

REV	DATA	ZMIANY
0.1	15.01.2023	<i>Adam Andrzej Keków (kekow@student.agh.edu.pl)</i>
0.11	20.01.2023	Całkowita zmiana koncepcji przechowywania wielomianów

SYMBOLICZNE OBLICZENIE POCHODNYCH WYRAŻENIA

Autor: Adam Andrzej Keków
Akademia Górniczo-Hutnicza

Kraków 2023

Spis treści

Lista oznaczeń:.....	3
Wstęp:	4
Wymagania systemowe:	4
Funkcjonalność:	4
Analiza Problemu:	4

Lista oznaczeń:

$\sin()$	Sinus
$\cos()$	Cosinus
$\ln()$	Logarytm naturalny (o podstawie z liczby e)
x^n	Potęgowanie „x” do potęgi „n”
*	znak mnożenia
/	znak dzielenia

Wstęp:

Projekt zakłada obliczanie pochodnych metodą symboliczną z interfejsem w postaci wiersza poleceń.

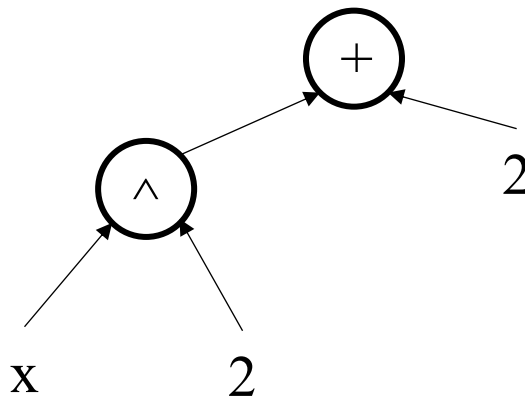
Wymagania systemowe:

Funkcjonalność:

Wejście oraz wyjście jako ciąg znaków w konsoli. Obliczanie pochodnych metodą symboliczną. Upraszczanie wyrażeń.

Analiza Problemu:

Równanie matematyczne możemy zdefiniować jako drzewo, gdzie każde działanie matematyczne tworzy jedną lub dwie gałęzie dla przykładu x^2+2 można zapisać jako:



Uwzględniając kolejność działań, można zbudować drzewo które przy zachowaniu kilku zasad da poprawną odpowiedź.

Obliczenia pochodnej zawsze zaczynają się od najdalszej gałęzi drzewa, przy czym lewa gałąź jest wcześniejsza.

Korzystając z stablicowanych wzorów łatwo zauważyć że metoda rekurencyjna jest możliwa.

Tak więc każde obliczenie pochodnej może oznaczać tylko i wyłącznie obliczenie algebraicznego wyrażenia lub pochodnej.

Przechowywanie wyniku, c plus plus nie posiada typu zdolnego przechowywać wielomiany lub funkcje matematyczne. Stworzono strukturę mogącą przechowywać wielomiany jako parę potęgi oraz liczby, resztę funkcji jako ciąg liter.

Projekt Techniczny:

Źródła:

Parser inspirowany: Bogusław Cyganek\BookCpp-master\Parser\

Find na vectorach w polynomial nie działa, wywala w jakieś dziwne nagłówki include. Operator == może być tego winny?

Odejmowanie jest do poprawy