

První praktický domácí úkol

Sestavení a ovládání

Programy pro řešení zadaných úloh jsou napsány v jazyce Java. Před jejich spuštěním je potřeba spustit JVM a programy zkompileovat. O tyto věci se stará make, který se ve složce nachází. Program pro řešení úlohy 1, resp. úlohy 2, se po kompilaci nazývá `"/ulohy/src/uloha1"`, resp. `"/ulohy/src/uloha2_1"`. Oba dva čtou ze standardního vstupu a vypisují na standardní výstup. Dají se tedy použít jako

```
cat file | java -cp ./ulohy/src uloha1 (resp. uloha2_1).
```

Popis výstupu

Úloha 1

Pro každý vrchol si vytvořím novou proměnnou, která mi bude reprezentovat pořadí vrcholu v uspořádání. Všechna "v" jsou nastavena na nezápornou hodnotu, protože nemůžu mít -14. vrchol v pořadí. Pro vrcholy, mezi kterými vede hrana, vypisuji podmínku, která zaručí, že dva sousední vrcholy budou mít různé pořadí minimálně o 1. Snažím se najít co nejmenší hodnotu pořadí, minimalizuji tedy proměnnou "m", která je větší nebo rovna pořadím všech vrcholů.

Úloha 2

Pro každou hranu si vytvořím proměnnou x, která je nezáporná a reprezentuje mi, že hrana bude, kvůli tomu že tvořila 3- nebo 4- cyklus, odebrána. Pro hrany i jejich váhy si vytvořím množiny jejich hodnot. Pomocí podmínek "hranovy_zakaz_3(4)" nastavím pro aspoň jednu hranu hodnotu proměnné "x" na 1. Snažím se minimalizovat váhu odebraných hran, to mi zaručí podmínka "pVO".

Co když není řešení?

Úloha 1

Když řešení existuje, glpsol hodnotu řešení vypíše. Když řešení neexistuje, znamená to, že graf obsahuje cyklus. Řešič potom tedy nedokáže nastavit hodnotu u takových vrcholů, proto selže.

Úloha 2

Myslím si, že tato úloha má vždy nějaké přípustné řešení. Bude jej tvořit odebrání všech hran, které odstraní všechny cykly. K optimálnímu se pak dá dostat rozumnější volbou odebíraných hran.