Prezentacja i opracowanie: wielomian Wilkinsona

Łukasz Wala
AGH, Wydział Informatyki, Elektroniki i Telekomunikacji
Metody Obliczeniowe w Nauce i Technice 2021/2022

Kraków, 16 marca 2022

1 Treść zadania

Dany jest wielomian Wilkinsona $W(x) = (x-1)(x-2)(x-3)...(x-20)$.

- Przekształcić wielomian do postaci $W(x) = x^{20} + a_{19}x^{20} + ... + a_1x + x_0$. Podać wyznaczone wartości współczynników a_i.
- Dla $x = 1$ oraz $x = 20$ dokładna wartość wielomianu wynosi 0. Obliczyć warość wielomianu dla $x = 1$ oraz $x = 20$. Do wyznaczenia wartości wielomianu wykorzystać schemat Hornera oraz dowolny schemat sumacyjny.

2 Przekształcenie wielomianu

Uwaga! Kod programu w języku C użytego do rozwiązania zadania znajduje w pliku `main.c`. Program można uruchomić przez wywołanie w terminalu komendy `make run`.

Do przekształcenia wielomianu użyty został naiwny algorytm mnożenia wielomianów wymnażający jego kolejne czynniki. Z racji, że wykonywane operacje to dodawanie i dzielenie liczb całkowitch, typ danych użyty do przchowywania współczynników a_i może być zarówno stałoprzecinkowy lub zmiennoprzecinkowy, o ile pomieści owe współczynniki, np `long long int` czy `double`. Współczynniki obliczone przez załączony program są następujące

```
a0: 2432902008176640000
a1: -8752948036761600000
a2: 13803759753640704000
a3: -12870931245150988800
a4: 8037811822645051776
a5: -3599979517947607200
a6: 1206647803780373360
a7: -311333643161390640
a8: 63030812099294896
a9: -10142299865511450
```

```
a10: 1307535010540395
a11: -135585182899530
a12: 11310276995381
a13: -756111184500
a14: 40171771630
a15: -1672280820
a16: 53327946
a17: -1256850
a18: 20615
a19: -210
a20: 1
```

Zgadzają się one z wartościami rzeczywistymi.

3 Obliczenie wartości wielomianu

Wartości wielomianu dla $x = 1$ oraz $x = 20$ zostały obliczone na dwa sposoby:

- z użyciem naiwnego schematu sumacyjnego
- z użyciem schematu Hornera

Tutaj obliczenia przeprowadzone zostały dla trzech różnych typów: `double`, `long double` oraz `long long`

```
-----DOUBLE-----
WYNIK x=1.0 HORNER: 1024.000000000000000
WYNIK x=1.0 SUMA: 1184.000000000000000
WYNIK x=20.0 HORNER: -27029504000.000000000000000
WYNIK x=20.0 SUMA: 3848290697216.000000000000000

-----LONG DOUBLE-----
WYNIK x=1.0 HORNER: 0.000000000000000
WYNIK x=1.0 SUMA: 0.000000000000000
WYNIK x=20.0 HORNER: 0.000000000000000
WYNIK x=20.0 SUMA: 2147483648.000000000000000

-----LONG LONG INT-----
WYNIK x=1.0 HORNER: 0
WYNIK x=1.0 SUMA: 1739658311799032064
WYNIK x=20.0 HORNER: 0
WYNIK x=20.0 SUMA: -9223372036854775808
```

Wyniki dla `double` bardzo ewidentnie się nie zgadzają z rzeczywistością, ponieważ, mimo że ten typ pozwalał na poprawne przedstawienie współczynników, chwilowe wartości otrzymywane podczas obliczeń znacznie wykraczają poza zakres, gdzie wszystkie liczby całkowite są bezbłędnie reprezentowane.

W przypadku `long long` i `long double` dla schematu Hornera, gdzie chwilowe wartości podczas wykonywania obliczeń są mniejsze, wynik jest poprawny. Dla naiwnego sumowania, niektóre wartości prawdopodobnie nadal wkraczają w zakres, gdzie liczby całkowite nie są reprezentowane dokładnie (dla `long double`) lub overflow'ują (dla `long long`), stąd niepoprawne wyniki.

Inną ciekawą obserwacją jest to, jak bardzo na wynik wpływają zmiany współczynników. Przy zmianie a_{19} o wartość 0.000000000000001 wyglądają następująco

```
-----LONG DOUBLE-----
WYNIK x=1.0 SHIFTED HORNER: 0.000000000000000
WYNIK x=20.0 SHIFTED HORNER: 1200000000.000000000000000
```

Dla $x = 1$ różnica jest niezauważalna dla 17 miejsc dziesiętnych, natomiast dla $x = 20$ jest już bardzo duża.