

Звіт
До додаткового завдання
З математичного аналізу
За темою: “Розклад у ряд Тейлора”
Тумояна Кірілла



APPLIED
SCIENCES
FACULTY.

1. Розклад у ряд Маклорена функції $1 / (1 + \sin(x))$:

$$f(x) = \frac{1}{1 + \sin(x)} = |\sin(x) = t| = \frac{1}{1 + t}$$
$$\frac{1}{1 + t} = 1 - t + t^2 - t^3 + \dots = \sum_{n=0}^{+\infty} (-1)^n t^n$$
$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots = \sum_{n=0}^{+\infty} (-1)^n \frac{x^{2n+1}}{(2n+1)!}$$
$$\frac{1}{1 + \sin(x)} = \sum_{n=0}^{+\infty} (-1)^n \left(\sum_{k=0}^{+\infty} (-1)^k \frac{x^{2k+1}}{(2k+1)!} \right)^n$$

2. Функціонал програми.

Модуль `main.py` містить 6 функцій:

- `sin_finder(amount_of_iterations, x)`

```
ans = 0
for i in range(amount_of_iterations):
    ans += ((-1) ** i) * (x ** (2 * i + 1)) / factorial(2 * i + 1)
return ans
```

Ця функція робить первинний розклад лише функції синуса у ряд Тейлора (Маклорена) в околі точки (x) та виводить значення в залежності від кількості ітерацій (`amount_of_iterations`).

- `sum_finder(amount_of_iterations, x)`

```
ans = []
for x in lst:
    y, i = 0, 0
    while i < 1000:
        y += ((-1) ** i) * (sin_finder(amount_of_iterations, x)) ** i
        i += 1
    ans.append(y)
return ans
```

Розкладає синус за повною формулою Тейлора (Маклорена) для $1 / (1 + \sin(x))$ в околі точки (x) та виводить значення в залежності від кількості ітерацій (`amount_of_iterations`).

- `graph_builder(amount_of_iterations)`

```
x = np.arange(-8, 5, 0.05)
plt.plot(x, sum_finder(amount_of_iterations, x), color="green")
plt.plot(x, (1 / (1 + np.sin(x))), color="red")
plt.ylim(-1, 15)
plt.title("Taylor for 1 / (1 + sin(x))")
plt.ylabel("y axis")
plt.xlabel("x label")
plt.show()
```

Виводить графік розкладу за рядом у кількості ітерацій (`amount_of_iterations`) та граф оригінальної функції, при цьому порівнюючи їх на проміжку $[-8; 5]$

- difference_finder(x)

```
nums, ans = [1, 5, 10, 25, 50], ""
for i in nums:
    try:
        ans += f"For the amount of iterations = {i} the taylor's number is {sum_finder(i, [x])[0]}, " \
               f"the difference is {abs(sum_finder(i, [x])[0] - (1 / (1 + sin(x))))}\n"
    except OverflowError:
        continue
return ans
```

Забезпечує порівняння значень у залежності від кількості ітерацій (1, 5, 10, 25, 50) із значенням оригінальної функції в точці (x)

- fault_finder(x, psy)

```
ans = 0
i = 1
while True:
    try:
        if abs(sum_finder(i, [x])[0] - (1 / (1 + sin(x)))) < psy:
            ans += i
            break
        else:
            i += 1
    except OverflowError:
        i += 1
return f"For the fault of {psy} the minimum amount of iterations is {ans}"
```

Знаходить за допомогою нескінченного циклу while True мінімальну кількість ітерацій, для яких значення розкладу відрізняється від значення оригінальної функції на (psy) в точці (x). Виконується для psy = 0,1; 0,001; 0,000001.

- `main(amount_of_iterations, x)`

```
print(f"The number we got is: {sum_finder(amount_of_iterations, [x])[0]}")
print(f"The real number is: {1 / (1 + sin(x))}")
print(f"The difference between them is: "
      f"{abs(sum_finder(amount_of_iterations, [x])[0] - (1 / (1 + sin(x))))}\n")
print(difference_finder(x))
print(fault_finder(x, 10 ** (-1)))
print(fault_finder(x, 10 ** (-3)))
print(fault_finder(x, 10 ** (-6)))
graph_builder(amount_of_iterations)
```

Запускає всі необхідні функції та виводить результати в консоль.

Усі функції мають виключну документацію.

Бібліотеки, застосовані в модулі:

- `math` (factorial, sin)
- `numpy`
- `matplotlib`

3. Запуск та приклад роботи програми.

Модуль може бути запущений через `if __name__ == "main"`. У результаті на екран буде виведено:

```
This is the Taylor's series for the function: 1 / (1 + sin(x))
Enter the number of iterations:
>>>
```

Після того, як користувач вводить довільну кількість ітерацій ($i < 100$), виводиться наступне:

```
This is the Taylor's series for the function: 1 / (1 + sin(x))
Enter the number of iterations:
>>> 10
Enter point (x coordinate):
>>>
```

Модуль працює швидко для порівняно невеликих значень x ($|x| < 30$).

```
This is the Taylor's series for the function: 1 / (1 + sin(x))
Enter the number of iterations:
>>> 10
Enter point (x coordinate):
>>> 3
The number we got is: 24.35054965368586
The real number is: 24.34527915938213
The difference between them is: 0.0052704943037298335

For the amount of iterations = 5 the taylor's number is 0.9177425677469911, the difference is 23.42753659163514
For the amount of iterations = 10 the taylor's number is 24.35054965368586, the difference is 0.0052704943037298335
For the amount of iterations = 25 the taylor's number is 24.345279159380482, the difference is 1.6484591469634324e-12
For the amount of iterations = 50 the taylor's number is 24.345279159380482, the difference is 1.6484591469634324e-12

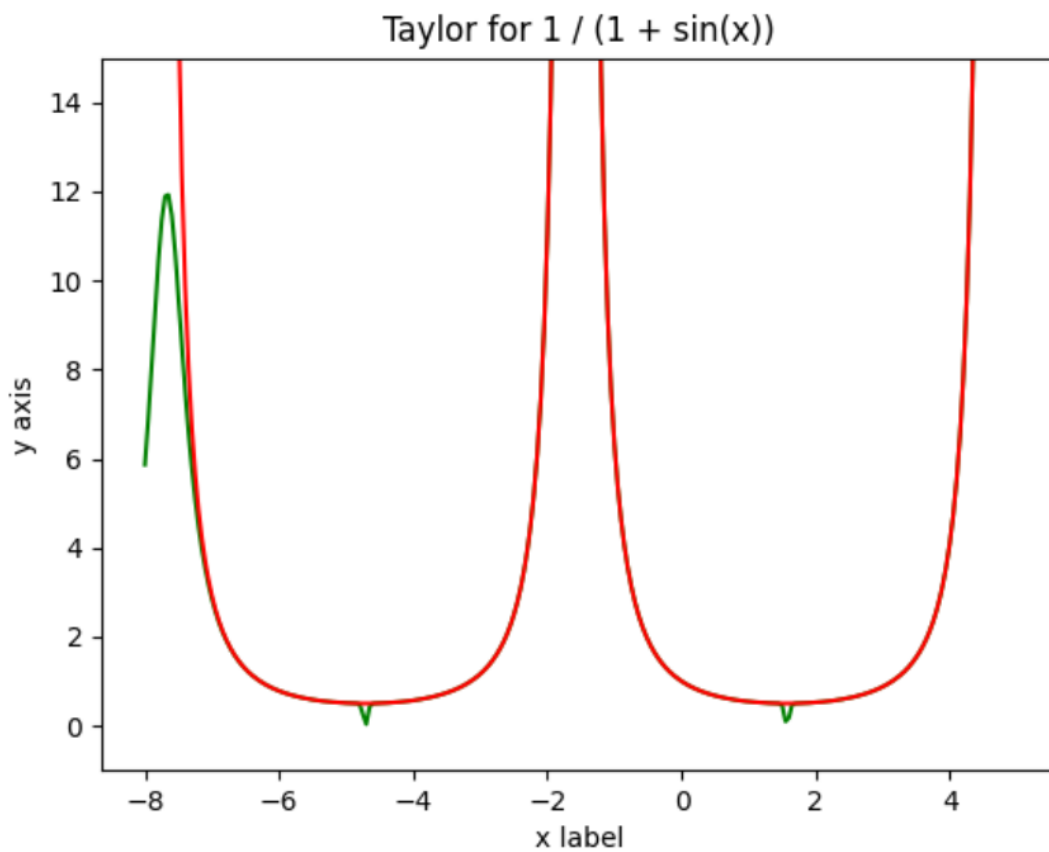
For the fault of 0.1 the minimum amount of iterations is 9
For the fault of 0.001 the minimum amount of iterations is 11
For the fault of 1e-06 the minimum amount of iterations is 13

Process finished with exit code 0
```

Після введення даних користувач отримує результат:

- значення у точці x за введеної кількості ітерацій;
- значення реальної функції в x ;
- різниця між цими значеннями;
- результат роботи функції `difference_finder(x)`;
- результат роботи функції `fault_finder(x, psy)`;
- два графіки (розкладу та реальної функції).

Нижче наведено приклад виведення графіків на екран:



Червоним намальовано графік оригінальної функції, зеленим – розкладу.

У випадку виникнення винятку користувач отримує інформацію про нього.