**Úvod do grafů**

**Definice grafu**

**Orientovaný graf**

Orientovaný graf obsahuje orientované hrany, což jsou uspořádané dvojice uzlů (vrcholů). Každá hrana má směr, začíná v počátečním uzlu a končí v koncovém uzlu.

Směr hran se v grafu znázorňuje šipkami.

Příklad:

Obsah obrázku řada/pruh, kruh, skica, ramínko

Obsah generovaný pomocí AI může být nesprávný.

**Neorientovaný graf**

V neorientovaném grafu jsou všechny hrany neorientované neboli nezáleží na pořadí uzlů, hrana představuje symetrické spojení dvou vrcholů.

Hrana může spojovat dva různé vrcholy nebo tvořit smyčku (spojení vrcholu se sebou samým).

Příklad:

**Obsah obrázku skica, kresba, kruh, diagram

Obsah generovaný pomocí AI může být nesprávný.**

**Ohodnocený graf**

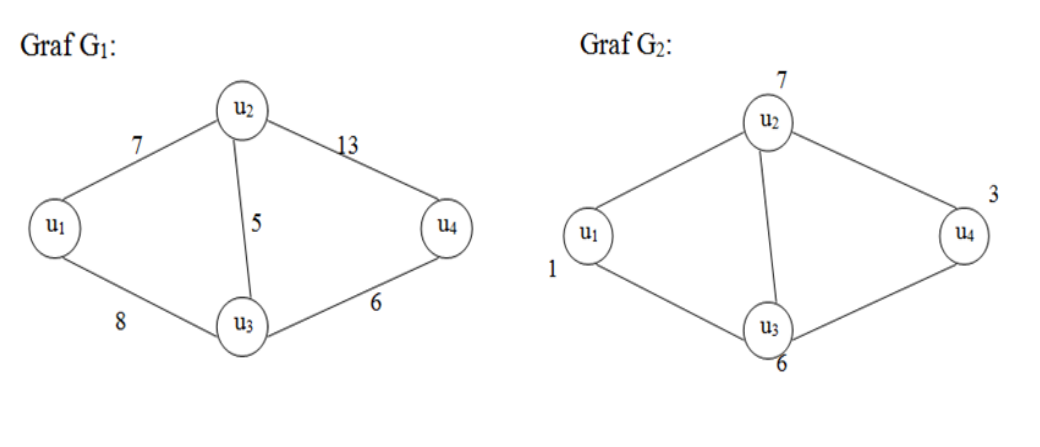
V ohodnoceném grafu jsou hrany nebo vrcholy opatřeny hodnotami (váhami). To umožňuje využití grafu pro výpočty, jako je hledání nejkratší cesty nebo optimalizace.

Rozlišujeme dva typy ohodnocených grafů:

Hranově ohodnocený graf (graf G1) - hodnota je přiřazena každé hraně (například vzdálenost mezi městy).

Uzlově ohodnocený graf (graf G2) - hodnota je přiřazena každému vrcholu (například doba trvání činností v síťovém diagramu).

Příklad:



**Kostra grafu**

Kostra grafu je podgraf souvislého neorientovaného grafu, který je: souvislý (existuje cesta mezi každými dvěma vrcholy), acyklický (neobsahuje žádné cykly), a obsahuje všechny vrcholy původního grafu.

Jde tedy o strom, který „pokryje“ celý graf, ale má nejmenší možný počet hran.

Minimální kostra

Kostra, která má nejmenší možný součet vah hran ze všech možných koster grafu.

Používá se například při: navrhování nejlevnější sítě silnic, kabelů nebo potrubí, optimalizaci spojení bez zbytečných nákladů.

Příklad: „Najděte nejlevnější cestu propojení všech bodů…“

Obsah obrázku řada/pruh, diagram

Obsah generovaný pomocí AI může být nesprávný.

**Matice sousednosti vs. seznam sousedů**

Matice sousednosti

Používá dvourozměrné pole (matici).

U neohodnocených grafů se používají Boolean hodnoty.

Obsah obrázku text, snímek obrazovky, diagram, vzor

Obsah generovaný pomocí AI může být nesprávný.U ohodnocených grafů obsahuje matice váhy hran.

Seznamem sousedů

Používá pole seznamů. (například pole záznamů, seznamů, množin)

Každý vrchol má svůj seznam, ve kterém jsou vrcholy, se kterými je spojen hranou.

Umožňuje efektivní práci s grafy, kde je malý počet hran vůči počtu vrcholů (řídké grafy).

Je vhodný, pokud potřebujeme flexibilní a paměťově úspornou reprezentaci grafu.

Obsah obrázku text, diagram, řada/pruh, Paralelní

Obsah generovaný pomocí AI může být nesprávný.

|  |  |  |
| --- | --- | --- |
|  | Matice sousednosti | Seznam sousedů |
| Vhodné pro | Husté grafy (Mnoho hran) | Řídké grafy (Málo hran) |
| Rychlost | Okamžitý | Pomalý |
| Obtížnost | Jednoduchá | Složitější |

**Reálné příklady využití (navigace, sítě, plánování)**

Hledání nejkratší cesty

Dopravní sítě: Určení nejkratší trasy mezi dvěma body na mapě. (například GPS navigace)

Síťová komunikace: Nalezení optimální cesty pro přenos dat v počítačové síti.

Logistika: Plánování tras pro doručování zásilek.

Sítě a konektivita

Sociální sítě: Modelování vztahů mezi uživateli, doporučování přátel nebo obsahu.

Ekonomické sítě: Analýza propojení firem, toků financí nebo závislostí v globálním trhu.

Inženýrské sítě: Optimalizace rozvodu elektřiny, plynu nebo vody.

Plánování a rozvrhování

Průmyslová automatizace: Řízení výrobních procesů a logistických toků.

Personalizované doporučení: Algoritmy pro doporučování obsahu (například filmy, produkty…) na základě chování uživatelů.

Sestavování rozvrhů: Tvorba školních rozvrhů, harmonogramů projektů nebo plánů výroby.

Analýza dat

Těžba dat: Objevování vzorů a vztahů v rozsáhlých datových souborech.

Strojové učení: Grafy se používají k reprezentaci dat (například rozhodovací stromy, neuronové sítě) a učení modelů.

Další využití

Hraní her: Modelování herních světů, navigace postav, strategie.

Bioinformatika: Analýza genetických sekvencí, proteinových interakcí.

Zpracování obrazu: Rozpoznávání objektů a segmentace obrázků pomocí grafových algoritmů.

**Problém hledání nejkratší cesty**

**Co znamená „nejkratší cesta“**

Nejkratší cesta mezi dvěma vrcholy je posloupnost hran vedoucí z výchozího bodu do cílového bodu tak, aby celková délka (součet vah hran) byla co nejmenší.

Vstup

Orientovaný a neorientovaný graf s nezápornými ohodnoceními hran.

Výchozí bod.

Cílový bod.

Výstup

Délka nejkratší cesty mezi výchozím a cílovým bodem.

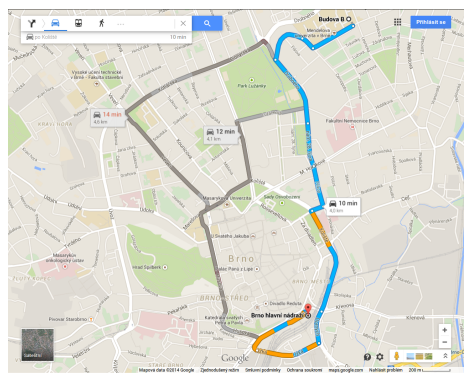
Aplikace

Nejkratší cesta v dopravním slova smyslu. (například GPS navigace)

Optimalizace výrobních procesorů.

Plánování tras v logistice nebo sítích.

Příklad: „Nejkratší cesta“



**Negativní hrany a jejich vliv**

Příklad: Hrana mezi vrcholy A a B má váhu -5. To znamená, že přejít touto hranou stojí -5 což je „výhodněji“, než jiné hrany s kladnými hodnotami.

Využití

Záporné hrany se vyskytují, například, kdy cesta přináší zisk místo nákladů (sleva, odměna, úspora…), minimalizování ztráty nebo optimalizování kombinace s různými dopady, nebo modelování časových rozdílů (například spoje s přestupy, kde se čekáním někdy ušetří čas).

Problém s negativními hranami

Ne všechny algoritmy si s nimi poradí. Dijkstrův algoritmus například předpokládá, že všechny hrany mají nezáporné váhy, s negativními hodnotami vrací nesprávné výsledky. Pro grafy s negativními hranami se používá například Bellman-Fordův algoritmus.

**Přehled algoritmů**

**Dijkstrův algoritmus**

Princip (greedy přístup, prioritní fronta)

Omezení (nefunguje s negativními hranami)

Časová složitost: O(E\*logV), kde E je počet hran a V počet vrcholů

**Bellman-Fordův algoritmus**

Princip (relaxace hran, až V−1 iterací)

Výhoda: funguje i s negativními hranami

Detekce záporných cyklů

Časová složitost: O(V\*E)

**Floyd-Warshallův algoritmus**

Princip (dynamické programování)

Vhodný pro všechny páry vrcholů

Časová složitost: O(V³)

**Praktická část**

Algoritmus: Bellman-Fordův

Jazyk: C#

Vstup:

Obsah obrázku diagram, řada/pruh, kruh, design

Obsah generovaný pomocí AI může být nesprávný.

Výstup:

Obsah obrázku text, snímek obrazovky, menu, Písmo

Obsah generovaný pomocí AI může být nesprávný.

**Zdroje:**

<https://slideplayer.cz/slide/2358609/>

<https://is.slu.cz/el/opf/zima2021/INMBPOAE/3113831/Prednaska_9.pdf>

<https://user.mendelu.cz/marik/inzmat/slidy/grafy.pdf>

<https://cs.wikipedia.org/wiki/Matice_sousednosti>

<https://portal.matematickabiologie.cz/index.php?pg=zaklady-informatiky-pro-biology--teoreticke-zaklady-informatiky--teorie-grafu--reprezentace-grafu--seznam-sousednosti>

<https://www.geeksforgeeks.org/dsa/dijkstras-shortest-path-algorithm-greedy-algo-7/>

<https://www.geeksforgeeks.org/dsa/floyd-warshall-algorithm-dp-16/>

<https://www.geeksforgeeks.org/dsa/bellman-ford-algorithm-dp-23/>