ЛАБОРАТОРНА РОБОТА №7

Тема: «Інтерполяційний багаточлен Ньютона»

Трубчанінов Андрій Сергійович

ΦIT 2-8

B-29

Код:

```
import numpy as np
import matplotlib.pyplot as plt
x = np.array([0.15, 0.16, 0.17, 0.18, 0.19, 0.20, 0.21, 0.22, 0.23, 0.24, 0.25])
y = np.array([4.4817, 4.9530, 5.4739, 6.0496, 6.6859, 7.3891, 8.1662, 9.0250, 9.9742, 11.0232, 12.1825])
\begin{tabular}{ll} \textbf{def first\_interpolation}(x,\ y,\ x0): \\ \end{tabular}
   n = len(x)
     f = np.zeros((n, n))
    f[:, 0] = y
     for j in range(1, n):
         for i in range(n - j):

f[i, j] = (f[i+1, j-1] - f[i, j-1]) / (x[i+j] - x[i])
    ans = 0
for j in range(n):
         prod = f[0, j]

for i in range(j):

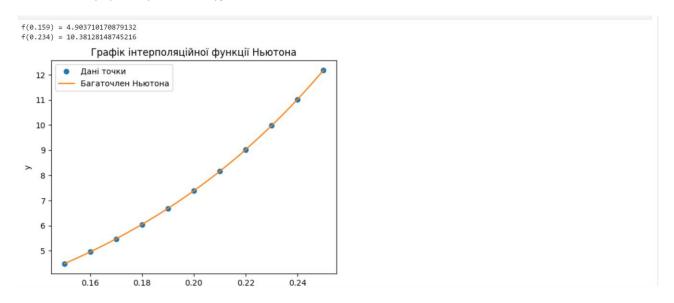
prod *= (x0 - x[i])
         ans += prod
# Друга інтерполяційна формула Ньютона
def second_interpolation(x, y, x0):
   n = len(x)
     f = np.zeros((n, n))
    f[:, 0] = y
```

```
return ans
# Друга інтерполяційна формула Ньютона
def second_interpolation(x, y, x0):
    n = len(x)
      f = np.zeros((n, n))
     f[:, 0] = y
     for j in range(1, n):
    for i in range(n - j):
        f[i, j] = (f[i+1, j-1] - f[i, j-1]) / (x[i+j] - x[i])
     ans = f[0, 0]
for j in range(1, n):
          prod = f[0, j]
for i in range(j):
    prod *= (x0 - x[i])
          ans += prod
# Обчислюємо значення функції в точках x = 0.159 та x = 0.234
x2 = 0.234
y1 = first_interpolation(x, y, x1)
y2 = second_interpolation(x, y, x2)
print(f"f({x1}) = {y1}")
print(f"f({x2}) = {y2}")
# Будуємо графік інтерполяційної функції
```

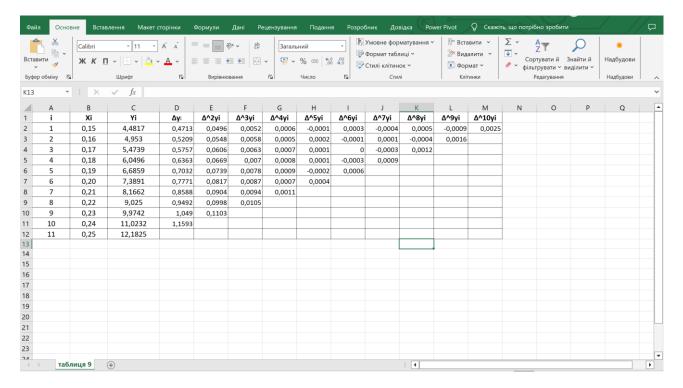
```
x2 = 0.234
y1 = first_interpolation(x, y, x1)
y2 = second_interpolation(x, y, x2)
print(f"f({x1}) = {y1}")
print(f"f({x2}) = {y2}")
# Будуємо графік інтерполяційної функції
xx = np.linspace(np.min(x), np.max(x), 100)
yy = np.zeros_like(xx)
for i in range(len(xx)):
    yy[i] = second_interpolation(x, y, xx[i])
plt.plot(x, y, 'o', label='Дані точки')
plt.plot(xx, yy, label='Багаточлен Ньютона')
plt.title("Графік інтерполяційної функції Ньютона")
plt.xlabel("x")
plt.ylabel("y")
plt.legend()
plt.show()
f(0.159) = 4.903710170879132
```

Графік інтерполяційної функції Ньютона

f(0.234) = 10.38128148745216



Excel:



Код зі скрінів:

```
import numpy as np import matplotlib.pyplot as plt
```

```
# Задані точки
x = np.array([0.15, 0.16, 0.17, 0.18, 0.19, 0.20, 0.21, 0.22, 0.23,
0.24, 0.25
y = np.array([4.4817, 4.9530, 5.4739, 6.0496, 6.6859, 7.3891,
8.1662, 9.0250, 9.9742, 11.0232, 12.1825])
# Перша інтерполяційна формула Ньютона
def first interpolation(x, y, x0):
  n = len(x)
  f = np.zeros((n, n))
  f[:, 0] = y
  for j in range(1, n):
    for i in range(n - j):
       f[i, j] = (f[i+1, j-1] - f[i, j-1]) / (x[i+j] - x[i])
  ans = 0
  for j in range(n):
    prod = f[0, i]
    for i in range(j):
       prod *= (x0 - x[i])
```

```
ans += prod
  return ans
# Друга інтерполяційна формула Ньютона
def second_interpolation(x, y, x0):
  n = len(x)
  f = np.zeros((n, n))
  f[:, 0] = y
  for j in range(1, n):
    for i in range(n - j):
       f[i, j] = (f[i+1, j-1] - f[i, j-1]) / (x[i+j] - x[i])
  ans = f[0, 0]
  for j in range(1, n):
     prod = f[0, j]
    for i in range(j):
       prod *= (x0 - x[i])
     ans += prod
  return ans
```

Обчислюємо значення функції в точках x = 0.159 та x = 0.234

```
x1 = 0.159
x2 = 0.234
y1 = first_interpolation(x, y, x1)
y2 = second_interpolation(x, y, x2)
print(f''f({x1}) = {y1}'')
print(f''f({x2}) = {y2}'')
# Будуємо графік інтерполяційної функції
xx = np.linspace(np.min(x), np.max(x), 100)
yy = np.zeros_like(xx)
for i in range(len(xx)):
  yy[i] = second_interpolation(x, y, xx[i])
plt.plot(x, y, 'o', label='Дані точки')
plt.plot(xx, yy, label='Багаточлен Ньютона')
plt.title("Графік інтерполяційної функції Ньютона")
plt.xlabel("x")
plt.ylabel("y")
plt.legend()
plt.show()
```