Algorytmy przechodzenia drzew

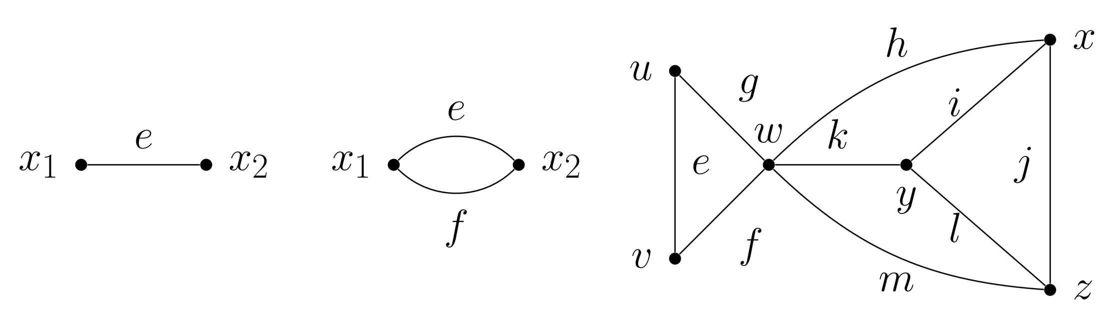
Patryk Szałas

Uniwersytet Marii Curie-Skłodowskiej w Lublinie

UMCS WYDZIAŁ MATEMATYKI, FIZYKI I INFORMATYKI

Digrafy

Digrafem nazywamy graf skierowany, w którym każda krawędź posiada ściśle określony punkt początkowy oraz końcowy, co nadaje relacjom między wierzchołkami konkretny kierunek.

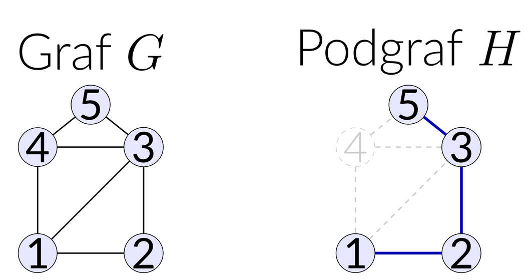


Przykłady grafów ilustrujące pojęcia drogi i cyklu.

Podgrafy i spójność strukturalna

Podgrafem nazywamy strukturę wyodrębnioną z grafu bazowego poprzez wybór podzbioru jego wierzchołków oraz krawędzi.

Usunięcie wierzchołka wymaga usunięcia wszystkich krawędzi incydentnych z tym wierzchołkiem. Zapewnia to spójność grafu i wyklucza istnienie krawędzi pozbawionych punktów końcowych.

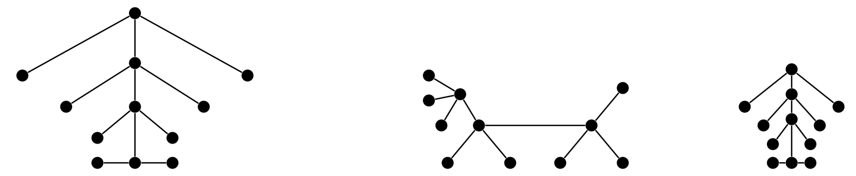


Podgrafy pozwalają definiować struktury szczególne, takie jak minimalne drzewa rozpinające. Ich konstrukcja wymaga jednak, aby graf bazowy był spójny – czyli taki, w którym każda para wierzchołków połączona jest drogą.

Definicja i własności drzew

Drzewem nazywamy spójny graf nieskierowany, który nie posiada cykli prostych. Musi on być zarazem:

- acykliczny - nie posiada żadnych cykli.
- spójny - stanowi jedną całość.



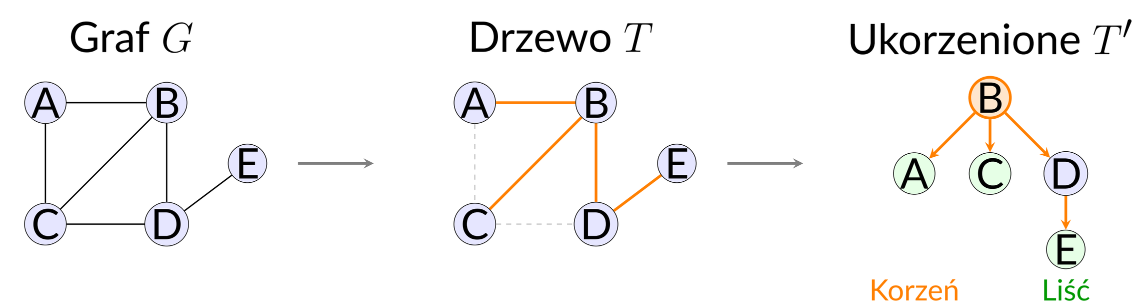
Graf acykliczny (las) zbudowany z trzech rozłącznych składowych (drzew).

Drzewa rozpinające i ukorzenione

Drzewem rozpinającym grafu spójnego G nazywamy taki jego podgraf, który jest drzewem i zawiera wszystkie wierzchołki grafu G.

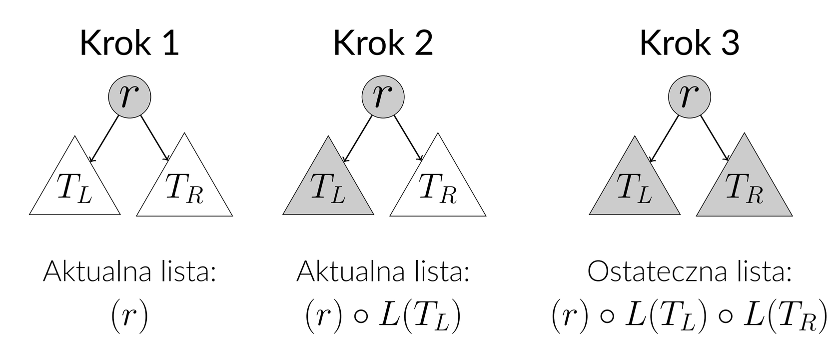
W zastosowaniach informatycznych operowanie na drzewach wymaga nadania im hierarchii, stąd wprowadzamy drzewa ukorzenione.

- Korzeń to jedyny węzeł bez przodka (rodzica).
- Wierzchołki sąsiednie to potomkowie (dzieci).
- Wierzchołki niemające potomków to liście.



PREORDER (Porządek prefiksowy)

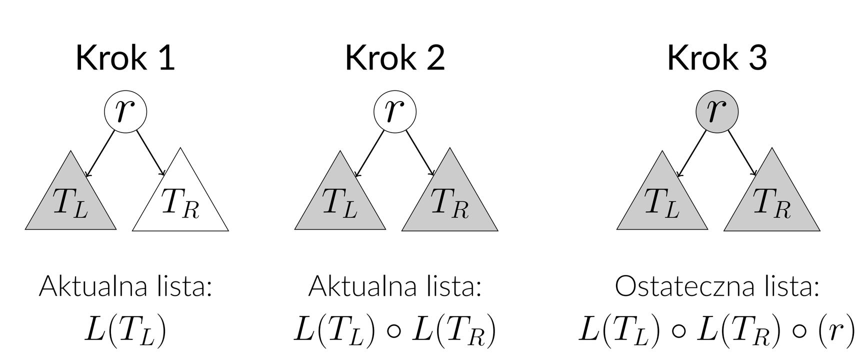
Węzły przetwarzane są od góry do dołu. Kolejność: korzeń, lewe poddrzewo, a na końcu prawe poddrzewo.



Zastosowanie: Metoda PREORDER jest optymalna do operacji klonowania (kopiowania) struktury drzewiastej, ponieważ korzeń nadrzędny tworzony jest jako pierwszy.

POSTORDER (Porządek postfiksowy)

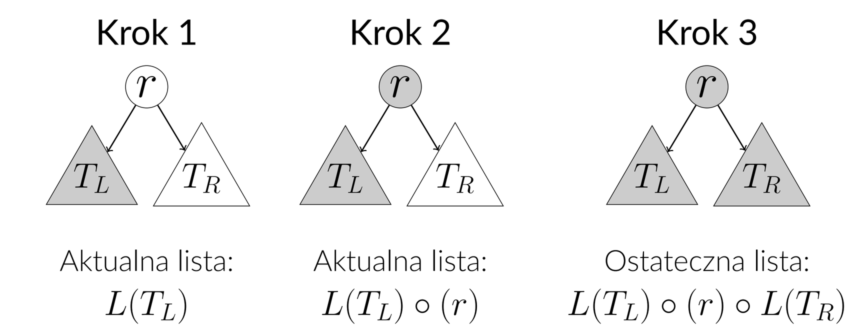
Węzły przetwarzane są od dołu do góry. Kolejność: lewe poddrzewo, prawe poddrzewo, a na końcu korzeń.



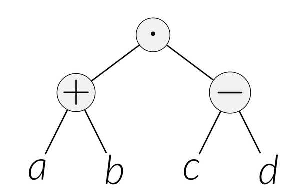
Zastosowanie: Algorytm POSTORDER znajduje zastosowanie przy bezpiecznym usuwaniu drzewa z pamięci. Usunięcie rodzica następuje dopiero po usunięciu wszystkich jego potomków, co zapobiega zaburzeniu spójności grafu w trakcie procesu jego całkowitej redukcji.

INORDER (Porządek infiksowy)

Węzły przetwarzane są od lewej do prawej. Kolejność: lewe poddrzewo, korzeń, a na końcu prawe poddrzewo.



Zastosowanie: Algorytm INORDER ma zastosowanie w odczytywaniu drzew wyrażeń matematycznych. Pozwala ona na wygenerowanie standardowej notacji infiksowej.



Drzewo wyrażenia arytmetycznego

Realizując rekurencyjny schemat *lewe poddrzewo → korzeń → prawe poddrzewo*, otrzymujemy ostatecznie ciąg wierzchołków:

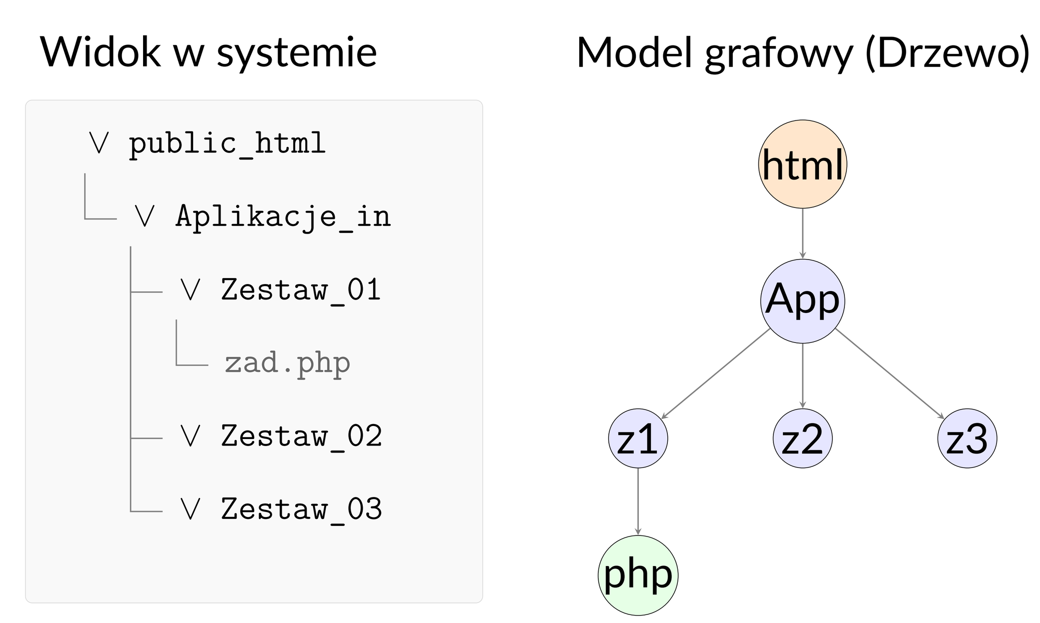
$$a, +, b, \cdot, c, -, d$$

Co po złączeniu stanowi dokładnie standardowy zapis wyrażenia matematycznego: $a + b \cdot c - d$.

Zastosowanie praktyczne: System plików

Drzewa matematyczne wykorzystujemy do sterowania systemem plików. Głównie wykorzystujemy algorytmy:

- **POSTORDER:** służy do usuwania i liczenia rozmiaru folderów (wymaga przetworzenia liści przed korzeniem).
- **PREORDER:** służy do kopiowania struktury katalogów (katalog bazowy tworzony jest przed jego zawartością).



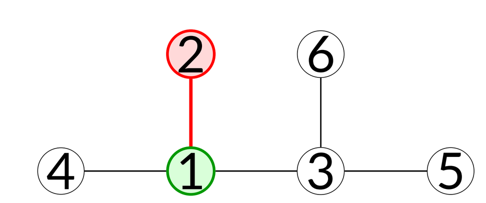
Kod Prüfera

Kod Prüfera pozwala jednoznacznie zapisać dowolne drzewo o n wierzchołkach jako ciąg $n-2$ liczb.

Algorytm generowania:

1. Znajdź liść (wierzchołek stopnia 1) o najmniejszej etykiecie.
2. Dopisz do kodu etykietę jego jedynego sąsiada.
3. Usuń ten liść oraz krawędź z nim incydentną.
4. Powtarzaj kroki 1-3, aż w grafie zostaną tylko dwa wierzchołki.

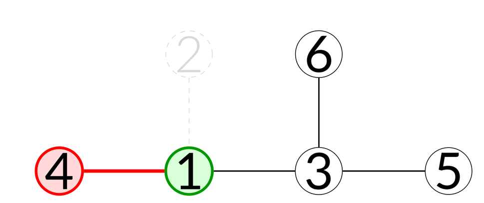
Krok 1



Usuwamy liść **2**, dodajemy **1**

Kod: [1]

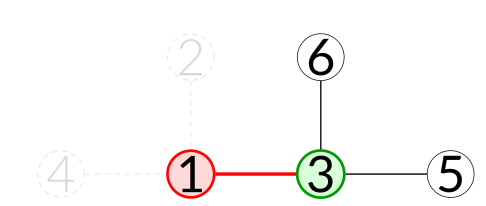
Krok 2



Usuwamy liść **4**, dodajemy **1**

Kod: [1, 1]

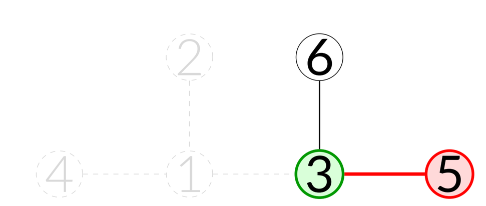
Krok 3



Usuwamy liść **1**, dodajemy **3**

Kod: [1, 1, 3]

Krok 4



Usuwamy **5**, zostają 2 wierzchołki.

Kod: [1, 1, 3, 3]

Bibliografia

[1] Biegańska T., Górnicka A., *Matematyka dyskretna: zbiór zadań*, Wydawnictwo Akademii im. Jana Długosza w Częstochowie, Częstochowa 2008.

[2] Krajka A., *Matematyka dyskretna*, Instytut Informatyki UMCS, Lublin 2011.

[3] Murat M., Gorgol I., *Matematyka dyskretna w zadaniach*, Wydawnictwo Naukowe PWN, Warszawa 2024.

[4] Rosen K. H., *Discrete Mathematics and Its Applications*, wyd. 8, McGraw-Hill, New York 2019.

[5] Ross K. A., Wright C. R. B., *Matematyka dyskretna*, wyd. 5, Wydawnictwo Naukowe PWN, Warszawa 2006.

[6] Szpirtowski A., *Matematyka dyskretna*, Wydawnictwo Uniwersytetu Gdańskiego, Gdańsk 2004.

[7] Włoch A., Włoch I., *Matematyka dyskretna. Podstawowe metody i algorytmy teorii grafów*, Oficyna Wydawnicza Politechniki Rzeszowskiej, Rzeszów 2008.

[8] Zakrzewski M., *Markowe wykłady z matematyki: Matematyka dyskretna*, Oficyna Wydawnicza GiS, Wrocław 2004.