

Rekurencje liniowe niejednorodne

Stefan Sokołowski

Elbląg, 29 III 2025

Zadanie 1:

Ciąg $(a_n)_{n \in \mathbb{N}}$ posiada następującą definicję rekurencyjną:

$$a_n \stackrel{\text{def}}{=} \begin{cases} -3 & \text{dla } n = 0 \\ 0 & \text{dla } n = 1 \\ 5a_{n-1} - 6a_{n-2} + 3^n & \text{dla } n \geq 2 \end{cases}$$

Podać nierekurencyjny wzór, uzależniający a_n od n .

Rekurencja **nie**jednorodna:

$$a_n = \sum_{i=1}^k c_i \cdot a_{n-i} + \underbrace{f(n)}_{\text{wyraz wolny rekurencji}} \quad (1)$$

Założmy, że $a_n^{(o)}$ jest ogólnym rozwiązaniem odpowiadającej rekurencji jednorodnej. Jak szukać rozwiązania **szczególnego** $a_n^{(s)}$ rekurencji **nie**jednorodnej (1)?

- **Przypadek, gdy $f(n)$ jest wielomianem** stopnia r :

– jeśli $a_n^{(o)}$ jest wielomianem stopnia p , to

$$a_n^{(s)} = n^{p+1} \cdot \left(\sum_{i=0}^r d_i \cdot n^i \right)$$

– jeśli $a_n^{(o)}$ nie jest wielomianem, to

$$a_n^{(s)} = \sum_{i=0}^r d_i \cdot n^i$$

• **Przypadek, gdy $f(n) = A \cdot \alpha^n$**

- jeśli α jest p -krotnym pierwiastkiem wielomianu charakterystycznego, to

$$a_n^{(s)} = B \cdot n^p \cdot \alpha^n$$

- jeśli α **nie** jest pierwiastkiem wielomianu charakterystycznego, to

$$a_n^{(s)} = B \cdot \alpha^n$$

		rozwiązanie ogólne		
		$a_n^{(o)}$ nie jest wielomianem	$a_n^{(o)}$ jest wielomianem stopnia s	α jest p -krotnym pierwiastkiem wiel. char.
wyraz wolny	$f(n)$ jest wielomianem stopnia r	$\sum_{i=0}^r g_i n^i$	$n^{s+1} \cdot \sum_{i=0}^r g_i n^i$	
	$f(n)$ jest fcją wykładn. $A \cdot \alpha^n$			$B \cdot n^p \cdot \alpha^n$

Zadanie 2:

Ciąg $(a_n)_{n \in \mathbb{N}}$ posiada następującą definicję rekurencyjną:

$$a_n \stackrel{\text{def}}{=} \begin{cases} -4 & \text{dla } n = 0 \\ 6 & \text{dla } n = 1 \\ 5a_{n-1} - 6a_{n-2} + 4^n & \text{dla } n \geq 2 \end{cases}$$

Podać nierekurencyjny wzór, uzależniający a_n od n .

Zadanie 3:

Ciąg $(a_n)_{n \in \mathbb{N}}$ posiada następującą definicję rekurencyjną:

$$a_n \stackrel{\text{def}}{=} \begin{cases} 0 & \text{dla } n = 0 \\ 1 & \text{dla } n = 1 \\ 4a_{n-1} - 4a_{n-2} + 2n & \text{dla } n \geq 2 \end{cases}$$

Podać nierekurencyjny wzór, uzależniający a_n od n .

Zadanie 4:

Ciąg $(a_n)_{n \in \mathbb{N}}$ posiada następującą definicję rekurencyjną:

$$a_n \stackrel{\text{def}}{=} \begin{cases} 0 & \text{dla } n = 0 \\ 1 & \text{dla } n = 1 \\ 1 & \text{dla } n = 2 \\ 3a_{n-1} - 3a_{n-2} + a_{n-3} + 2 & \text{dla } n \geq 3 \end{cases}$$

Podać nierekurencyjny wzór, uzależniający a_n od n .

Zadanie 5:

Ciąg $(a_n)_{n \in \mathbb{N}}$ posiada następującą definicję rekurencyjną:

$$a_n \stackrel{\text{def}}{=} \begin{cases} 0 & \text{dla } n = 0 \\ 2/3 & \text{dla } n = 1 \\ 2a_{n-1} - a_{n-2} + n & \text{dla } n \geq 2 \end{cases} \quad (2)$$

Podać nierekurencyjny wzór, uzależniający a_n od n .

Zadanie 6 (domowe na punkty):

Poniższy fragment programu w C sortuje (czyli: porządkuje wg „wzrostu”) tablicę liczb rzeczywistych `tab[0 .. dlug-1]` :

```
void insert_sort(int dlug, double tab[ ]) {
    if (dlug >= 2) {
        insert_sort(dlug-1, tab);
        dlug--;
        while (dlug >= 1 && tab[dlug-1] > tab[dlug]) {
            zamień(tab, dlug-1, dlug); dlug--;
        }
    }
}
```

Ile zamian wykonuje ten program (w najgorszym przypadku) dla ciągu o danej długości? Proszę sformułować odpowiednią zależność rekurencyjną do tego fragmentu programu, uzasadnić ją, a następnie rozwiązać.

Uwaga:

Czas na przysłanie (na s.sokolowski@ans-elblag.pl): 2 tygodnie od zajęć z lab.5 — potem już nie przyjmuję rozwiązań. Liczba punktów zależy od oryginalności i elegancji rozwiązania.
