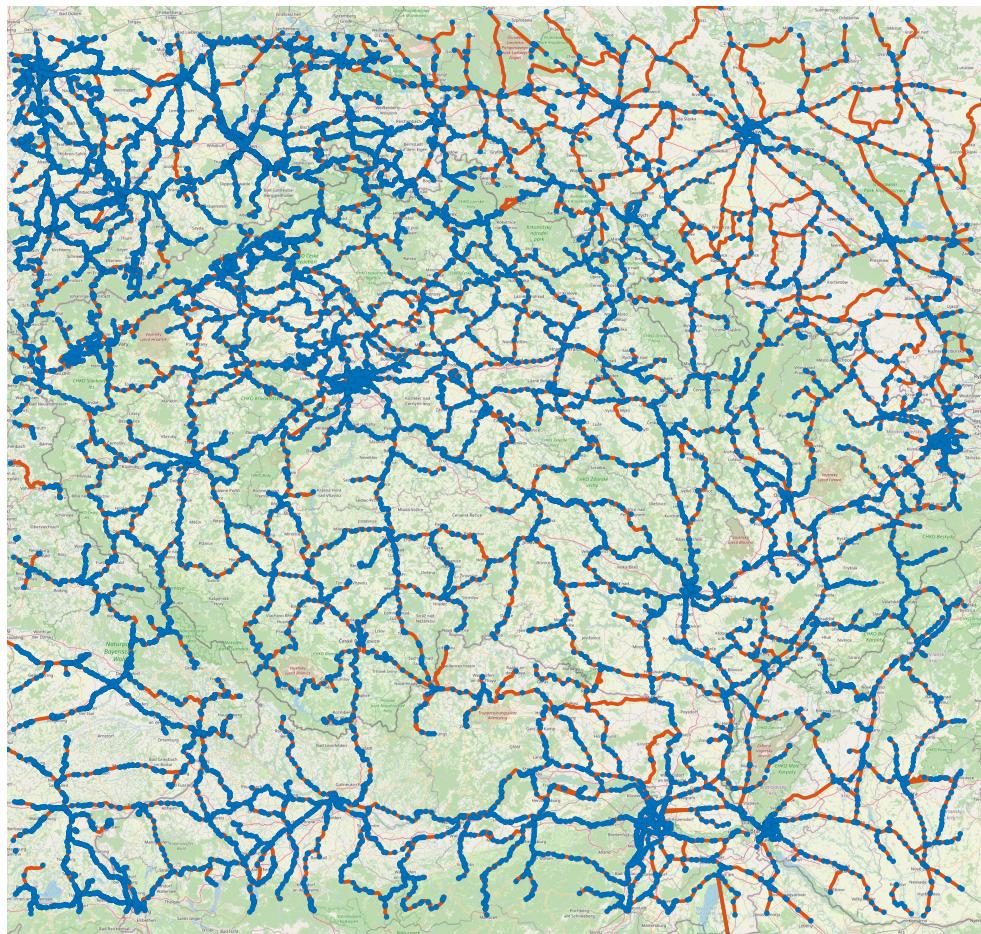


**ZADÁNÍ SEMESTRÁLNÍ PRÁCE**  
**HLEDÁNÍ MINIMÁLNÍ KOSTRY GRAFU**

### Příběh v pozadí

Z šestého patra naší drahé fakulty se k nám dostala data – neorientovaný *nesouvislý* ohodnocený graf představující železniční síť České republiky a blízkého okolí. Pro ukázku je graf vizualizován na obrázku 1. V rámci budoucí výzkumné činnosti je třeba vytvořit výkonnou softwarovou platformu, která tento graf načte do paměti počítače a dovolí nad ním spouštět rozličné analýzy. Pro začátek bychom například rádi věděli kolik metrů kolejí můžeme z aktuální sítě odstranit, aniž by byla omezena dostupnost některých železničních stanic. Nebo by nás zajímalо, jak by vypadala nejkratší možná železniční síť, která by propojovala všechny stanice.



Obrázek 1: Graf železniční sítě vizualizovaný nástrojem QGIS.

## Zadání

Naprogramujte v jazyce ANSI C přenositelnou<sup>1</sup> **konzolovou aplikaci**, která bude zpracovávat grafové struktury uložené v rozsáhlých tabulkových souborech. Již na úvod je třeba upozornit na fakt, že se jedná o data z reálné praxe, která obsahují chyby (více vrcholů grafu se stejným identifikátorem apod.). Po úspěšném načtení grafu do operační paměti bude nad těmito daty spouštěna dvojice vybraných grafových algoritmů. Vaše vstupní i výstupní soubory typu **csv** je možné vizualizovat pomocí aplikace *QGIS*, která je dostupná na webové adrese

<https://qgis.org/en/site/forusers/download.html>.

Program se bude spouštět příkazem **graph.exe** s nesledujícími přepínači:

- v <vrcholy.csv> Povinný parametr specifikující vstupní soubor typu **csv**, který obsahuje informace o vrcholech grafu. Při zadání nevalidní tabulky program vypíše chybovou hlášku „**Invalid vertex file.**\n“ a skončí s návratovou hodnotou 1. Zaručte, aby načtení vrcholů grafu bylo vždy první operací.
- e <hrany.csv> Povinný parametr určující vstupní soubor typu **csv**, jehož řádky definují hrany neorientovaného nesouvislého ohodnoceného grafu. Při zadání nevalidní tabulky program vypíše chybovou hrášku „**Invalid edge file.**\n“ a skončí s návratovou hodnotou 2.
- mst <eout.csv> Nepovinný přepínač, jehož zadání spustí nad načteným grafem algoritmus hledání jeho minimální kostry. Nalezené hrany budou vypsány do souboru daného symbolem <**eout.csv**>. Formát výstupu musí být stejný jako u souboru <**hrany.csv**>, aby bylo možné výsledek vizualizovat. Pořadí výsledných hran je dáno identifikátorem **id** sestupně.
- mrn <mout.csv> Nepovinný přepínač, který spustí generování nové minimální železniční sítě propojující všechny vrcholy grafu. Nová množina hran bude zapsána do souboru určeného symbolem <**mout.csv**>, a to ve stejném formátu jako jsou hrany v souboru <**hrany.csv**>. Pořadí výsledných hran je dáno jejich délkou **clength** sestupně, která bude zaokrouhlena na 3 desetinná místa. Takto seřazeným hranám postupně přidružujte identifikátor náležící množině přirozených čísel. Vzniká tedy *souvislý* graf složený z vrcholů <**vrcholy.csv**> a hran <**mout.csv**>.

Přepínače vždy bezprostředně předcházejí dané umístění souborů. Není-li některý z povinných parametrů uveden, aplikace končí příslušnou chybovou hláškou a návratovou hodnotou. Pokud nebude uveden žádný z nepovinných parametrů, aplikace neprovede žádnou akci a skončí s návratovou hodnotou **EXIT\_SUCCESS**.

Váš program může být během testování spuštěn například takto:

```
graph.exe -e edges.csv -mst emst.csv -mrn emrn.csv -v vertcs.csv
```

Hotovou práci odevzdajte v jediném archivu typu ZIP prostřednictvím automatického odevzdávacího a validačního systému. Postupujte podle instrukcí uvedených na webu předmětu. Archiv nechť obsahuje všechny zdrojové soubory potřebné k přeložení programu, **makefile** pro Windows i Linux (pro překlad v Linuxu připravte souboru pojmenovaný **makefile** a pro Windows **makefile.win**) a dokumentaci ve formátu PDF vytvořenou v typografickém systému **TeX (LATEX)**. Bude-li některá z částí chybět, kontrolní skript Vaši práci odmítne.

<sup>1</sup> Je třeba, aby bylo možné Váš program přeložit a spustit na PC s operačním prostředím Win32/64 (tj. operační systémy Microsoft Windows NT/2000/XP/Vista/7/8/10) a s běžnými distribucemi Linuxu (např. Ubuntu, Mint, OpenSUSE, Debian, atp.). Server, na který budete Vaši práci odevzdávat a který ji otěstuje, má nainstalovaný operační systém Debian GNU/Linux 10 Buster s jádrem verze 4.19.0-11-amd64 a s překladačem gcc 8.3.0.

## Specifikace vstupních dat

Vrcholy a hrany grafů jsou uloženy v oddělených vstupních souborech typu `csv`. Pro oddělení jednotlivých sloupců se využívá znak čárky `,`. Tabulky vrcholů a hran mají následující relevantní atributy:

- WKT** Obsahem tohoto sloupce je geografický popis objektu. Bod je popsán zeměpisnou šířkou  $\varphi$  a délkom  $\lambda$ . Jedná se o polární souřadnice určující bod na povrchu koule s poloměrem  $r = 6\ 371\ 110\ m$ . Při výpočtu vzdálenosti mezi dvěma body tedy prostá Eukleidovská vzdálenost stačit nebude, ale bude muset být využita tzv. *ortodroma*. Hrany (železnice) jsou dány jako lomené čáry procházející sérií bodů. Je třeba věnovat pozornost tomu, že pro oddělení jednotlivých bodů lomené čáry se opět využívá znak čárky `,`. Jedná se však stále o jeden atribut. Právě díky tomuto sloupci je možné graf vykreslit v mapových podkladech pomocí softwaru QGIS.
- id** Účelem tohoto sloupce je jednoznačná identifikace vrcholu nebo hrany grafu. Kvůli chybám při tvorbě dat ovšem tento identifikátor není unikátní. Při načítání dat tedy berte pouze první výskyt vrcholu nebo hrany s daným identifikátorem a ostatní ignorujte.

Atributy důležité pro konkrétní tabulky jsou uvedeny v textu níže. Ostatní atributy sice nejsou pro grafové algoritmy relevantní, ale jejich zahodení je v případě spuštění s přepínačem `-mst` nepřijatelné. Hodnoty všechny vstupních řádků musejí být přítomny i ve výstupních souborech, a to v nezměněné podobě. Při použití s přepínačem `-mrn` nechte sloupce, `nation` a `cntryname` prázdné.

### Tabulka vrcholů grafu

Tabulka vrcholů obsahuje uspořádanou množinu sloupců **WKT**, **id** a **sname**. Pokud na vstupu bude zadána tabulka s jinými atributy, aplikace skončí s chybovým hlášením. Rovněž při tvorbě nového souboru je třeba tuto strukturu dodržet.

### Tabulka hran grafu

Tabulka hran obsahuje uspořádanou množinu sloupců **WKT**, **id**, **nation**, **cntryname**, **source**, **target**, **clength**. Pokud na vstupu bude zadána tabulka s jinými atributy, aplikace skončí s chybovým hlášením. Rovněž při tvorbě nového souboru je třeba tuto strukturu dodržet.

Hrana je definována mezi vrcholy jejichž identifikátory jsou uvedeny ve sloupcích **source** a **target**. Ohodnocení hrany je dána vzdáleností mezi těmito vrcholy a je k dispozici ve sloupci **clength** vyjádřená v metrech.

Opět je zde potřeba sledovat chyby. V mnoha případech mezi dvěma vrcholy existuje více hran, které jsou samozřejmě stejně délky, ale různého identifikátoru. V tomto případě postupujte obdobně jako u problému s nejednoznačností identifikátoru **id** u vrcholů grafu, tzn. akceptujte pouze první výskyt hrany mezi vrcholy  $\{u, v\}$  a ostatní ignorujte. Rovněž v datech existují případy existence hrany  $\{u, u\}$ , tj. hrana ideálně nulové délky. Tyto hrany rovněž ignorujte.

## Užitečné techniky a odkazy

Uvedené techniky je možné (ale nikoliv nezbytně nutné) využít při řešení úlohy. Protože se jedná o postupy standardní, lze k nim nalézt velké množství dokumentace:

1. **Minimální kostra grafu**

[https://en.wikipedia.org/wiki/Minimum\\_spanning\\_tree](https://en.wikipedia.org/wiki/Minimum_spanning_tree)

2. **Kruskalův algoritmus**

<https://www.algoritmy.net/article/1417/Kruskaluv-algoritmus>

3. **Výpočet ortodromy**

<https://www.aldebaran.cz/~brichnac/skola/ortodroma.pdf>

**Řešení úlohy je zcela ve vaší kompetenci** – zvolte takové algoritmy a techniky, které podle vás nejlépe povedou k cíli. Pokud bude něco nejasného, neváhejte mne kontaktovat.