



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**

FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV INFORMAČNÍCH SYSTÉMŮ**

DEPARTMENT OF INFORMATION SYSTEMS

**DEMONSTRACE - KARGERŮV ALGORITMUS**

DEMONSTRATION - KARGER'S ALGORITHM

**PROJEKTOVÁ DOKUMENTACE**

PROJECT DOCUMENTATION

**AUTOR PRÁCE**

AUTHOR

**KATEŘINA PILÁTOVÁ, MICHAL TABÁŠEK**

**BRNO 2017**

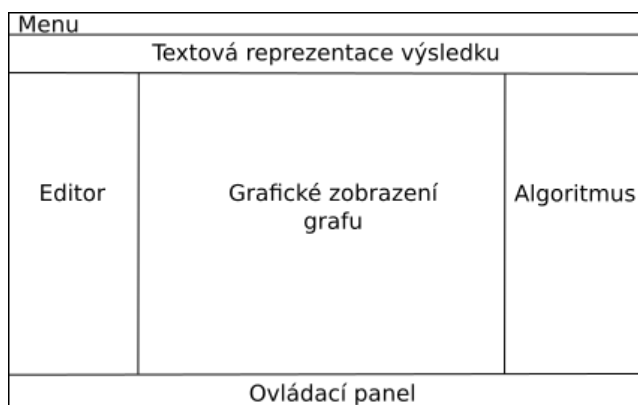
## Úvod

Cílem tohoto projektu je návrh a implementace demonstrační aplikace Kargerova algoritmu pro nalezení minimálního řezu. Daný algoritmus byl navržen a implementován s využitím vhodných doporučených nástrojů.

## Návrh aplikace

Předběžný návrh aplikace je důležitým krokem ve vývoji. Při tomto postupu je možno se vyhnout případným problémům, které by mohly nastat v pozdější fázi vývoje. Prvním krokem byl návrh základního rozložení hlavního okna aplikace.

- 



Obrázek 1: Layout hlavního okna aplikace.

## 1 Použité nástroje

Dalším krokem návrhu aplikace byl výběr programovacího jazyka, ve kterém bude aplikace napsána. Výsledná aplikace by měla mít grafické rozhraní (GUI) a měla by být spustitelná na jakékoliv platformě, především na studentském serveru Merlin. Na základě těchto kritérií byl zvolen jazyk Java spolu s knihovnou Swing pro tvorbu grafického rozhraní. Ze seznamu povolených formátů grafů byl vybrán formát `mxGraph` a knihovna `JGraphX` s tímto formátem pracující.

### Swing

Swing je knihovna plně založená na platformě Java a postavená na AWT (Abstract Windowing Toolkit), sloužící k ovládání počítače pomocí grafického rozhraní. S využitím této knihovny je možno vytvářet okna (`JFrame`), dialogy (`JOptionPane`), tlačítka (`JButton`), seznamy a mnoho dalšího [2].

### JGraphX

`JGraphX` je knihovna licencovaná pod licencí BSD, založená na Java Swing frameworku, poskytující funkcionalitu pro vizualizaci a interakci s grafy založenými na systému uzelnáhrana. `JGraphX` také obsahuje funkce pro podporu XML šablon, různých importů a exportů a rozvržení grafu [1].

## Implementace

Hlavní částí vývoje byla implementace algoritmu podle zvolené varianty zadání, v případě tohoto projektu o Kargerův algoritmus.

### 1 Algoritmus

Jedná se o deterministický randomizovaný algoritmus, který náhodně vybírá hranu mezi všemi hranami grafu a slučuje koncové uzly této hrany, dokud nezůstanou pouze dva super uzly [3]. Pseudokód hlavní smyčky programu:

---

**Algorithm 1** Karger Algorithm

---

```
1: Let  $G = (V, E)$ 
2: while  $|V| > 2$  do
3:    $(u, v) \leftarrow \text{uniform random edge from } F$ 
4:    $\text{Merge}(\bar{u}, \bar{v}, V)$ 
5:    $F \leftarrow F \setminus E_{\bar{u}\bar{v}}$ 
6: Return one of the supernodes in  $V$  and  $|E_{uv}|$ 
```

---

a spojení dvou uzlů:

---

**Algorithm 2**  $\text{Merge}(\bar{u}, \bar{v}, V)$ 

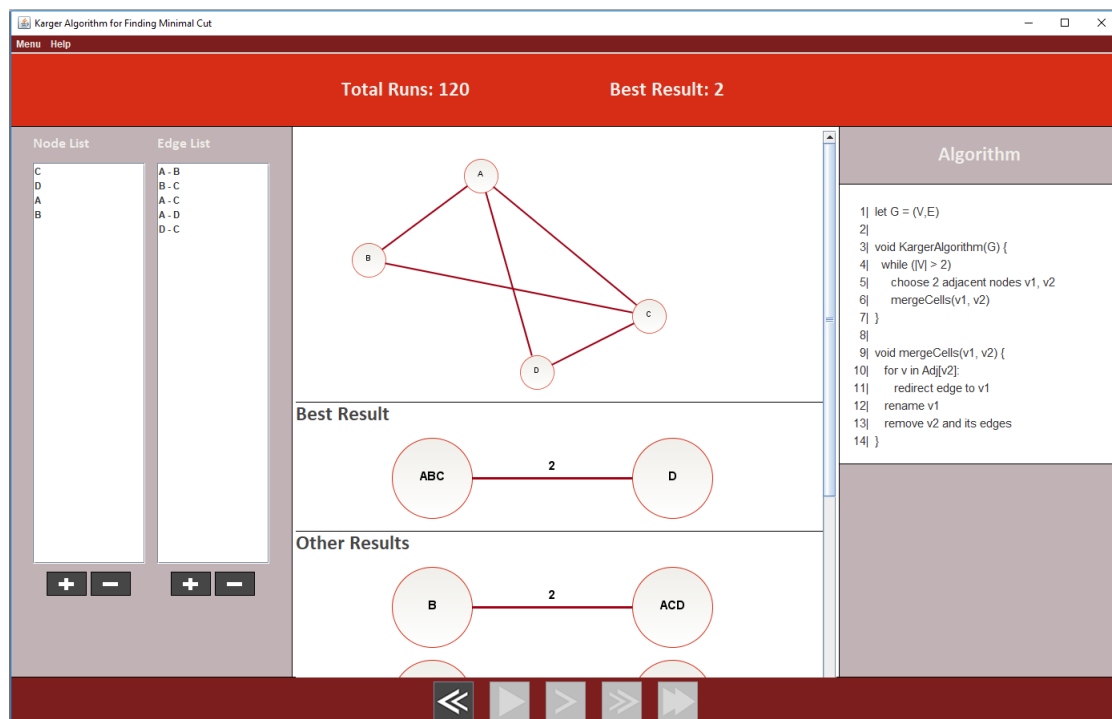
---

```
1:  $x \leftarrow \text{new supernode}$ 
2:  $V(x) \leftarrow V(a) \cup V(b)$  //merge the vertices
3: for each  $d \in V \setminus \{a, b\}$  do
4:    $E_{xd} \leftarrow E_{ad} \cup E_{bd}$  //operation using linked list
5:    $V \text{ gets}(V \setminus \{a, b\}) \cup \{x\}$ 
```

---

### 2 Struktura aplikace

Celá aplikace je tvořena jedním hlavním oknem MainWindow. Rozložení hlavního okna je zobrazeno v návrhu aplikace na obrázku 1. Po spuštění aplikace proběhne načtení a zobrazení defaultního grafu. Uživatel má možnost vybrat pro zpracování jiný graf. Graf je možno editovat přidáním nebo odebráním uzlů a hran. Případná komunikace s uživatelem je realizována prostřednictvím dialogových oken.



Obrázek 2: Zobrazení výsledku aplikace algoritmu na daný graf.

## Aplikace

### 1 Minimální požadavky

- OpenJDK
- Ant
- JGraphX

### 2 Spuštění a instalace

Odevzdaný archiv obsahuje:

- src - adresář se zdrojovými soubory
- libs - adresář s využitými knihovnami
- latex - zdrojové kódy této dokumentace
- readme
- dokumentace.pdf - tato dokumentace
- build.xml - skript pro kompilaci a spuštění aplikace pomocí Ant

Pro spuštění aplikace je nutno mít nainstalovaný OpenJDK a Java knihovnu Apache Ant. Aplikace se spouští přes soubor karger.jar.

### 3 Možnosti aplikace

- Vytvoření/načtení/uložení grafu ve formátu XML
- Editace grafu - přidání a odebrání uzlu/hrany
- Ovládání grafu pomocí ovládacího panelu - reset, další krok, dokončení jednoho běhu, dokončení algoritmu
- Podrobnější krokování provádění algoritmu spolu s vyznačením právě provedených částí pseudokódu
- Přesun uzlů pomocí kliknutí a tažení myši

### Závěr

Cílem práce bylo vytvořit aplikace, která demonstruje Kargerův algoritmus pro nalezení minimálního řezu. Samotné řešení projektu s využitím knihovny JGraphX bylo poučné a vedlo k pochopení fungování Kargerova algoritmu. Výsledná aplikace je vydařená a toto řešení by mělo uživatele pomoci minimálně s pochopením základního principu algoritmu.

# Literatura

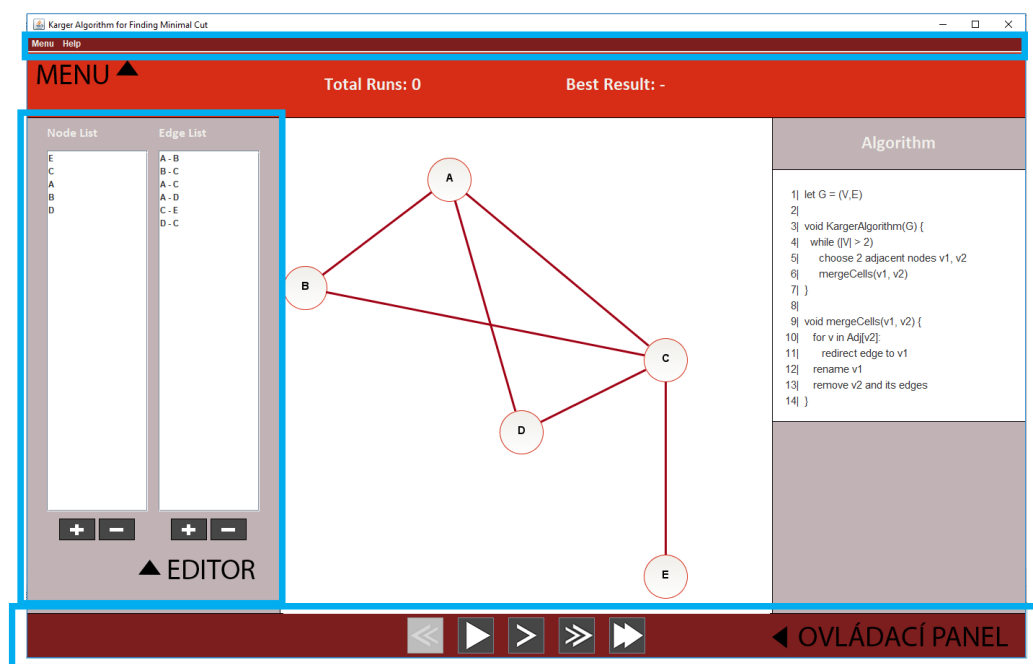
- [1] *JGraphX (JGraph 6) User Manual*. [Online; navštíveno 12.10.2017].  
URL <https://github.com/jgraph/jgraphx>
- [2] *Lesson: Getting Started with Swing*. [Online; navštíveno 12.10.2017].  
URL <https://docs.oracle.com/javase/tutorial/uiswing/start/index.html>
- [3] Williams, V. V.: *Min Cut and Karger's Algorithm*. Květen 2016, [Online; navštíveno 12.10.2017].  
URL <http://web.stanford.edu/class/archive/cs/cs161/cs161.1166/lectures/lecture15.pdf>

# Příloha A

## Uživatelská příručka

Tato aplikace slouží k demonstraci Kargerova algoritmu.

### Ovládání aplikace

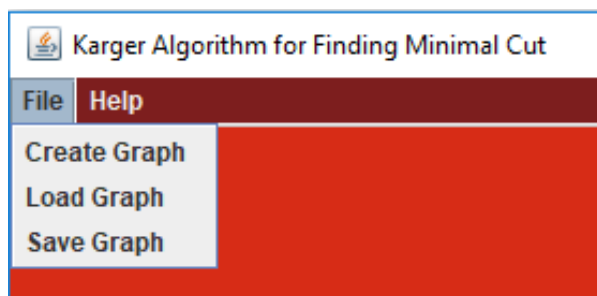


Obrázek A.1: Rozhraní ovládání aplikace.

Aplikaci lze ovládat pomocí:

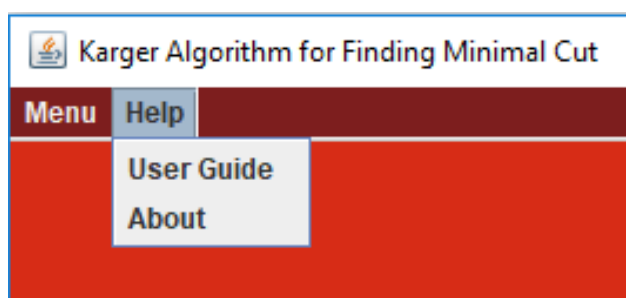
- MENU - Práce se souborem, nápověda.
- EDITORU - Úprava grafu.
- OVLÁDACÍHO PANELU - Aplikace algoritmu na graf.

## 1 Menu



Možnosti práce se souborem:

- vytvoření nového grafu
- načtení uloženého grafu ve formátu XML
- uložení grafu ve formátu XML





Možnosti nápovědy:






- uživatelská příručka
- aplikaci

## 2 Editor

Editor obsahuje seznam uzlů (Node list) a seznam hran (Edge list). Ke každému z těchto seznamů jsou přidružena dvě tlačítka:

-  - Vložení uzlu/hrany do grafy
-  - Odebrání uzlu/hrany z grafu

## 3 Ovládací panel

-  - Reset předchozích kroků.
-  - Krok vpřed.
-  - Dokončení jednoho běhu algoritmu.
-  - Úplně dokončení algoritmu, dojde k zobrazení výsledků.
-  - Krokování algoritmu v pravém panelu a zobrazování změn na grafu.