

Michał Bronikowski
Deklaruję zadania numer: 1,2,3,4,5,6

1. Wiadomo, że rozwiązaniem równania kwadratowego

$$ax^2 + bx + c = 0 (a \neq 0)$$

są liczby:

(a) $\frac{-b + \sqrt{b^2 - 4ac}}{2a}$

(b) $\frac{-b - \sqrt{b^2 - 4ac}}{2a}$

Pokaż na kilku przykładach, że bezpośrednie stosowanie tych wzorów może być niebezpieczne.

Odp:

Rozwiązanie: *zad1.rb*

Dane: *zad1-dane.txt*

2. Użyj komputera do wyznaczania wartości numerycznych kolejnych elementów ciągu (x_n) . Zdefiniowanego rekurencyjnie w następujący sposób:

$$x_0 = 1$$

$$x_1 = \frac{1}{5}$$

$$x_{n+2} = \frac{26}{5}x_{n+1} - x_n$$

Skomentuj otrzymane wyniki. Czy są one wiarygodne?

Rozwiązanie:

Zastanówmy się jaki jest wzór jawny tego ciągu.

Niech

$$x_{n+2} = r^{n+2}$$

Wtedy:

$$r^{n+2} = \frac{26}{5}r^{n+1} - r^n // : r^n$$

$$r^2 = \frac{26}{5}r^1 - 1$$

$$r^2 - \frac{26}{5}r^1 + 1 = 0$$

Skorzystamy ze wzorów Viete'a:

$$\begin{aligned}r_1 + r_2 &= \frac{26}{5} \\ r_1 \times r_2 &= 1\end{aligned}$$

Więc $r_1 = \frac{1}{5}$ a $r_2 = 5$.

Przedstawmy naszą zależność rekurencyjną dla 2 i 3 elementu naszego ciągu w postaci równania:

$$x_2 = \frac{1}{25} = A \times \left(\frac{1}{5}\right)^2 + B \times 5^2$$

$$x_3 = \frac{1}{125} = A \times \left(\frac{1}{5}\right)^3 + B \times 5^3$$

Wnioskujemy z tego, że:

$$A = 1$$

$$B = 0$$

Mamy więc doczynienia z ciągiem o postaci:

$x_n = \left(\frac{1}{5}\right)^n$ Można zauważyć, że ciąg ten jest malejący. Przejdźmy do wyznaczenia wartości tego ciągu przy użyciu komputera.

Plik: *zad2.rb*

Po wypisaniu 30 pierwszych elementów tego ciągu mogę stwierdzić, że otrzymane przeze mnie wyniki nie są wiarygodne, ponieważ przekazywane na wyjście elementy nie są w porządku malejącym, a wręcz w pewnym momencie zaczynają rosnąć, a wykazałem, że ciąg powinien być malejący.

3. Wykorzystując własności szeregów naprzemiennych, ustal ilu teoretycznie wyrazów szeregu:

$$\pi = 4 \sum_{k=0}^{\infty} \frac{(-1)^k}{2k+1}$$

Należy użyć do obliczenia wartości π z błędem mniejszym niż 10^{-7}

Z kryterium **Leibniza**

Jeśli ciąg a_n jest malejący i zbieżny do zera to szereg $\sum_{n=1}^{\infty} (-1)^n a_n$ jest malejący. Nasz ciąg a_n jest postaci:

$$a_k = \frac{1}{2k+1}$$

- (a) Czy ciąg jest zbieżny

$$\lim_{k \rightarrow \infty} \frac{1}{2k+1} = 0$$

Ciąg jest więc zbieżny.

- (b) Czy ciąg jest malejący?

$$\frac{1}{2k+1} < \frac{1}{2k+2}$$

Ciąg jest malejący.

Skoro wiem, że szereg jest zbieżny skorzystam z własności:

$$|S_{k+1} - S_k| = a_{k+1}$$

Zależy nam na tym, aby błąd był mniejszy od 10^{-7} więc:

$$|S_{k+1} - S_k| = a_{k+1} < 10^{-7}$$

Dalej:

$$\frac{1}{2(k+1)+1} < 10^{-7} // \times (2k+1)$$

$$1 < 10^{-7} \times (2k+3)$$

$$k = 5000000 - 3$$

$$k = 4999997$$

Teraz wykonałem odpowiedni eksperyment z wykorzystaniem komputera.

Źródło: *zad3.rb*, *testzad3.rb*

Wynika z niego, że dla $k = 4999997$ program nie wyznacza π z błędem mniejszym niż 10^{-7} . Błąd jest mniejszy od 10^{-7} dla $k = 5000000$.

4. Wykorzystując własności szeregów naprzemiennych, sprawdź, że do obliczenia $\ln 2$ z błędem mniejszym niż $\frac{1}{2} \times 10^{-6}$ trzeba użyć ok. 2 milionów wyrazów szeregu:

$$\ln x = \sum_{k=1}^{\infty} (-1)^{k-1} \frac{(x-1)^k}{k}$$

Z kryterium **Leibniza**

Jeśli ciąg a_n jest malejący i zbieżny do zera to szereg $\sum_{n=1}^{\infty} (-1)^n a_n$ jest malejący. Nasz ciąg a_n i dla $x = 2$ jest postaci:

$$a_k = \frac{1}{k}$$

- (a) Czy ciąg jest zbieżny

$$\lim_{k \rightarrow \infty} \frac{1}{k} = 0$$

Ciąg jest więc zbieżny.

- (b) Czy ciąg jest malejący?

$$\frac{1}{k} < \frac{1}{k+1}$$

Ciąg jest malejący.

Skoro wiem, że szereg jest zbieżny skorzystam z własności:

$$|S_{k+1} - S_k| = a_{k+1}$$

Zależy nam na tym, aby błąd był mniejszy od $\frac{1}{2} \times 10^{-6}$ więc:

$$|S_{k+1} - S_k| = a_{k+1} < \frac{1}{2} \times 10^{-6}$$

Dalej

$$\frac{1}{k+1} < \frac{1}{2} \times 10^{-6} // \times 2$$

$$\frac{2}{k+1} < 10^{-6}$$

$$k+1 = 2 \times 10^6$$

$$k = 2 \times 10^6 - 1$$

Dalej wykaż, że wykorzystanie prostego związku

$$\ln 2 = \ln\left[e \times \frac{2}{e}\right]$$

może znacznie przyspieszyć obliczenia.

Odp:

$$\ln[e \times \frac{2}{e}] = \ln e + \ln \frac{2}{e} = 1 + \ln \frac{2}{e}$$

Sprawdźmy teraz ilu elementów ciągu musimy użyć aby obliczyć $\ln \frac{2}{e}$ z błędem mniejszym niż $\frac{1}{2} \times 10^{-6}$. Podstawmy do naszego szeregu za $x = \frac{2}{e}$. Otrzymujemy:

$$\ln x = \sum_{k=1}^{\infty} (-1)^{k-1} \frac{(\frac{2}{e} - 1)^k}{k}$$

Dalej:

$$a_k = \frac{(\frac{2}{e} - 1)^k}{k}$$

Więc:

$$\frac{(\frac{2}{e} - 1)^k}{k} < \frac{1}{2} \times 10^{-6} // \times 2$$

$$\frac{2(\frac{2}{e} - 1)^k}{k} < 10^{-6}$$

Jedynе sensownе wytłumaczenie jakie przychodzi mi do głowy jest takie, że w porównaniu do ciągu $\frac{2}{k}$ ten maleje szybciej ponieważ w liczniku mamy $2(\frac{2}{e} - 1)^k$, które dla nieparzystych k jest ujemne.

- Wykorzystując pomysł z poprzedniego zadania zaproponuj szybki algorytm do wyznaczania \ln bardzo dużych liczb.

```

1 def calc(x,k)
2   sum = 0
3   for i in 1..k
4     sum+=(((−1) ** (i−1)) * (((x−1) ** i) / i))
5   end
6   sum
7 end
8
9 def ln x
10  if x > 5
11    temp = x / Math::E
12    return 1 + 1 + ln(temp/Math::E)
13  else
14    return 1 + calc((x / Math::E),30)
15  end
16 end

```

Listing 1: Algorytm do wyznaczania logarytmu naturalnego

Opis:

Na początku sprawdzam czy x jest większy od 5, aby podczas dzielenia przez e wynik nie był większy od 2. Następnie wyciągam z tej liczby e i do wyniku dodaje 1 i rekurencyjnie wywołuję funkcję \ln z tym co mi zostało po wyciągnięciu e . Jeżeli argument funkcji \ln jest mniejszy równy 5 to postępuję zgodnie z pomysłem z poprzedniego zadania.

Test: *testzad5.rb*

6. W języku **PWO ++** funkcja **ATAN** oblicza \arctg z dużą dokładnością, ale tylko wtedy gdy $|x| \leq 1$. Zaproponuj algorytm wyznaczający w języku **PWO ++** \arctg z dużą dokładnością dla $|x| > 1$.
7. Wykorzystując pomysł z poprzedniego zadania zaproponuj szybki algorytm do wyznaczania \ln bardzo dużych liczb.

```
1 def ATan(x)
2   return Math.atan(x) if x.abs <= 1
3   print "Wywołano funkcja ATan z |x| > 1"
4 end
5
6 def FATan(x)
7   if x < 0
8     a = - (Math::PI / 2.0)
9     return a - ATan(1.0/x)
10  else
11    a = (Math::PI / 2.0)
12    return a - ATan(1.0/x)
13  end
14 end
```

Listing 2: Kod w Ruby

Korzystam z faktu, że gdy $x \geq 0$ to:

$$\arctg(x) + \arctg\left(\frac{x}{2}\right) = \frac{\pi}{2}$$

A gdy $x < 0$ to:

$$\arctg(x) + \arctg\left(\frac{x}{2}\right) = -\frac{\pi}{2}$$