

Algoritmizace – Grafy, prohledávání stavového prostoru, Řazení

Grafy

Grafy jsou způsobem reprezentace vztahů a propojení mezi určitými objekty / prvky. Objektům jsou přiřazeny vrcholy (též zvané uzly) a při zobrazení vztahu propojíme dva vrcholy přímkou – hranou.

Orientovaný a neorientovaný graf

U orientovaných grafů se jedná o uspořádané dvojice, zatímco u neorientovaných se jedná pouze o množinu prvků. Orientovaný graf poznáme podle toho, že místo přímky jako hrany využijeme šipku, čímž definujeme směr. Směr nám definuje, kudy se můžeme po hraně pohybovat.

Ohodnocený graf

Ohodnocený graf nám říká, jaká je vzdálenost mezi určitými vrcholy, případně délku hrany. Může také ale spoužít k jednoduchému přiřazení čísel ke hranám.

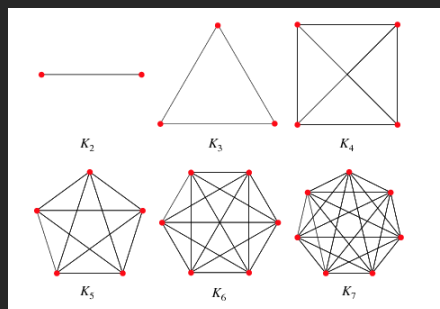
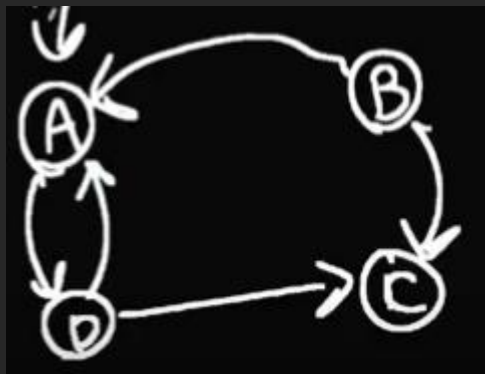
Úplný graf

Jedná se o graf, ve kterém je každý vrchol spojen se všemi ostatními vrcholy hranou.

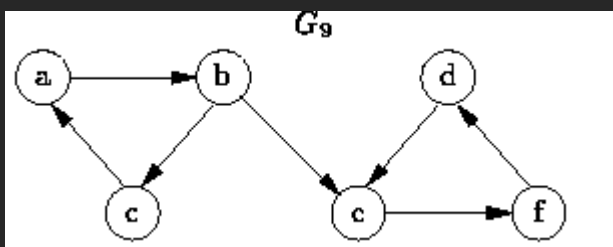
Cyklický a nocyklický

Cyklický graf je graf, ve kterém můžeme chodit dokola nekonečně skrze všechny body a nocyklický graf je takový, ve kterém se můžeme zaseknout a nejít skrze všechny body jednou cestou bez vrácení.

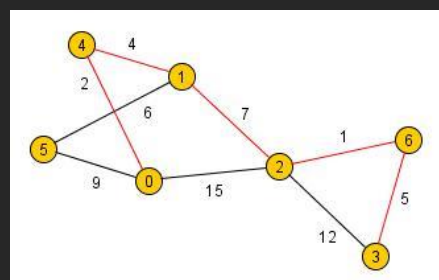
Necyklický graf s neorientovanou stranou AB Úplný graf



Orientovaný graf s okruhem A-B-C a C-D-F



Ohodnocený graf



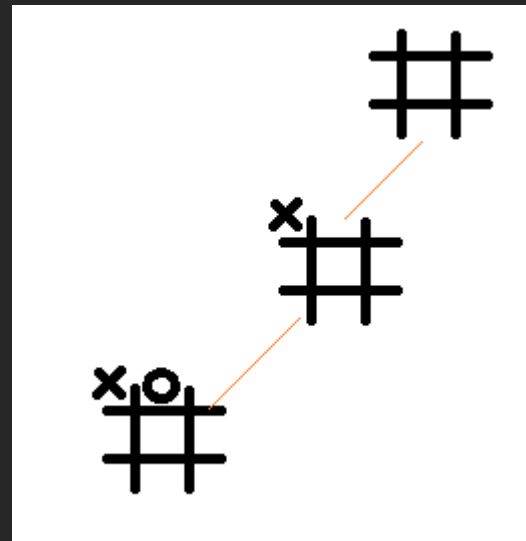
Prohledávání stavového prostoru

Stavový prostor je soubor všech stavů, které mohou v určitém problému nastat. V Javě se dá uložit do různých datových struktur, výhodná datová struktura je například strom.

Prohledávání stavového prostoru je určitá akce, ve kterém se snažíme najít určitý stav, který je nejlepší / nejvýhodnější / výsledkově pro nás dobrý. Prohledávání stavového prostoru jde do dvou směrů – do šířky, nebo do hloubky.

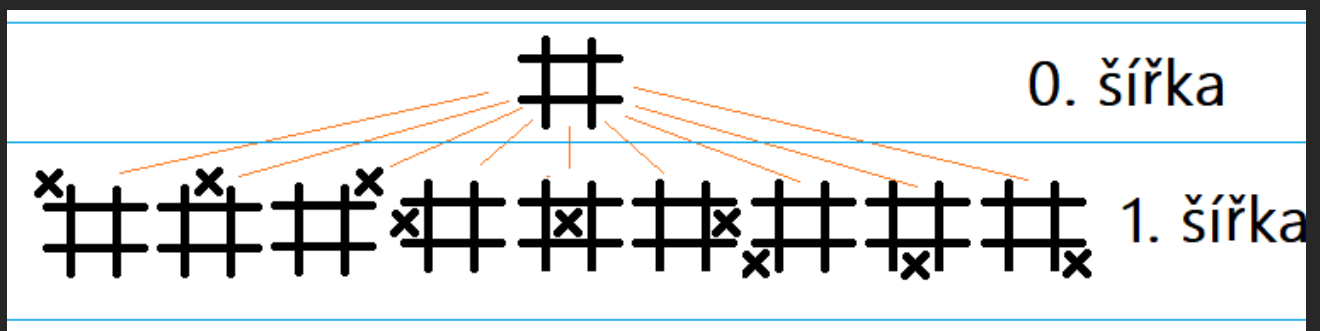
Dobrý příklad mohou být piškvorky, kde v případě, že si jsme jisti, že začíná křížek, je jako první stav prázdná mřížka

Při prohledávání do hloubky se vytváří nový stav, který okamžitě generuje další. Na obrázku vidíme dva první kroky prohledávání algoritmem do hloubky.



Příklad hledání do hloubky

Při hledání do šířky vygenerujeme nejdříve jednu vrstvu a poté všechny stavy v této vrstvy vygenerují další.



Vzhledem k tomu, že je stavový prostor množina všech možných výsledků, je potřeba k jeho kompletnímu projetí hrubá síla – bruteforce algoritmus.

Řazení

Řazení je operace, během kterého srovnáme určitou datovou strukturu plnou určitých dat dle potřeb. Využívají se k tomu různé typy řazení, každé má svojí rychlost a každý je vhodný pro určitý počet dat.

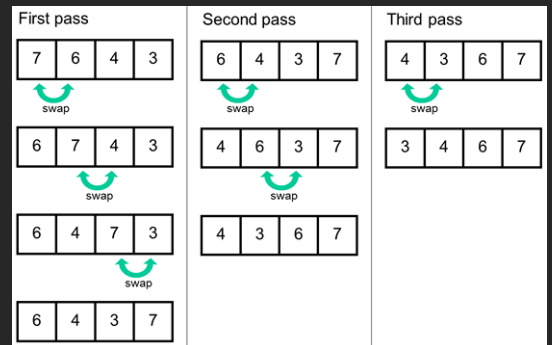
Je hodně sortů, třeba Radix, Selection, Stalin sort, Heap sort, ale nejznámější jsou tyto.

Bubble Sort

Bubble sort funguje jednoduše, vezme první dva prvky a na základě velikosti je vymění. Poté vezme následující prvek a udělá to samé.

Bubble Sort potřebuje celý jeden cyklus k tomu, aby mohl uznat, jestli je pole seříděné nebo ne.

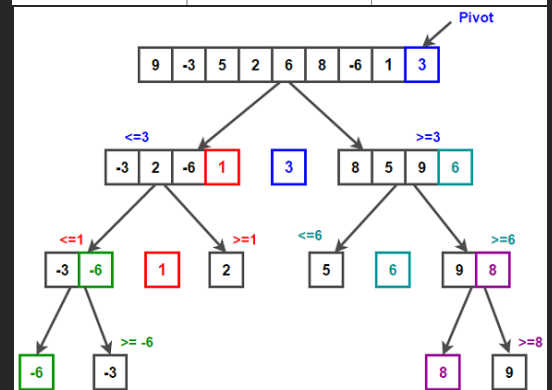
Časová náročnost: Kvadratická, n^2



Merge Sort

„Rozděl a panuj“. Tento algoritmus vezme nesetříděné pole a dělí je na poloviny tak dlouho, dokud nejsou pouze dvojice. Následně dvojici seřadí a poté všechny dvojice spolu s další dvojicí spojuje tak dlouho, dokud nemá celé pole.

Časová náročnost: $n \log n$



Quick Sort

Quick sort spočívá ve vybrání pivotu. Podle pivotu následně rozřadíme na prvky, které jsou menší a na prvky, které jsou větší. Ony dvě skupiny zase budou mít svého pivotu, který podskupinu rozdělí na další podpodskupiny a takhle do té doby, dokud není seřazeno.

Časová průměrná náročnost: $n \log n$

