

PODSTAWY PROGRAMOWANIA

I rok Automatyka i Robotyka Eka PWr

Ćwiczenia – Zestaw 3

Zakres materiału

Algorytmy, ich zapis, ręczna symulacja, analiza złożoności obliczeniowej.

Zadania

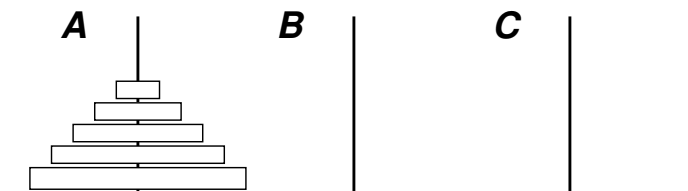
1. Zapisać w postaci diagramu blokowego lub pseudokodu algorytm wyznaczający największy wspólny dzielnik dwóch liczb. Przeprowadzić ręczną symulację¹ algorytmu dla przykładowych danych.
Wskazówka: Jednym z rozwiązań powyższego problemu jest algorytm Euklidesa.
2. Przeformułować przedstawiony na wykładzie algorytm wyznaczania ciągu liczb Fibonacciego w postaci rekurencyjnej do postaci iteracyjnej. Przeprowadzić ręczną symulację algorytmu dla przykładowych danych.
3. Dla algorytmu potęgowania x^n z zadania 9a z zestawu 2
 - (a) przeprowadzić ręczną symulację dla wybranych danych wejściowych,
 - (b) wyznaczyć złożoność obliczeniową przyjmując, że operacją elementarną jest operacja mnożenia.
4. Przeanalizować podany poniżej algorytm potęgowania szybkiego a następnie:
 - (a) przeprowadzić ręczną symulację dla wybranych danych wejściowych,
 - (b) pokazać, że algorytm poprawnie wylicza wartość x^n ,
 - (c) wyznaczyć złożoność obliczeniową przyjmując, że operacjami elementarnymi są operacje mnożenia i dzielenia (*wskazówka*: wyznaczyć liczbę operacji w najgorszym i najlepszym przypadku),
 - (d) porównać ze złożonością algorytmu z poprzedniego zadania.

```
r ← 1
while n > 0
do
  if not 2 | n
    r ← r · x
  fi
  x ← x · x
  n ← n div 2
od
```

Uwaga: zapis „2|n” oznacza „2 dzieli n”, natomiast **div** jest operacją dzielenia całkowitego.

¹Proste przykłady sposobu przeprowadzenia ręcznej symulacji zawarte są w plikach pdf dołączonych do tych ćwiczeń jako Materiały.

5. Problem wieży Hanoi. Wieża składa się z trzech prętów (A, B, C), na których umieszczono krążki o różnych średnicach. Krążki można przenosić tylko pojedynczo, nie można też umieszczać większego na mniejszym. Zadanie polega na przeniesieniu wieży z pręta A na C (ułożenie wejściowe pokazane jest na rysunku)



- (a) zaproponować algorytm układania wieży Hanoi o dowolnej wysokości, (zakładamy, że istnieje funkcja *przełoz*(x,y), która przenosi krążek z pręta x na pręt y),
- (b) wyznaczyć złożoność algorytmu przy założeniu, że operacją elementarną jest przełożenie krążka.
6. Zaproponować algorytm wyszukiwania słowa w tekście (dokładnego dopasowania). Danymi wejściowymi algorytmu są dwa napisy (tablice znaków), z których pierwszy jest tekstem, w którym poszukujemy słowa, a drugi – poszukiwanym słowem. Algorytm powinien zwracać pozycję pierwszego wystąpienia poszukiwanego słowa lub -1 gdy słowo w tekście nie występuje.
7. Zaproponować algorytm sprawdzania czy dany ciąg znaków jest palindromem (słowem czytany tak samo w przód i wstecz, np. *kajak*). Algorytm powinien na wejściu otrzymywać napis (tablicę znaków) i zwracać 1, gdy słowo jest palindromem, a 0 – gdy nim nie jest. (*wersja trudniejsza*: analizie poddawane są całe zdania, pomijają się wtedy odstępki i znaki przestankowe – takim palindromem jest np. „kobyła ma mały bok”).
8. Znaleźć algorytm wyznaczający przy użyciu operacji sumowania elementy *i*-tego wiersza trójkąta Pascala, którego kilka pierwszych poziomów przedstawiono poniżej.

0				1									
1				1	1								
2				1	2	1							
3				1	3	3	1						
4				1	4	6	4	1					
5				1	5	10	10	5	1				
6				1	6	15	20	15	6	1			
7				1	7	21	35	35	21	7	1		
8				1	8	28	56	70	56	28	8	1	
9				1	9	36	84	126	126	84	36	9	1
			

Wskazówka: Zastanowić się w jaki sposób można wyznaczyć szukane wiersze przy użyciu jednej tablicy jednowymiarowej.