ANL L3

19 October, 2023 11:35

L3.1. $\boxed{\textbf{Włącz komputer!}}\boxed{\text{2 punkty}}$ Dla jakich wartości x obliczanie wartości wyrażeń

a) $(x^3 + \sqrt{x^6 + 2023^2})^{-1}$, **b)** $\log_2 x - 2$, **c)** $x^{-3}(\pi/2 - x - \operatorname{arctg}(x))$

może wiązać się z utratą cyfr znaczących wyniku? Zaproponuj sposoby obliczenia wyniku dokładniejszego. Pokaż, że sposoby te **działają w praktyce**.

L3.2. $\boxed{\textbf{Włącz komputer!}}\boxed{\text{1 punkt}}$ Podaj (w miarę) bezpieczny numerycznie **algorytm** obliczania zer równania kwadratowego $ax^2 + bx + c = 0$ $(a \neq 0)$. **Przeprowadź testy** dla odpowiednio dobranych wartości a, b i c pokazujące, że Twój algorytm jest lepszy od *metody szkolnej* bazującej jedynie na dobrze znanych wzorach $x_{1,2} = (-b \pm \sqrt{b^2 - 4ac})/(2a)$.

L3.3. $\boxed{\text{1 punkt}}$ Wyprowadź wzór na wskaźnik uwarunkowania zadania obliczania wartości funkcji f w punkcie x.

L3.4. $\boxed{\text{2 punkty}}$ Sprawdź dla jakich wartości x zadanie obliczania wartości funkcji f jest źle uwarunkowane, jeśli:

a) $f(x) = (x + 2023)^7$, **b)** $f(x) = \cos(3x)$, **c)** $f(x) = (1 + x^6)^{-1}$.

L3.5. $\boxed{\text{2 punkty}}$ Załóżmy, że dla każdego $x \in X_{fl}$ zachodzi $\mathrm{fl}(\mathrm{tg}(x)) = \mathrm{tg}(x)(1 + \varepsilon_x)$, gdzie $|\varepsilon_x| \leq 2^{-t}$, natomiast t oznacza liczbę bitów przeznaczoną na zapamiętanie mantysy. Niech dane będą liczby maszynowe y_1, y_2, y_3, y_4 oraz taka liczba maszynowa x, że $x \cdot 2^{-8}$ też jest liczbą maszynową. Sprawdź czy poniższy algorytm obiczania wartości wyrażenia $\sum_{i=1}^{4} y_i \mathrm{tg}(4^{-i}x)$ jest numerycznie poprawny:

```
S:=0;

 for i from 1 to 4
   do
     S:=S+y[i]*tg(4^(-i)*x)
   od;

Return(S)
```

L3.6. $\boxed{\text{1 punkt}}$ Sprawdź czy następujący algorytm obliczania wartości wyrażenia $w(x) := x + 4x^{-1}$ $(x \neq 0)$ jest algorytmem numerycznie poprawnym:

```
u:=x;
v:=4/x;

Return(u+v)
```

W rozważaniach przyjmij, że x jest liczbą maszynową.

L3.7. $\boxed{\text{2 punkty}}$ Zbadaj czy podany niżej algorytm wyznaczania iloczynu liczb maszynowych $x_1, x_2, \ldots, x_n$ (zakładamy zatem, że $\mathrm{rd}(x_k) = x_k$, $1 \leq k \leq n$) jest algorytmem numerycznie poprawnym.

```
I:=x[n];

 for k=n-1 downto 1
   do
     I:=I*x[k]
   end;

return(I)
```

L3

zad.	1	2	3	4	5	6	7	all
pkt.			1	2				
max pkt.	2	1	1	2	2	1	2	10