11/213, Tymoteusz Treppa

W zadaniu mam zadaną całkę:

I mam wyznaczyć wartości h(15°), h(30°), h(45°) i porównać je z przybliżeniem dla małych amplitud równym h(0)=π/2.

Do obliczenia numerycznego tych wartości użyję metody trapezów, która jest najprostszą metodą przybliżenia całki oznaczonej. Całka to inaczej pole powierzchni pod wykresem funkcji całkowanie. Ta metoda polega na podzieleniu obszaru granic całki na pewną ilość odcinków n o równej długości. Następnie liczymy pole trapezu na każdym takim odcinku, gdzie podstawy trapezu to wartości funkcji od początku i końca tego małego odcinka n. Wysokość to długość takiego odcinka. Na koniec sumujemy wszystkie pola małych trapezów i otrzymujemy przybliżoną wartość całki. Im większa liczba odcinków n (l kroków) tym trapezy są bardziej dopasowane do wykresu funkcji i tym większa dokładność obliczonej wartości.

Objaśnienie kodu w pythonie:

* Zaimportowanie biblioteki numpy

import numpy as np

* Zapisanie mojej funkcji do scałkowania

def funkcja(alfa\_0, alfa):

return 1 / np.sqrt(1 - (np.sin(alfa\_0 / 2))\*\*2 \* (np.sin(alfa))\*\*2)

* Zamiana stopni na radiany, ponieważ w pythonie używane są radiany zamiast stopni

alfa\_15 = np.deg2rad(15)

alfa\_30 = np.deg2rad(30)

alfa\_45 = np.deg2rad(45)

* Metoda trapezów. F – to będzie całkowana funkcja, a i b granice funkcji, n – ilość moich przedziałów, alfa\_0 – podstawiony kąt

def trapez(f, a, b, n, alfa\_0):

h = (b - a) / n

wynik = 0.5 \* (f(alfa\_0, a) + f(alfa\_0, b))

for i in range(1, n):

wynik += f(alfa\_0, a + i \* h)

wynik \*= h

return wynik

* Ustalenie liczby przedziałów, im więcej tym lepsza dokładność

l\_przedziałów = 10000

* Wywołanie mojej funkcji liczącej całkę metodą trapezów dla poszczególnych katów

h\_15 = trapez(funkcja, 0, np.pi/2, l\_przedziałów, alfa\_15)

h\_30 = trapez(funkcja, 0, np.pi/2, l\_przedziałów, alfa\_30)

h\_45 = trapez(funkcja, 0, np.pi/2, l\_przedziałów, alfa\_45)

* Wydrukowanie moich wyników oraz wartości przybliżonej h(0)

print("h(15°):", h\_15)

print("h(30°):", h\_30)

print("h(45°):", h\_45)

print("Przybliżenie dla małych amplitud h(0) =", np.pi/2)

Wartości jakie otrzymałem wynoszą:

h(15°): 1.577551660763661,

h(30°): 1.5981420021125397,

h(45°): 1.6335863074581474,

a przybliżenie dla małych amplitud wynosi h(0) = 1.5707963267948966.