

# Algorytmy Numeryczne — Zadanie 1

Jakub Ronkiewicz, 238155

Informatyka: III rok sp. Tester programista gr. 1

23 października 2017

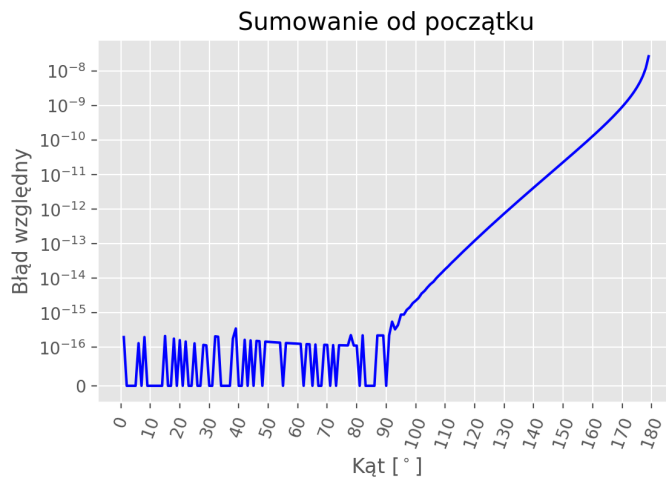
Zadanie polegało na napisaniu programu obliczającego wartość funkcji  $\sin(x)$  za pomocą trzech różnych metod:

- sumując elementy szeregu potęgowego od początku
- sumując elementy szeregu potęgowego od końca
- obliczając kolejny wyraz szeregu na podstawie poprzedniego

Wyniki moich funkcji porównuje do funkcji bibliotecznej. Do obliczenia funkcji  $\sin(x)$  korzystałem z szeregu Taylora:

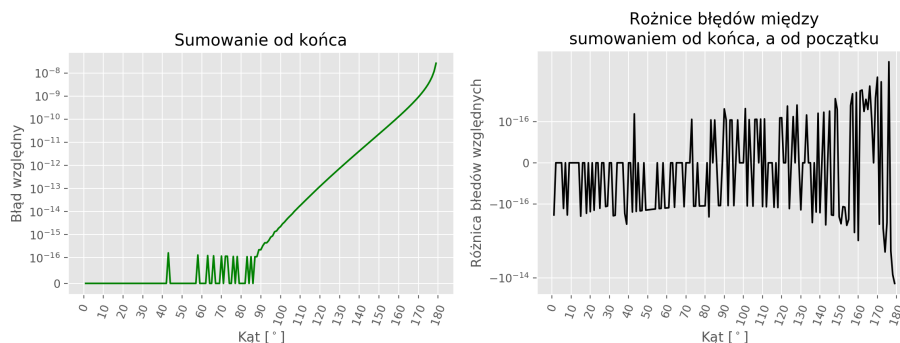
$$\sin(x) = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2 \cdot n + 1)!} \cdot x^{2 \cdot n + 1}$$

## Sumowanie elementów szeregu potęgowego od początku



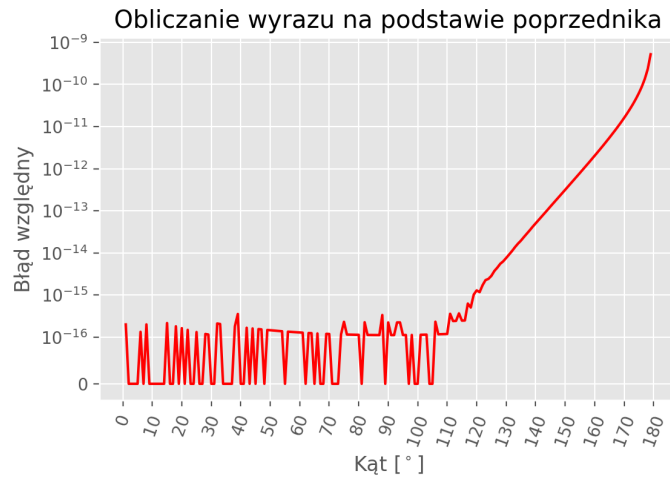
Błąd względny znacząco rośnie dopiero powyżej 90°.

## Sumowanie elementów szeregu potęgowego od końca



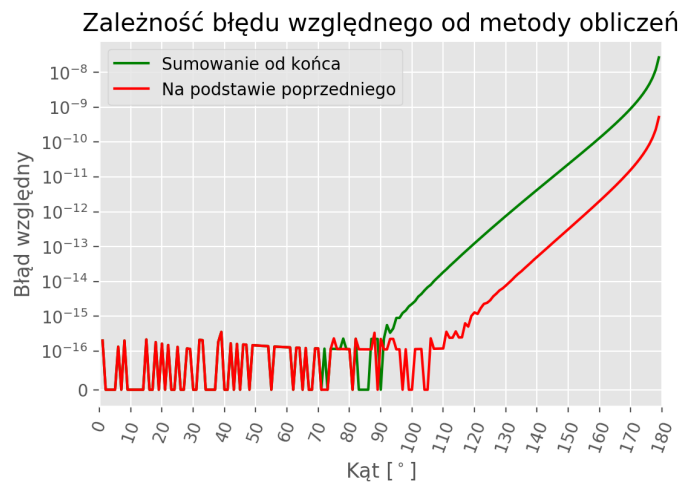
Można zauważyć, że błędy w porównaniu z *sumowaniem od początku* są mniejsze dla argumentów do 90°. Powyżej 90° błędy są podobne.

## Obliczając kolejny wyraz szeregu na podstawie poprzedniego



Błąd względny 3 metody dla argumentów do  $110^\circ$  oscyluje w podobnych granicach (rzędu  $10^{-16}$ ), dla większych argumentów błąd szybko rośnie.

## Która metoda jest najlepsza?



Na wykresie pojawiają się tylko dwie metody, ponieważ wcześniej ustaliliśmy, że metoda *sumowania od końca* jest dokładniejsza niż metoda *sumowania od początku*. Dla argumentów do  $90^\circ$  obie metody mają bardzo podobne błędy, jednak dla większych argumentów 3 metoda jest wyraźnie dokładniejsza. Konkludując metodę *obliczania kolejnego wyrazu szeregu na podstawie poprzednika* można uznać za najdokładniejszą.