Informatyka stosowana, studia dzienne, I st. | Semestr IV
Metody Numeryczne | 2020/2021
Prowadzący: mgr inż. Paweł Tarasiuk | Środa 14:30

Daniel Modrzejewski 229963 229963@edu.p.lodz.pl
Mateusz Srebnik 230004 230004@edu.p.lodz.pl

Zadanie 4.: Całkowanie numeryczne złożoną kwadraturą Newtona-Cotesa oraz kwadraturą Gaussa

1. Cel

Celem zadania było poznanie, zaimplementowanie i porównanie ze sobą dwóch metod całkowania numerycznego – złożonej kwadratury Newtona-Cotesa opartą na trzech węzłach (wzór Simpsona) oraz kwadratury Gaussa-Czebyszewa.

2. Wprowadzenie

Kwadratura to synonim całkowania numerycznego – metoda ta polega na przybliżeniu całki danej funkcji za pomocą sum ważonych całkowanej funkcji w wielu punktach.

Rozpatrywane w sprawozdaniu całki miały postać:

$$\int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x)dx \approx Q(f)$$

W przypadku kwadratury Gaussa-Czebyszewa całkę przybliża się za pomocą sumy:

$$Q(f) = \sum_{i=0}^{n} A_i f(x_i)$$

Gdzie:

$$A_i = \frac{\pi}{n+1}$$

Oraz:

$$x_i = cos\frac{(2\text{i}+1)\pi}{2n+2}$$

W przypadku kwadratury Newtona-Cotesa całka była przybliżana za pomocą wzoru:

$$Q(f) = w(x_0) + w(x_n) + 4\sum_{1}^{n-1}(i \; mod \; 2 + 1) * w(x_1 + i * d) + 2\sum_{1}^{n-1}(i \; mod \; 2) * w(x_1 + i * d)$$

Gdzie:

$$w(x) = \frac{1}{\sqrt{1-x^2}} f(x),$$

d – długość przedziału

Całka w naszym przypadku jest liczona jako suma całek (jednocześnie liczone są granice):

$$\int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x)dx = \int_{0}^{1} \frac{1}{\sqrt{1-x^2}} f(x)dx + \int_{-1}^{0} \frac{1}{\sqrt{1-x^2}} f(x)dx$$

Całki w kolejnych iteracjach są liczone na przedziałach (0;n) , (0;(n+1)/2) …itp.
((n;0) , ((n-1)/2;0) …itp. Dla n < 0)
Gdzie na początku n = 0.5 (n = -0.5 dla n < 0)

W przypadku Metody Gaussa całka liczona jest dla 2,3,4 i 5 węzłów. W przypadku kwadratury Newtona-Cotesa całka jest liczona dopóki różnica pomiędzy kolejnymi iteracjami nie osiągnie zakładanej dokładności.

3. Opis implementacji

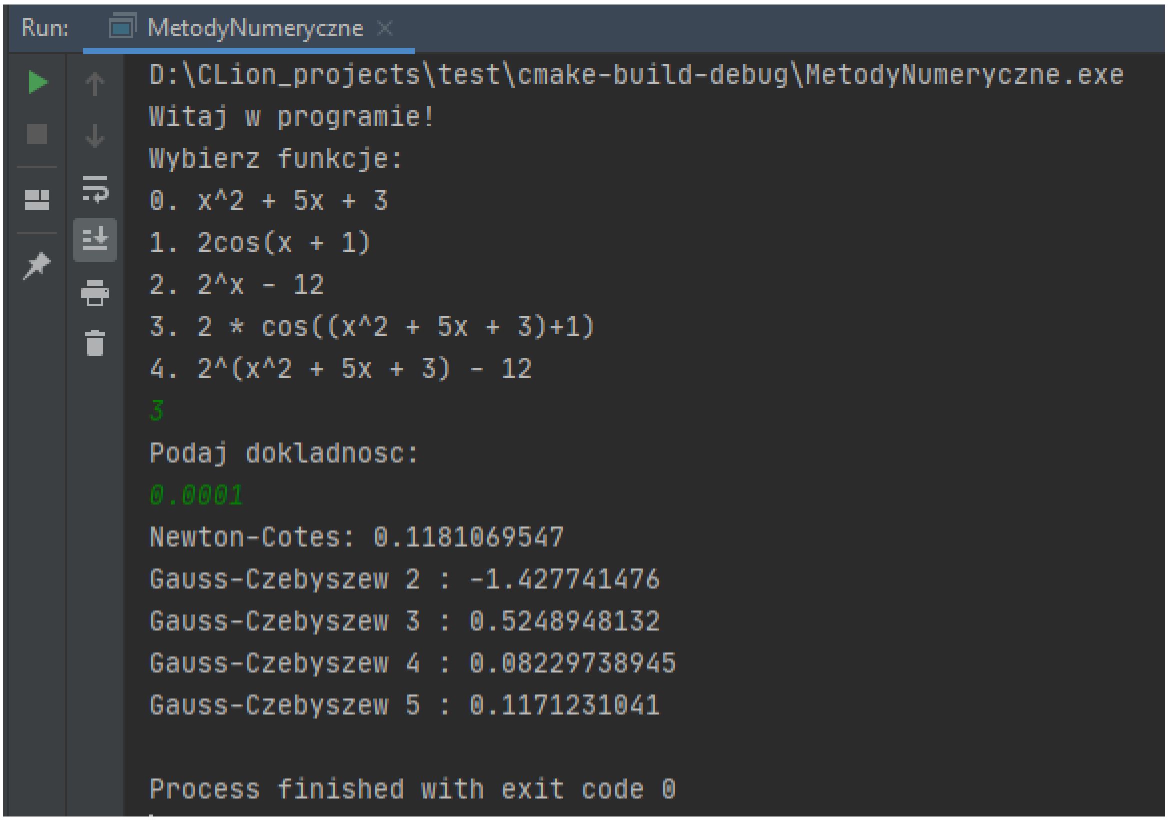
Aplikacja została utworzona w języku Cpp. Interfejs użytkownika jest interfejsem tekstowym, gdzie użytkownik wybiera jedną z 5 funkcji oraz określa dokładność dla kwadratury Newtona-Cotesa.
Na potrzeby programu została utworzona klasa abstrakcyjna „Model”, deklarująca metodę „calculate” oraz „getName”. Ta klasa abstrakcyjna jest implementowana przez klasy reprezentujące wbudowane funkcje nieliniowe, metoda „calculate” jest definiowana w taki sposób aby zwracała wartość funkcji dla podanego parametru x, natomiast metoda „getName” w taki sposób aby zwracała jaki jest wzór danej funkcji

Zostały również utworzone klasy:

- GaussCzebyszew implementująca kwadraturę Gaussa-Czebyszew,
- NewtonCotes implemenująca kwadraturę Newtona-Cotesa opartą na trzech węzłach

4. Materiały i metody

Po utworzeniu programu, został on uruchomiony i za pomocą tekstowego interfejsu użytkownika została wybrana badana funkcja oraz wartość dokładności.



5. Wynik (w tabelach obok napisu „Wynik” zapisany jest wynik uzyskany z portalu wolframalpha.com)

5.1. Funkcja $x^2 + 5x - 3$

Metoda	Wynik (10.996)
Newton-Cotes dokładność: 0.0001	10.99447108
Gauss-Czebyszew 2 węzły	10.99557429
Gauss-Czebyszew 3 węzły	10.99557429
Gauss-Czebyszew 4 węzły	10.99557429
Gauss-Czebyszew 5 węzły	10.99557429

5.2 Funkcja $2\cos(x + 1)$

Metoda	Wynik (2.59771)
Newton-Cotes dokładność: 0.0001	2.597719805
Gauss-Czebyszew 2 węzły	2.597850199
Gauss-Czebyszew 3 węzły	2.597707395
Gauss-Czebyszew 4 węzły	2.597708037
Gauss-Czebyszew 5 węzły	2.597708035

5.3 Funkcja $2^x - 12$

Metoda	Wynik (-34.1687)
Newton-Cotes dokładność: 0.0001	-34.16408296
Gauss-Czebyszew 2 węzły	-34.1687042
Gauss-Czebyszew 3 węzły	-34.16868885
Gauss-Czebyszew 4 węzły	-34.16868882
Gauss-Czebyszew 5 węzły	-34.16868882

5.4 Funkcja $2co\,s(x^2 + 5x + 4)$

Metoda	Wynik (0.118005)
Newton-Cotes dokładność: 0.0001	0.1181069547
Gauss-Czebyszew 2 węzły	-1.427741476
Gauss-Czebyszew 3 węzły	0.5248948132
Gauss-Czebyszew 4 węzły	0.08229738945
Gauss-Czebyszew 5 węzły	0.1171231041

5.5 Funkcja $2^x 2 + 5x + 3 - 12$

Metoda	Wynik (271.758)
Newton-Cotes dokładność: 0.0001	271.7571995
Gauss-Czebyszew 2 węzły	254.7720099
Gauss-Czebyszew 3 węzły	269.8695204
Gauss-Czebyszew 4 węzły	271.6028651
Gauss-Czebyszew 5 węzły	271.7479989

6. Dyskusja

Algorytm kwadratury Newtona-Cotesa uzyskuje wyniki bliskie tym z portalu wolframalpha.com. Algorytm osiąga założoną dokładność już przy stosunkowo niewielkiej dokładności (0.0001). W kwadraturze Gaussa-Czebyszewa widać dużą różnice dla różnych ilości węzłów użytych do całkowania (im więcej węzłów tym bardziej dokładny wynik). Czasem jednak (podpunkt 5.1) wartości obliczane za pomocą tej metody nie zmieniają się w zależności od ilości użytych węzłów. Jest to spowodowane szybkim osiągnięciem wartości przekraczającej dokładność zmiennych typu double. W tym samym przypadku, widać, że kwadratura Newtona-Cotesa osiągnęła zakładaną dokładność jednak wynik działania algorytmu nie był tak dokładny jak w przypadku metody Gaussa-Czebyszewa dla 2 węzłów.

7. Wnioski

- Algorytmy kwadratury Newtona-Cotesa wydaje się być bardziej niezawodny i z zakładaną dokładnością oblicza wartość podanej całki.
- Można stwierdzić, że im większa liczba węzłów tym większa dokładność metody Gaussa-Czebyszewa.

Literatura

Plik „Materiały wykładowe część III" zamieszczony na platformie wikamp

https://pl.wikipedia.org/wiki/Metody_Newtona-Cotesa

https://pl.wikipedia.org/wiki/Metoda_Simpsona

https://pl.wikipedia.org/wiki/Kwadratury_Gaussa

http://home.agh.edu.pl/~zak/downloads/4-met_num_lato.pdf

http://www.algorytm.org/procedury-numeryczne/calkowanie-numeryczne-metoda-simpsona.html

https://www.fuw.edu.pl/~jnareb/zajecia/int-gauss.pdf

http://fluid.itcmp.pwr.wroc.pl/~znmp/dydaktyka/metnum/simpson.pdf