Vysoké učení technické v Brně
Fakulta informačních technologií

Algoritmy
Rovinnost grafu 2022/2023

Technická zpráva

Rastislav Mazúr (xmazur09)
Jakub Kavka (xkavka02)
Ondřej Podroužek (xpodro03)
Vladimír Mečiar (xmecia00)

Brno, 7. prosince 2022

Obsah

1 Zadání varianty

Graf je rovinný, jestliže lze jeho vrcholy (jako body) a hrany (jako spojité křivky) umístit do roviny tak, aby se žádné dvě hrany nekřížily.

Vytvořte program, který rozhodne, zda je graf rovinný.
Součástí projektu bude načítání grafů ze souboru a vhodné testovací grafy. V dokumentaci uveďte teoretickou složitost úlohy a porovnejte ji s experimentálními výsledky. Nepředpokládá se, že by nalezení rozmístění vrcholů a hran v rovině bylo součástí řešení. Pozor! Otestování rovinnosti grafu je netriviální problém. Můžete použít algoritmus založený na tvrzení pana Kuratowského (1930): Rovinný graf neobsahuje podgraf izomorfní s grafy K(5) nebo K(3,3).

2 Práce v týmu

2.1 Rozdělení práce

První s schůzku jsme měli v oktobri. Touto dobou jsme se seznamovali a začali sme rozoberat zakladnu problematiku grafov. Sucastou bolo aj urcenie algoritmu na odhalenie planarity grafu. Od tejto casti sme nanestasie dlho nijako nepokrocili nakolko sme mali na starosti ine poinnosti suvisiace so skolou

Práci jsme neměli dopředu rezdělenou a úkoly jsme si rozdělovali podle potřeby v průběhu práce.

Navrh a implementacia automatu pre prijmanie znakov

- Ondrej Podrouzek

Testovanie automatu

- Rastislav Mazur

Navrh, implementacia a testovanie datovej struktury pre ukladanie grafu

- Vladimír Mečiar

Implementacia finalnej logiky programu

- Jakub Kavka

Testovanie finalnej logiky programu

- Rastislav Mazur

Tvorba dokumentace

- Vladimír Mečiar

Revizia kodu a dokumentacie

- Rastislav Mazur, Ondrej Podrouzek

2.2 Verzování

Pro sdílení a verzování kódu jsme použili verzovací systém GitHub.

V projektu jsme pracovali na více částech současně. Pro každou větší část jsme vytvářeli vlastní větve, které jsme po dokončení mergovali do větve `develop`. Do finální verze repozitara `main` jsme prepojili až finální otestovanou verzi a následně jsme přímo v ní prováděli jen poslední úpravy.

2.3 Komunikace

Jako komunikační platformy jsme využili *Discord*, který sloužil jako hlavní komunikační kanál. Pro sdílení učebních materiálů jsme vytvořili `Discord` server.

3 Nacitanie dat

3.1 Vstupne data

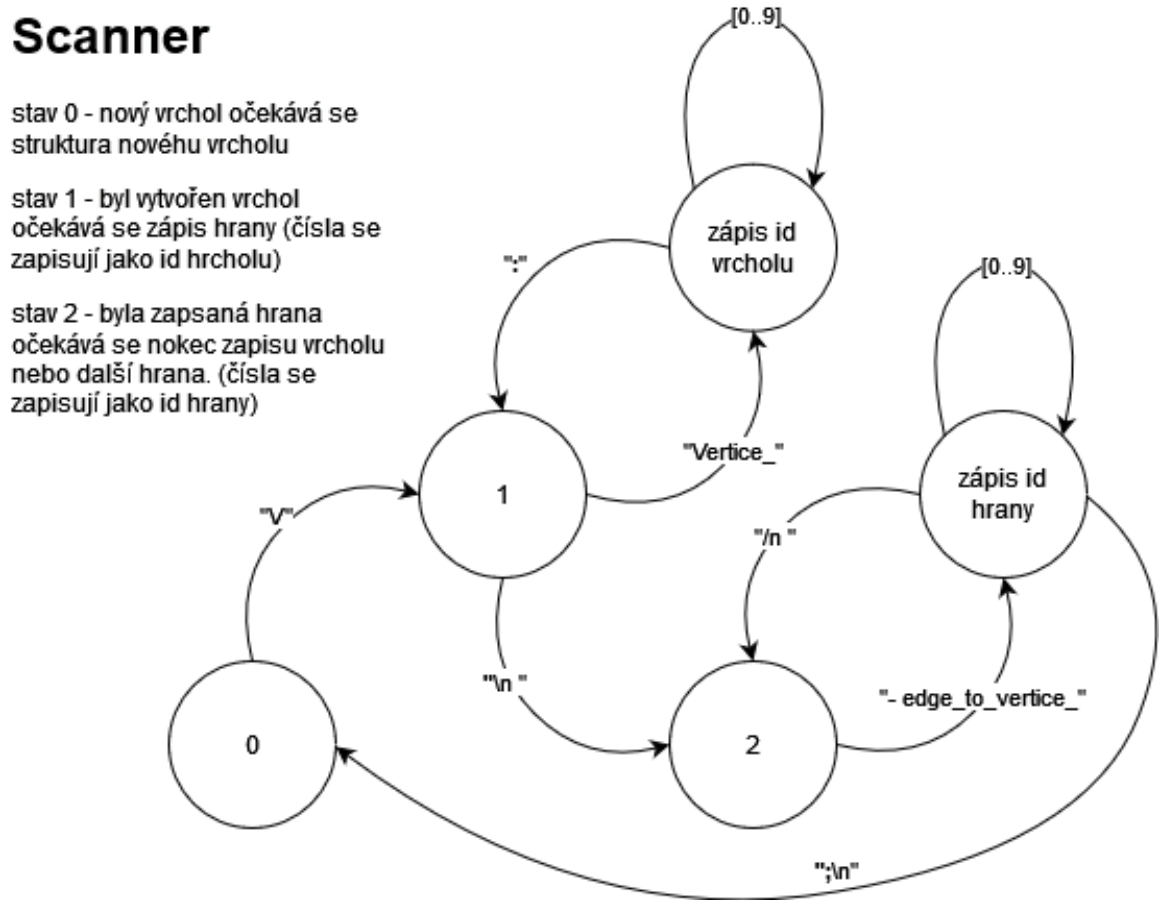
Struktura ktoru potrebujeme nacitat (graf), je definovana ako mnozina vrcholov a hran vrcholov. Pre nase potreby sme pouzili ulozenie v texte s predpisanym formatom.

```
Vertice_id:
  - edge_to_vertice_id
  - edge_to_vertice_id
  - edge_to_vertice_id
  - edge_to_vertice_id;
```

Vdaka takemuto ulozeniu sme mohli pouzit lexikalny analyzator na nacitanie iste formatu.

3.2 Lexikalny analyzator

Lexikalny analyzator je vlastne deterministicky konecny automat (DKA), ktory vie spracovavat istu sekvenciu znako. V pripade neznamej sekvencie znakov nacitanie ukonci a vypise chybovu hlasku

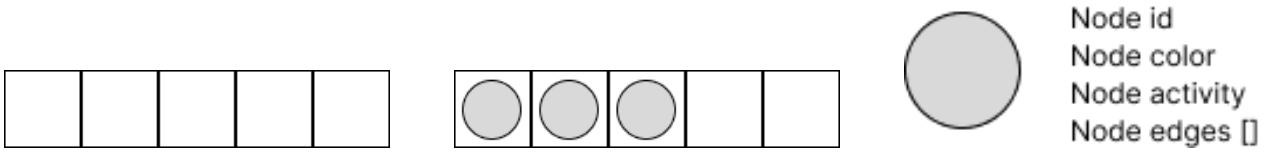


Obrázek 1: DKA

4 Ukladanie dat

5 Datova struktura

Pre ulozenie dat sme si potrebovali vybrat vhodnu datovu strukturu. V nasom pripade sme pouzili nafukovacie pole. Nadedinovali sme si ju ako strukturu s nazvom vektor. Potom nam zostalo uz len ulozit uzly.Pre tie sme si nadefinovali zvlast strukturu

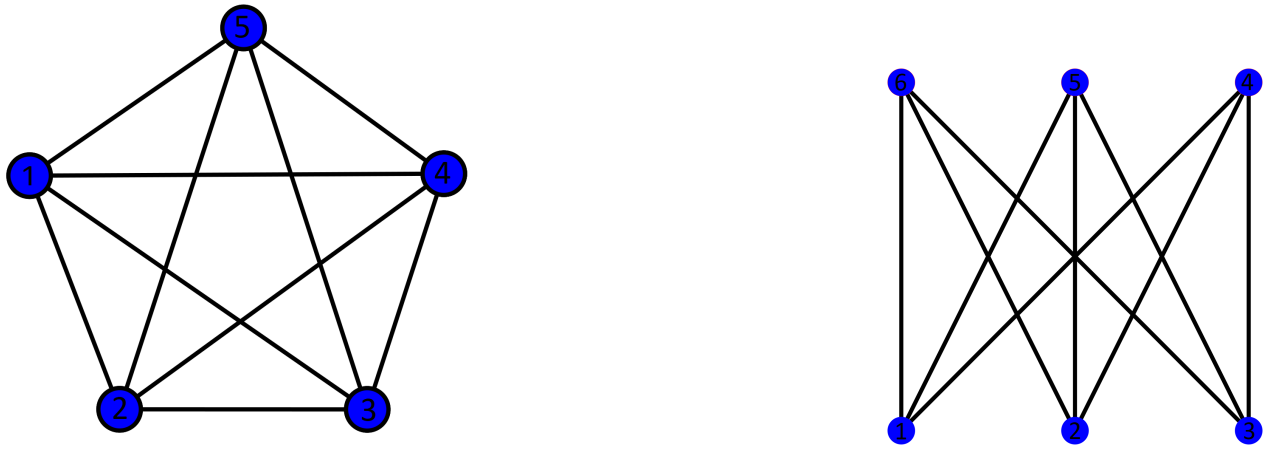


(a) Vektor (b) Graf (c) Uzol

Obrázek 2: Three simple graphs

6 Finalna logika

Na testovanie rovinnosti existuje niekolko podmienok. Zial vacsina z nich je len postacujuca, nie nutna. Jedine co zostalo pouzit je Kuratowskeho teorem. Jeho riesenie spociva v tom ze sa snazime v grafe najst podgraf grafov 3a a 3b. Zial, nevyhoda tohoto algoritmu je neefektivita. Musime porovnat vela uzlov medzi sebou



(a) Graf K_5

(b) Graf $K_{3,3}$

Obrázek 3: Nelinearne grafy

7 Testovanie

Na testovanie sme pouzili porovnanie aktualnych vysledkov programu s referencnymi vysledkami. Rozdelenie bolo pouzite ako v pribalenom Makefile.

- Testovanie datovych struktur
- Testovanie Lexikalneho analyzatoru
- Testovanie finalnej logiky algpritmu

Napomocne nam boli skripty v adresari test

8 Závěr

Při řešení jsme narazili na spousta nejasností, které jsme museli řešit na Discordu, nebo některá na fóru projektu.

Celkově byl tento projekt přínosný. Naučili jsme se práci v týmu a organizování času. Velkým přínosem byly zejména zkušenosti s verzovacími systémy *Git*.