

Zadanie C

FAD² - pochodne mieszane

Język programowania: C++

Symboliczne obliczanie kolejnych pochodnych danej funkcji $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ bardzo szybko prowadzi do bardzo rozbudowanych formuł i tym samym do dużych błędów numerycznych. Często też nie mamy dostępnego wzoru funkcji, lecz wartości są obliczane przez pewien podprogram. Dlatego w automatycznym różniczkowaniu podstawowe operacje arytmetyczne oraz funkcje elementarne zamienia się na odpowiednie operacje i funkcje operujące na ciągach współczynników wielomianów Taylora (lub wartości kolejnych pochodnych) zwanych **dżetami** funkcji. Z tego też powodu prezentowane tu techniki są nazywane propagacją dżetów.

Celem zadania będzie zaimplementowanie dżetów służących do obliczania wszystkich pochodnych funkcji dwóch zmiennych do rzędu drugiego w danym punkcie tzn. dla danej funkcji $f(x, y)$ i punktu (x_0, y_0) należy obliczyć

$$\left(f(x_0, y_0), \frac{\partial f(x_0, y_0)}{\partial x}, \frac{\partial f(x_0, y_0)}{\partial y}, \frac{\partial^2 f(x_0, y_0)}{\partial x^2}, \frac{\partial^2 f(x_0, y_0)}{\partial x \partial y}, \frac{\partial^2 f(x_0, y_0)}{\partial y^2} \right)$$

lub w bardziej zwięzłym zapisie

$$(f(x_0, y_0), f_x(x_0, y_0), f_y(x_0, y_0), f_{xx}(x_0, y_0), f_{xy}(x_0, y_0), f_{yy}(x_0, y_0))$$

Wejście

Zadanie polega na policzeniu wartości i wszystkich pochodnych cząstkowych do rzędu 2 dla podanej funkcji *funkcja* w M punktach $(x_1, y_1), (x_2, y_2), \dots, (x_M, y_M)$ ($0 < M < 1000000$).

Wejście programu ma następujący format:

M
 $x_1 \ y_1$
 $x_2 \ y_2$
 \vdots
 $x_M \ y_M$

Funkcja *funkcja* jest zdefiniowana w pliku **funkcja.h** jako szablon o następującym nagłówku

```
template <typename T>  
T funkcja(const T & x, const T & y);
```

Od typu T oczekuje się następujących operacji

- $+$, $-$, $*$, $/$ pomiędzy obiektami typu T ,
- $+$, $-$, $*$, $/$ pomiędzy stałą a obiektem typu T ,
- operatora negacji $-$,
- funkcji elementarnych \sin , \cos , \exp ,
- operatora przypisania,
- konstruktora domyślnego (ustawia wartość 0) i kopiującego

Wyjście

Dla każdego z punktów w osobnej linii należy wypisać oddzielone pojedynczą spacją kolejno: wartość funkcji oraz kolejno pochodne cząstkowe $f_x, f_y, f_{xx}, f_{xy}, f_{yy}$:

$f(x_1, y_1) \ f_x(x_1, y_1) \ f_y(x_1, y_1) \ f_{xx}(x_1, y_1) \ f_{xy}(x_1, y_1) \ f_{yy}(x_1, y_1)$
 \vdots
 $f(x_M, y_M) \ f_x(x_M, y_M) \ f_y(x_M, y_M) \ f_{xx}(x_M, y_M) \ f_{xy}(x_M, y_M) \ f_{yy}(x_M, y_M)$

Uwagi

Z uwagi na błędy zaokrągleń wyniki będą porównywane z pewnym dopuszczalnym błędem zarówno względnym ε_r jak i bezwzględnym ε_a (zazwyczaj około 10^{-8}). Dokładnie: jeżeli oczekiwany wynik to x a program zwróci \bar{x} , to wynik ten będzie odrzucony jeżeli $|x - \bar{x}| > \varepsilon_a$ oraz $|\frac{x - \bar{x}}{x}| > \varepsilon_r$ dla $x \neq 0$.

Przykład

Plik **funkcja.h**

```
template <typename T>
T funkcja(const T & x, const T & y){
    T wynik = sin(x*x - 2*(y+1))/exp(-y*y+cos(x*y));
    return wynik;
}
```

Dla wejścia (test0/0)

4
0.0 1.0
1.0 -1.0
-2.0 2.0
10.0 0.1

Dokładne do 15 cyfr po przecinku wyjście (obliczone symbolicznie przez Mathematicę).

$\varepsilon_r = \varepsilon_a = 10^{-8}$

0.756802495307928	0.000000000000000	2.82089223234308	-0.550484746419296
0.000000000000000	6.74275395752475		
1.33254939776314	2.83254191138382	-5.49764070691010	0.924483034856005
-5.99962825432166	18.5387068096272		
-95.4459944863016	30.2587880515848	-149.953283536275	1999.58989772411
649.252143179076	284.818881946363		
-0.23489123252023	-10.8099926844176	-0.94449708714517	91.0586053228664
-102.754369536573	-11.062545637455		

W powyższej tabelce linie zostały złamane (na wyjściu powinny być 4 linie).