

Problematyka:

- Program ma wyliczyć przybliżenie pochodnej funkcji $f(x) = \sin(x^2)$ ze wzorów na różnicę w przód i różnicę centralną:

W przód:

- $D_h f(x) = \frac{f(x+h) - f(x)}{h}$

Centralna:

- $D_h f(x) = \frac{f(x+h) - f(x-h)}{2h}$

- Następnie ma przeanalizować jak zachowują się błąd $|D_h f(x) - f'(x)|$ dla punktu $x = 0.2$ przy zmianie parametru h dla dwóch typów zmiennoprzecinkowych (float, double)

Wyjaśnienie programu:

- W programie wykorzystane są funkcje z 2 dodatkowych bibliotek (numpy, matplotlib) aby “nie wynajdywać koła od nowa”
- Funkcje stworzone przeze mnie dla potrzeby rozwiązania problemu:
 - my_fun: zwraca wartość funkcji podanej w zadaniu dla danego punktu oraz danego typu zmiennoprzecinkowego.
 - exact_my_fun_derivative: zwraca wartość dokładnej pochodnej funkcji podanej w zadaniu dla danego punktu oraz danego typu zmiennoprzecinkowego.
 - my_fun2: funkcja robi to samo co “my_fun” lecz dla innej funkcji w celu poeksperymentowania
 - exact_my_fun_derivative: funkcja robi to samo co “exact_my_fun_derivative” lecz dla innej pochodnej funkcji w celu poeksperymentowania
 - forward_derivative: funkcja oblicza różnicę w przód dla danego punktu, wartości h , funkcji oraz typu zmiennoprzecinkowego
 - central_derivative: funkcja oblicza różnicę centralną dla danego punktu, wartości h , funkcji oraz typu zmiennoprzecinkowego
 - calculate_error: funkcja oblicza błąd ze wzoru podanego w zadaniu $|D_h f(x) - f'(x)|$ dla wszystkich wartości h podanych w argumencie funkcji i zwraca je w tablicy
 - create_plot_graph: funkcja tworzy wykres z podanymi danymi używając biblioteki matplotlib
- Główna logika programu:
 - Program sprawdza argumenty wywołania programu i dla odpowiednich argumentów tworzy wykresy z wykorzystaniem wszystkich wyżej wymienionych funkcji:
 - python NUM1.py float 1:
 - dla podanych argumentów tworzy wykres dla błędów obu różnic w typie zmiennoprzecinkowym float i dla podanej w zadaniu funkcji
 - python NUM1.py double 1:
 - dla podanych argumentów tworzy wykres dla błędów obu różnic w typie zmiennoprzecinkowym double i dla podanej w zadaniu funkcji

- python NUM1.py float 2:
 - dla podanych argumentów tworzy wykres dla błędów obu różnic w typie zmiennoprzecinkowym float dla innej wymyślonej funkcji
- python NUM1.py double 2:
 - dla podanych argumentów tworzy wykres dla błędów obu różnic w typie zmiennoprzecinkowym double dla innej wymyślonej funkcji

Wyjaśnienie problematyki:

Podane wzory na przybliżenie pochodnej funkcji nie są dokładne co zależy od paru czynników. Wartość parametru h jest jednym z najważniejszych czynników i aby wynik był jak najbardziej poprawny trzeba go odpowiednio dobrać. Kolejnym z czynników jest typ liczb na jakich pracujemy. W programie analizujemy dwa typy jakimi są float i double. Typ o pojedynczej precyzji - float składa się z 32 bitów a typ o podwójnej precyzji - double ma ich 64.

Wnioski:

- Jak mogłoby się wydawać im mniejsza wartość h to nie do końca mniejszy błąd przybliżenia lecz dla odpowiedniej wartości h błąd jest najmniejszy
 - lecz istnieje moment w którym dalsze zwiększanie wartości h zwiększa coraz bardziej błąd, najczęściej jest to tuż po dobraniu takiego h , w którym błąd jest najmniejszy
- Typ zmiennych odgrywa bardzo dużą rolę, i patrząc na wyniki typ double potrafi odwzorować wartość pochodnej w bardziej dokładny sposób lecz trzeba pamiętać że zabiera więcej pamięci i jest dłużej obliczany przez procesor
- Najmniejszy błąd z daną funkcją możemy obliczyć używając typu double, różnicy centralnej oraz odpowiednio dobranej wartości h .

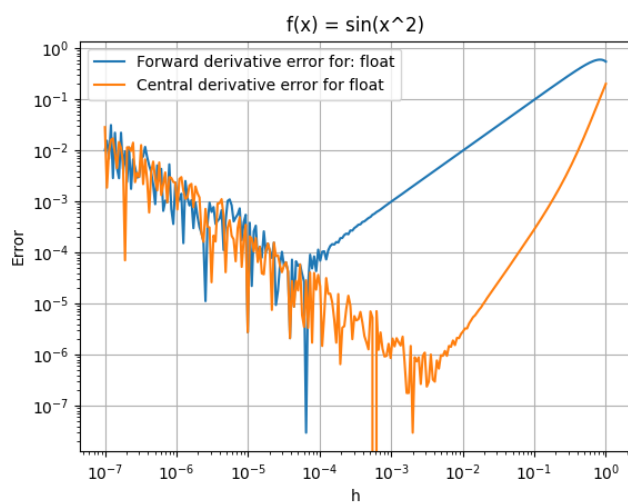
Uruchamianie programu:

- make float: uruchamia wykresy dla obu funkcji dla typu float
- make double uruchamia wykresy dla obu funkcji dla typu double

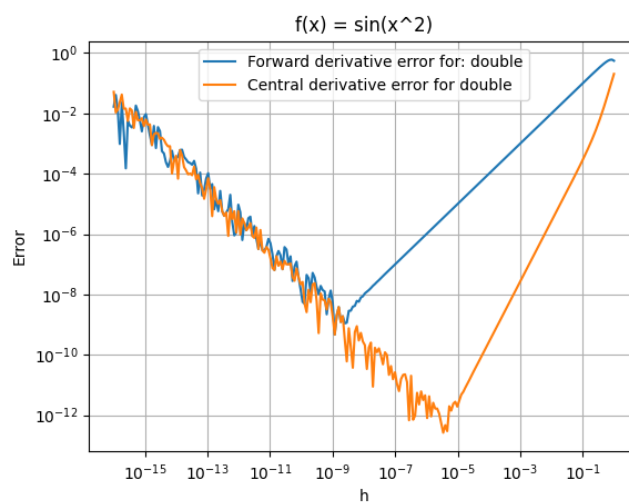
W razie problemów można zmienić w pliku make wywołanie pythona z "python" na "python3"

Wykresy powstałe dla funkcji podanej w zadaniu:

Float

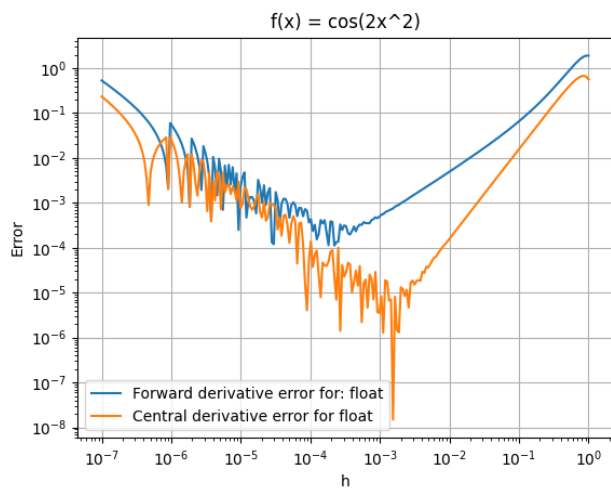


Double



Wykresy powstałe dla funkcji $f(x) = \cos(2x^2)$:

Float



Double

