

Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

ЛАБОРАТОРНА РОБОТА № 3

з дисципліни «Алгоритми та методи обчислень»

на тему «Інтерполяція функцій»

ВИКОНАЛА:
студентка 2 курсу
групи ІВ-92
Бабенко В.В.
Залікова - 9201

ПЕРЕВІРИВ:
Доцент кафедри ОТ
Порєв В.М.

Київ - 2021

Хід роботи

Мета: Ознайомлення з інтерполяційними формулами Лагранжа, Ньютона, рекурентним співвідношенням Ейткена, методами оцінки похибки інтерполяції.

Завдання:

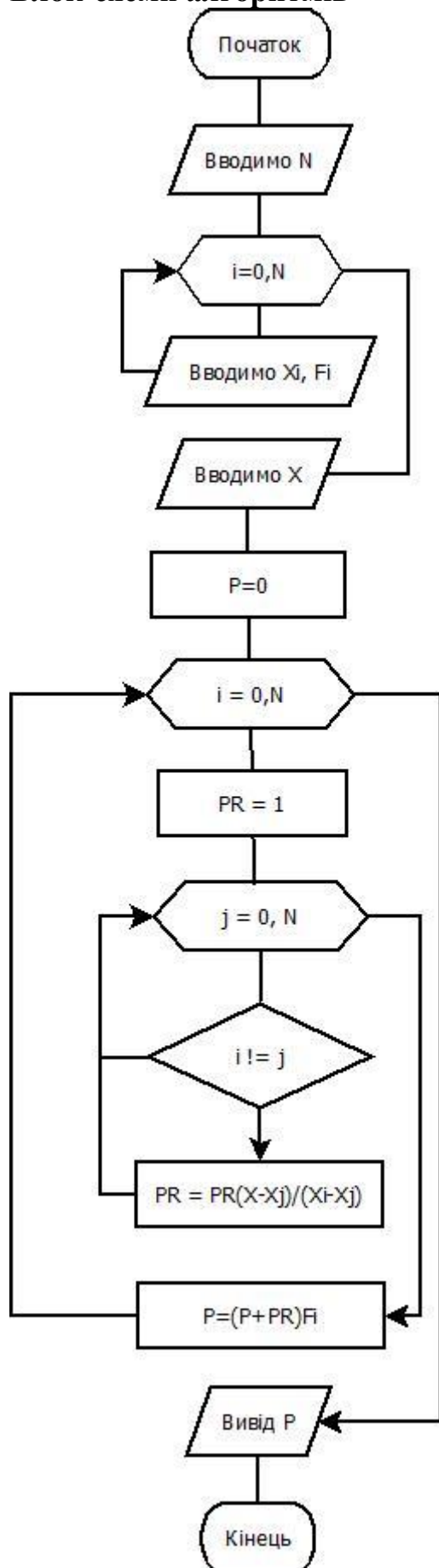
Закріплення, поглиблення і розширення знань студентів при вирішенні практичних обчислювальних завдань. Оволодіння обчислювальними методами і практичними методами оцінки похибки обчислень. Придбання умінь і навичок при програмуванні та налагодженні обчислювальних завдань на комп'ютері.

Варіанти завдання

Варіант 1

$f(x)$	$[a, b]$	формула
$\sin x^2$	$[0, 2]$	1.3

Блок-схеми алгоритмів



Задаємо кількість вершин

Вводимо вузлові точки

Точка інтерполяції

Обчислення інтерполяційного
полінома Лагранжа

На данному алгоритмі наведено блок-схему інтерполяційного полінома Лагранжа, а також текстовий опис алгоритму

Роздруківка тексту програми

```
import tkinter as tk
import math
import matplotlib.pyplot as plt
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
from tkinter import messagebox

width = 300
height = 225
# проміжок від 0 до 2
x = tuple(2*i/10 for i in range(11))
# реалізація інтерполяції методом Лагранжа
def lagr(arr_xs, arr_ys, x, n=0):
    res = 0
    for i in range(len(arr_xs)-n):
        pr = 1
        for j in range(len(arr_xs)-n):
            if i != j:
                pr *= ((x-arr_xs[j])/(arr_xs[i] - arr_xs[j]))
        res += pr*arr_ys[i]
    return res

k = 10
# проміжок від 0 до 2
xs = tuple(2 * i / (10 * k) for i in range((len(x) - 1) * k + 1))
y = []
inte = []
# функція за варіантом
for i in x:
    y.append(math.sin(i ** 2))
# додавання до інтерпольованих значень
for i in xs:
    inte.append(lagr(x, y, i))

# графік
def graph():
    global d
    d = 0
# похибка
def pohf():
    ys1 = tuple(lagr(x, y, i, n=d) for i in xs)
    print(len(ys1), len(x))
    plt.plot(xs, ys1, 'g-')
    plt.plot(x, tuple(ys1[i*k]-y[i] for i in range(len(x))), 'y-')
    fig.canvas.draw_idle()
# функція відображення інтерпольованої функції на графіку
def intLine():
    plt.plot(xs, inte, 'b-')
    global s
    s = fig.canvas.draw_idle()
# збільшення степеня
def increment():
    global d
    if d != 10:
        d += 1
        lab.config(text="Степінь: %s-%s=%s"%(len(x)-1, d, len(x)-d-1))
    else:
```

```

        messagebox.showerror("Помилка!", "Степінь не може бути меншою ніж
0")
# зменшення степеня
def decrement():
    global d
    if d != 0:
        d -= 1
        lab.config(text="Степінь: %s-%s=%s"%(len(x)-1, d, len(x)-d-1))
    else:
        messagebox.showerror("Помилка!", "Степінь не може бути більшою ніж
10")
# GUI
wind = tk.Toplevel()
wind.resizable(width=False, height=False)
plt.close()

fig = plt.figure(1)
plt.plot(x, y, 'ro', label="Заданий графік")
canvas = FigureCanvasTkAgg(fig, master=wind)
plot_widget = canvas.get_tk_widget()
tk.Button(wind, text='Інтерпольована лінія', command=intLine).grid(row=1,
column=1)
tk.Button(wind, text='Зменшити степінь многочлена',
command=increment).grid(row=2, column=0)
tk.Button(wind, text='Розрахувати похибку', command=pohf).grid(row=2,
column=1)
tk.Button(wind, text='Збільшити степінь многочлена',
command=decrement).grid(row=2, column=2)
lab = tk.Label(wind, text='Степінь: 10-0=10')
lab.grid(row=3, column=1)
plot_widget.grid(row=0, column=0, columnspan=4)

root = tk.Tk()
root.title("Лабораторна робота №3")
root.geometry("%sx%s"%(width, height))
root.resizable(width=False, height=False)
tk.Label(root, text='Лабораторна робота №3\nГрупа ІВ-92\nБабенко
Вікторія\nВаріант 1',
font=('Times New Roman', 18), background="yellow").place(x=0, y=5,
width=width, height=120)
tk.Button(root, text="Графік", font=('Roman', 15), background="lightblue",
command=graph).place(x=75, y=125, width=150)
root.config(background="blue")

root.mainloop()

```

Роздруківка результатів виконання програми

