

## Úloha 126 – Procházení grafu metodou BFS (Breadth First Search)

### Zadání

Z textového souboru načtete reprezentaci grafu G a uložte ji. Každá hrana grafu má ohodnocení = 1. Pro implementaci metody BFS použijte frontu. Prohledávání grafu začne ve Vámi zvoleném uzlu, a do textového souboru se uloží výsledný BF strom.

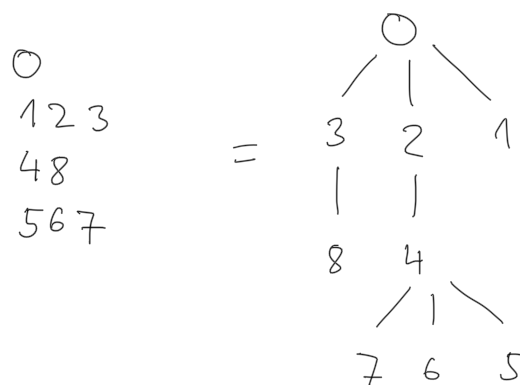
### Rozbor problematiky

#### Formální definice

Prohledávání grafu metodou BFS neboli prohledávání do šířky je základní grafový algoritmus pro procházení grafu. Tento algoritmus systematicky prochází graf z libovolného uzlu a pro dočasné ukládání nenavštívených uzlů používá frontu. Díky ukládání do fronty dochází k procházení grafu po vlnách, kdy jsou uzly zpracovány v pořadí daném jejich vzdáleností od kořene.

#### Popis průběhu algoritmu

Na počátku algoritmu proběhne inicializace, poté se nastaví počáteční hodnoty pro jediný uzel, u kterého známe všechny informace. Rozeběhne se cyklus, který běží, dokud je fronta neprázdná (tzn. uzly se kterými pracujeme). Na počátku fronty vyjmemu uzel a pro všechny ostatní následníky budeme zjišťovat, zda byl bod již nalezen anebo prozkoumán. Když bude tato podmínka splněna, nastaví se informace pro tyto uzly (vzdálenost od počátku nastaví o jedna větší než uzel, kterého je tento uzel následníkem). Vzdálenost je větší právě o jedna, aby byla splněna podmínka, že je mezi uzly právě jedna hrana. Následně se pro tyto následníky nastaví předchůdce na hodnotu následníka uzlu. Po nastavení těchto vlastností je vložen do fronty. Když jsou prohledáni všichni následníci uzlu, uzel se zavře a pokračuje se na další.



Obrázek 1 - testovací příklad a vizualizace stromu

Popis funkcí

```
class Vertex
```

Obsahuje statickou proměnou, která se o jeden zvyšuje s každým vrcholem, který přijme.

```
def __init__(self, label)
```

Konstruktor obsahující prázdný list sousedů, který se průběhem doplňuje přiřazováním.

```
def add_neighbour(self, v)
```

Zkontroluje, zda je vertex zapsaný v sousedech a když ne, tak ho mu dá append.

```
class Graph
```

```
def __init__(self, lines)
```

Funkce dostane v konstruktoru pole řádků (na každém indexu jeden řádek) a pro každý řádek rozdělí podle mezery. Třída bude mít v sobě uložený objekt s vrcholy. Pro každý řádek z řádků rozdělí mezerou, což vrátí pole o dvou prvcích. První index do Label1 a druhý do Label2. Pokud následně label1 nebude v uloženém seznamu, tak ho na pozici uloží pomocí třídy Vertex. Podmínky pro label1 a label2 kontrolují, zda je již objekt v seznamu a když ne, tak ho vytvoří. Tyto podmínky se přeskočí, když se bude hrana vyskytovat již mezi uloženými labely. V tomto případě se hodí jako k label1 soused label2 a opačně též.

```
def read(input_name)
```

Čtecí funkce, která načte textový soubor zvolený uživatelem jako parametr. Nejprve fyzický soubor načte a poté rozdělí, aby na každém indexu byl jeden řádek. Funkce obsahuje 3 validační výjimky, kterými ověří, že soubor existuje, že k němu má přístup, a že je obecně validní pro použití programem

```
def breadth_first(graph)
```

Na začátku se inicializuje fronta, která bude prázdná, poté se inicializuje pole visited, které má zatím na všech indexech False (od indexu 0 po index velikosti grafu – počtu vrcholů). Následně se do fronty přidá první vertex a za index tohoto prvního vrcholu dáme True. Inicializují výsledek, u kterého vím, že je samostatný. A vytvořím proměnou currentLevel, ta bude na začátku prázdná. Ve frontě se již nachází first vertex a tak mohu procházet, vyjmu vertex ze fronty pomocí .pop, která bere první prvek, který jsem tam jako první vložil. Koukne, jestli je vertex v aktuální úrovni (currentLevel), což není, takže se podmínka přeskočí. For cyklus přidá souseda do fronty jestliže nebyl navštívený, jeho index změním na True (byl navštíven) a přidám

ho do aktuálního levelu – sousedy prvního vrcholu. Pro sousedy se opakuje while dokud nejsou prozkoumáni všichni. Podmínka if přidává aktuální úroveň stromu

`def save(file_name, tree)`

Index úrovně v rozsahu délky souboru (stromu). Pro první vrchol je podmínka, aby nebyla mezera na začátku souboru, label se dá do stringu a uloží se, v případě více vrcholů skočí else, které mezi stringy přidá mezery. Pokud se index nerovná poslednímu indexu (je dosažena poslední úroveň), udělá nový řádek, odsadí a zapíše do souboru .txt.

Funkce ukládá vertexy po zjištěných levelech tak, že výsledný strom v textové souboru vypadá takto:

0

1 2 3

4 8

5 6 7

kdy jsou čísla rozdělena po úrovních.

Vstupy a výstupy

*Vstup*

Vstupem programu je textový soubor ve formátu číselných párů na jednotlivých řádcích.

*Výstup*

Výstupem programu je textový soubor, který má na řádcích reprezentované úrovně stromu.