

Lista zadań z matematycznych podstaw informatyki nr 3.

(zadania 1 – 4 to przypomnienie zadań z list poprzednich)

Zad. 1. Pokaż, że jeżeli funkcje g , oraz h są pierwotnie rekurencyjne, to także funkcja f zdefiniowana równościami

$$f(\vec{x}, 0) = g(\vec{x}) \quad \text{oraz} \quad f(\vec{x}, n) = h(f(\vec{x}, (n)_0), f(\vec{x}, (n)_1), n, \vec{x}) \quad \text{dla } n > 0$$

jest pierwotnie rekurencyjna.

Zad. 2. Pokaż, że jeżeli funkcja g jest pierwotnie rekurencyjna, to także funkcja f zdefiniowana równościami

$$f(0, x) = g(x) \quad \text{oraz} \quad f(n, x) = f((n)_0, f((n)_1, x)) \quad \text{dla } n > 0$$

jest pierwotnie rekurencyjna.

Zad. 3. Rozważmy następujący algorytm:

Dane: $m \geq 1$ oraz $n \geq 1$;

var L, i, p : **positive integer**, T : **array** $[1..m]$ **of non_negative integer**;

function $next(var p)$: **boolean**;

begin $p := 1$;

while $p \leq m$ **and** $T[p] = 0$ **do** $p := p + 1$;

$next := p \leq m$ **end**;

begin

$L := n$; **for** $i := 1$ **to** $m - 1$ **do** $T[i] := 0$; $T[m] := 1$;

while $next(p)$ **do**

for $i := 1$ **to** $p - 1$ **do** $T[i] := L$;

$T[p] := T[p] - 1$; $L := L + 1$

$write(L)$

end.

Dla jakich m i n podany algorytm kończy pracę? Jeżeli coś liczy, to oceń jego złożoność rozumianą na przykład jako liczba wykonan głównej pętli **while**.

Zad. 4. Przyjmijmy, że

$$U(n, a, x) = \begin{cases} a & \text{jeżeli } (n)_0 = 0, \\ a \cdot x + (n)_1 & \text{jeżeli } (n)_0 = 1, \\ U((n)_2, U((n)_1, a, x), x) & \text{jeżeli } (n)_0 > 1. \end{cases}$$

Udowodnij, że

- 1) U jest funkcją pierwotnie rekurencyjną,
- 2) dla dowolnego wielomianu w o współczynnikach naturalnych istnieją n i a takie, że

$$U(n, a, x) = w(x)$$

dla wszystkich x ,

- 3) pewna funkcja pierwotnie rekurencyjna nie jest wielomianem o współczynnikach naturalnych i zrób to metodą przekątniową.

Zad. 5. Udowodnij, że wszystkie formuły postaci

$$\neg\neg\varphi \Rightarrow \varphi, \varphi \Rightarrow \neg\neg\varphi \text{ oraz } \neg\neg\varphi \Leftrightarrow \varphi$$

są prawami logicznymi.

Zad. 6. Udowodnij w rozważanym systemie podstawowe własności koniunkcji, a więc prawo $\alpha \wedge \beta \Rightarrow \alpha$, możliwość dołączania koniunkcji $\alpha \wedge \beta$ do dowodu w którym występują już formuły α i β , itp.

Zad. 7. Udowodnij wybrane prawa kontrapozycji.

Zad. 8. Udowodnij prawo $\exists x \forall y \varphi \Rightarrow \forall y \exists x \varphi$.

Zad. 9. (Twierdzenie o stałej.) Przypuśćmy, że stała c nie występuje w formule φ , ani w aksjomatach teorii T . Jeżeli formuła $\varphi[x \leftarrow c]$ jest twierdzeniem teorii T , to w teorii T można dowieść także formuły $\forall x \varphi$ oraz φ . Co więcej, można to zrobić nie używając w tych dowodach stałej c (formuły te można dowieść w języku bez stałej c).

Zad. 10. (Twierdzenie o poprawności dla rachunku zdań.) Przypuśćmy, że posługujemy się językiem rzędu zero, w którym są tylko 0 argumentowe symbole relacyjne (a więc zmienne zdaniowe) i w którym nie ma sensu posługiwać się kwantyfikatorami. Pokaż, że każde prawo logiczne zapisane bez użycia kwantyfikatorów jest tautologią rachunku zdań.