == xj then 1 else xj - x) xs

gauss :: Points -> Func

gauss points x = (if x > a then firstGauss else secondGauss) spoints x where

spoints = sort points

a = fst $ spoints !! ((length spoints - 1) `div` 2)

firstGauss :: Points -> Func

firstGauss points x = (y0 .+. summ yi) new\_x where

(xs, ys) = unzip points

i = ((length xs - 1) `div` 2)

y0 = \\_ -> ys !! i

x0 = xs !! i

new\_x = (x - x0) / ((xs !! 1) - (xs !! 0))

yi = zipWith transform [0..(i-1)] pols

yy = deltas ys

transform :: Int -> Func -> Func

transform i0 p = first .+. second where

i1 = i0 + 1

i2 = 2\*i1 - 1

first :: Func

first = ((yy !! i2 !! (i - i0)) / (facts !! i2)) \*. p

second :: Func

second = ((yy !! (i2 + 1) !! (i - i1)) / (facts !! (i2 + 1))) \*. (p .\*. (\t -> t - (realToFrac i1)))

d :: [Double] -> [Double]

d list = zipWith (-) (tail list) list

deltas :: [Double] -> [[Double]]

deltas list = take (length list) $ iterate d list

facts :: [Double]

facts = scanl (\*) 1 [1..]

pols :: [Func]

pols = scanl (\p1 n -> p1 .\*. (\t -> t^2 - n^2)) id [1..]

secondGauss :: Points -> Func

secondGauss points x = (y0 .+. summ yi) new\_x where

(xs, ys) = unzip points

i = ((length xs - 1) `div` 2)

y0 = \\_ -> ys !! i

x0 = xs !! i

new\_x = (x - x0) / ((xs !! 1) - (xs !! 0))

yi = zipWith transform [1..i] pols

transform :: Int -> Func -> Func

transform i1 p = first .+. second where

i2 = 2\*i1 - 1

first :: Func

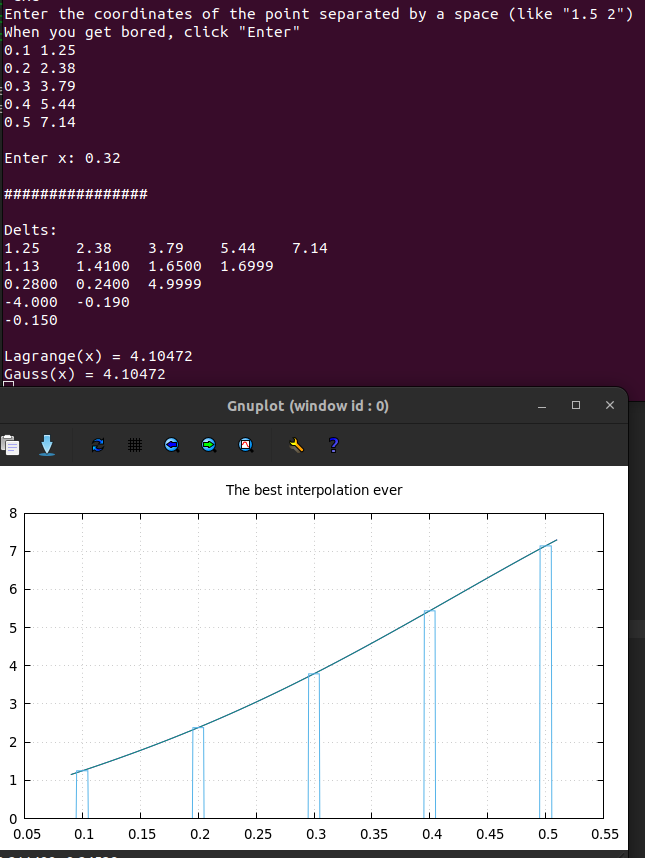
first = ((yy !! i2 !! (i - i1)) / (facts !! i2)) \*. p

second :: Func

second = ((yy !! (i2 + 1) !! (i - i1)) / (facts !! (i2 + 1))) \*. (p .\*. (\t -> t + (realToFrac i1)))

yy = deltas ys

Результаты работы программы:



Вывод: реализовал на языке Haskell интерполяцию функции, заданную поточечно. Построил многочлены Лагранжа и Гаусса. Они дают почти одинаковые значения в силу единственности интеполирующей функции. Возникающие расхождения могут иметь причину только в неточности компьютерных вычислений.