

1. Лабораторна робота. Застосування мови Python до наближених обчислень у математичному аналізі

Мета: засвоїти можливості роботи мови Python до застосування розв'язування задач математичного аналізу.

Теоретичні відомості та методичні рекомендації

Використання Python у математичному аналізі робить процес обчислень більш ефективним та зручним, дозволяючи зосередитися на суті задачі, а не на технічних деталях реалізації алгоритмів.

У Python доступний широкий набір арифметичних операторів для виконання математичних обчислень. Основні з них дивись у таблиці 1.1.

Таблиця 1.1 – Арифметичні операції.

Операція	Опис
$x+y$	Додавання. Складає два числа.
$x-y$	Віднімання. Віднімає одне число від іншого
$x*y$	Множення. Множить два числа.
x/y	Ділення. Ділить одне число на інше, результатом є число з плаваючою комою.
$x//y$	Цілочисельне ділення. Ділить одне число на інше, повертаючи лише цілу частину результату.
$x\%y$	Остача від ділення. Повертає залишок від ділення одного числа на інше.
$x**y$	Піднесення до степеня. Підносить число до вказаного степеня.

Порядок виконання цих операцій відповідає математичним правилам: спочатку виконуються операції піднесення до степеня, потім множення, ділення, цілочисельне ділення та остача від ділення, і нарешті додавання та віднімання. Для зміни порядку виконання операцій можна використовувати круглі дужки.

Окрім основних арифметичних операцій та функцій Python надає програмісту математичну бібліотеку `math`, яка містить додаткові математичні функції та сталі. Для використання бібліотеки її необхідно імпортувати командою:

```
import math
```

Основні функції модуля `math`:

Арифметичні функції:

- `ceil(x)`: повертає найменше ціле число, більше або рівне x .
- `floor(x)`: повертає найбільше ціле число, менше або рівне x .
- `fabs(x)`: повертає абсолютне значення x .
- `factorial(x)`: повертає факторіал числа x .
- `fmod(x, y)`: повертає остачу від ділення x на y .

- `trunc(x)`: відкидає дробову частину числа x , повертаючи його цілу частину.

Степеневі та логарифмічні функції:

- `exp(x)`: повертає значення експоненти.
- `log(x[, base])`: повертає логарифм числа x за основою `base`. За замовчуванням основа дорівнює e .
- `log10(x)`: повертає десятковий (за основою 10) логарифм числа x .
- `pow(x, y)`: повертає значення x у степені y .
- `sqrt(x)`: повертає квадратний корінь з x .

Тригонометричні функції:

- `sin(x)`: повертає синус кута x , заданого в радіанах.
- `cos(x)`: повертає косинус кута x , заданого в радіанах.
- `tan(x)`: повертає тангенс кута x , заданого в радіанах.
- `asin(x)`: повертає арксинус x .
- `acos(x)`: повертає арккосинус x .
- `atan(x)`: повертає арктангенс x .
- `degrees(x)`: конвертує кут x з радіанів у градуси.
- `radians(x)`: конвертує кут x з градусів у радіани.

Гіперболічні функції:

- `sinh(x)`: повертає гіперболічний синус x .
- `cosh(x)`: повертає гіперболічний косинус x .
- `tanh(x)`: повертає гіперболічний тангенс x .
- `asinh(x)`: повертає зворотний гіперболічний синус x .
- `acosh(x)`: повертає зворотний гіперболічний косинус x .
- `atanh(x)`: повертає зворотний гіперболічний тангенс x .

Спеціальні функції:

- `erf(x)`: повертає функцію помилки для x .
- `gamma(x)`: повертає гамма-функцію для x .

Константи:

- `pi`: число π (приблизно 3.14159).
- `e`: число e (приблизно 2.71828).

Розширений набір функцій для роботи з комплексними числами містить бібліотека `cmath`. Бібліотека містить додаткові функції і сталі. Для позначення уявної частини, поруч з її значенням використовують символ `j`. Комплексний тип у Python, позначають `complex`.

Наприклад,

```
import math
```

```
# Приклад використання деяких функцій
print(math.sqrt(16))          # Виведе: 4.0
print(math.log10(100))       # Виведе: 2.0
print(math.sin(math.pi/2))    # Виведе: 1.0
```

У Python списки (list) є фундаментальним типом даних, який представляє собою впорядковану змінювану колекцію елементів довільних типів. Вони дозволяють зберігати набір значень, до яких можна звертатися за індексом, змінювати, додавати або видаляти елементи.

Створення списків.

Списки можна створювати кількома способами. Перший спосіб – це за допомогою квадратних дужок.

Власне потрібно просто перерахувати елементи списку у квадратних дужках через кому:

```
# Порожній список
empty_list = []
```

```
# Список з елементами
numbers = [1, 2, 3, 4, 5]
fruits = ['яблуко', 'банан', 'вишня']
mixed = [1, 'два', 3.0, [4, 5]]
```

Другий – за допомогою функції list():

```
# Створення списку з ітерованого об'єкта
chars = list('python') # ['p', 'y', 't', 'h', 'o', 'n']
```

Список, це послідовність, у якій всі елементи занумеровані, починаючи з 0. У Python існує механізм по-елементної роботи зі списком. Для того, щоб звернутися до відповідного елемента списку, необхідно вказати номер цього елемента у квадратних дужках. Функція len(list) повертає кількість елементів у списку.

Завдання до лабораторної роботи

1. Записати до одновимірного масиву значення функції $f(x)$ на відрізку $[a; b]$ з заданим шагом h . Значення функції округлити до 4 знаків після коми. На екран вивести перші 10 значень з отриманого масиву. Знайти найменший і найбільший елемент масиву:

$$1) f(x) = e^{-5x}(x^3 - 10x + 9) + \frac{\log_5^3 x}{x^2}, x \in [0,2; 2], h = 0,2.$$

$$2) f(x) = \sin\left(\frac{3x}{2} + \pi\right) + x^3 - 5x^2 + 7x, x \in [0,01; 1], h = 0,01.$$

$$3) f(x) = \sqrt{\frac{x^2+9x-1}{x+6}} + 7 \cos(5\pi + x), x \in [0,02; 4], h = 0,02.$$

$$4) f(x) = \ln(1 + 5x) - \frac{e^{3x+4}}{x^3+8x+9}, x \in [0,04; 2], h = 0,02.$$

$$5) f(x) = \frac{(x+5)^3}{x^5+x^2+2} - \operatorname{arctg}\left(\frac{x}{7}\right), x \in [0,01; 1], h = 0,01.$$

$$6) f(x) = e^{9x}(-x^5 + 9x - 12) + \frac{\log_8^3 2x}{x^4}, x \in [0,2; 2], h = 0,2.$$

$$7) f(x) = \cos\left(\frac{5x}{2} + \pi\right) + x^5 - 8x^2 + 3x, x \in [0,01; 1], h = 0,01.$$

8) $f(x) = \sqrt{\frac{x^2+3x-1}{x+4}} + 10 \sin(3\pi - 2x)$, $x \in [0,02; 4]$, $h = 0,02$.

9) $f(x) = \ln(1 + 10x) - \frac{e^{5x+9}}{x^3+10x+11}$, $x \in [0,04; 2]$, $h = 0,02$.

10) $f(x) = \frac{(x+2)^5}{x^3+3x^2+1} - \operatorname{arccctg}\left(\frac{x}{8}\right)$, $x \in [0,01; 1]$, $h = 0,01$.

2. Написати програму для наближеного обчислення:

1) e^{15} .

2) $\sin 2$.

3) $\cos 4$.

4) $\ln 3$.

5) $\operatorname{sh} 1$.

6) e^{17} .

7) $\sin 6$.

8) $\cos 9$.

9) $\ln 11$.

10) $\operatorname{sh} 4$.

3. Написати програми для обчислення:

1) суми всіх елементів матриці, що належать головній діагоналі;

2) кількості нульових елементів матриці;

3) суми всіх елементів матриці, що належать побічній діагоналі;

4) суми всіх недіагональних елементів матриці;

5) кількості ненульових елементів матриці.

4. Обчислити наближено визначений інтеграл

1) Від функції $f(x) = e^{-5x^2}$ на проміжку $[0; 1]$.

2) Від функції $f(x) = \cos 6x^2$ на проміжку $[0; 1]$.

3) Від функції $f(x) = \sin 3x^2$ на проміжку $[0; 1]$.

4) Від функції $f(x) = \frac{1}{\ln 2x}$ на проміжку $[0; 1]$.

5) Від функції $f(x) = \frac{\cos 7x}{x}$ на проміжку $[0; 1]$.

6) Від функції $f(x) = \frac{\sin 5x}{x}$ на проміжку $[0; 1]$.

7) Від функції $f(x) = e^{-12x^2}$ на проміжку $[0; 1]$.

8) Від функції $f(x) = \frac{1}{\ln 7x}$ на проміжку $[0; 1]$.

9) Від функції $f(x) = \frac{\cos 2x}{x}$ на проміжку $[0; 1]$.

10) Від функції $f(x) = \frac{\sin 8x}{x}$ на проміжку $[0; 1]$.

5. Обчислити наближено похідну

1) Від функції $f(x) = \frac{12x^7-8x^3-x}{x^4+x+1}$ у точці $x_0 = 2$.

2) Від функції $f(x) = \frac{\cos^5 x}{\sqrt{x}}$ у точці $x_0 = 5$.

3) Від функції $f(x) = (8x^3 - 3x) \sin 7x^2$ у точці $x_0 = -2$.

- 4) Від функції $f(x) = \ln\left(\frac{1}{x}\right)$ у точці $x_0 = 1$.
- 5) Від функції $f(x) = x \cdot \operatorname{arctg} x^3$ у точці $x_0 = \pi$.
- 6) Від функції $f(x) = x^{5x-9} + x^{20} - e^{3x}$ у точці $x_0 = -3$.
- 7) Від функції $f(x) = \operatorname{ctg}\left(\frac{1}{x^2}\right)$ у точці $x_0 = 6$.
- 8) Від функції $f(x) = e^{3 \sin x + 8 \cos x}$ у точці $x_0 = \frac{\pi}{2}$.
- 9) Від функції $f(x) = \frac{\operatorname{arctg} 4x}{16x^2 + 1}$ у точці $x_0 = 1$.
- 10) Від функції $f(x) = \log_4(9 \operatorname{arcsin} x^2)$ у точці $x_0 = 4$.

Питання для самоконтролю

1. Наведіть порядок виконання операцій у Python.
2. Перерахуйте арифметичні операції у Python.
3. Охарактеризуйте математичну бібліотеку `math`.
4. Наведіть функції модуля `math`, які відповідають за тригонометричні функції.
5. Наведіть функції модуля `math`, які відповідають за гіперболічні функції.
6. Наведіть функції модуля `math`, які відповідають за спеціальні функції і константи. Приведіть приклади.
7. Охарактеризуйте математичну бібліотеку `cmath`. Яка причина наявності двох модулів?
8. Охарактеризуйте комплексний тип у Python.
9. Наведіть способи створення списків.
10. За допомогою якої функція можна визначити кількість елементів у списку?