

Domácí úkol 12: Zážiteková jízda Brnem

Poslední domácí úkol budete implementovat v Prologu. Jako u předchozích velkých úkolů, i za tento můžete získat až dva body do hodnocení.

Motivační příběh

Blíží se konec semestru a s ním i zkuškové období, které musí být jaksepatří zakončeno důstojnou oslavou. Parta studentů blíže nespecifikované školy se proto rozhodla, že pokud se jí podaří složit všechny zkoušky, uspořádá pořádné kolečko po brněnských hospodách, restauracích, barech a klubech.

Takové kolečko se však obtížně absolvuje pěšky.¹ Naši studenti si proto vytipovali několik zastávek brněnské hromadné dopravy, na kterých se určitě musí stavit (ať už kvůli jídlu, pití nebo přibrání kamarádů). Protože se však při takovém tažení může stát ledasco, je zcela nežádoucí se na jedné zastávce objevit vícekrát. Výjimkou je hlavní nádraží, na kterém se všichni sejdou a ze kterého na konci cesty odjedou domů nočním rozjezdem.

Vaším úkolem je pomoci jim tuto cestu naplánovat.

Kostra řešení

Ze studijních materiálů si stáhněte kostru řešení. Ta je tentokrát obsáhlejší, než tomu bylo u minulých domácích úloh. K dispozici máte zejména níže zmíněné predikáty.

- `c/2`, jímž modelujeme *jednosměrné* spoje mezi zastávkami, přičemž nás nezajímá, zda se jedná o spoj šalinou, trolejbusem, autobusem, nebo pěší. Nijak nezohledňujeme ani počet linek, které obě zastávky spojují. Pokud si budete chtít přidat (třeba pro účely testování) vlastní zastávky a spoje mezi nimi, stačí vám přidat příslušné fakty `c(zastavkaA, zastavkaB)`. Přidávejte vždy **jen jeden směr** spojení; ten druhý řeší následující predikát.
- `connexion/2`, který „zobousměrňuje“ `c/2`. Jinými slovy `connexion(A, B)` uspěje, pokud mezi zastávkami A a B jezdí přímý spoj. V úloze nepočítáme s existencí jednosměrných linek ani zastávek.
- `stops/1`, jenž do volné proměnné unifikuje seznam všech zastávek. Tento seznam se získává z predikátů `c/2`, bude proto fungovat i tehdy, budete-li přidávat nové zastávky.
- `dump_dot/1`, pomocí něžž můžete celou mapu s vyznačenou trasou výletu exportovat do formátu `dot`. Více informací naleznete v sekci Zobrazení výsledků.

Ostatní predikáty jsou používány predikátem `dump_dot/1` a není nutné se jimi zabývat. Pro zachování funkčnosti exportu do `dotu` byste je však neměli měnit.

Zadání

Naprogramujte predikát `tour/1`, který do volné proměnné unifikuje posloupnost (seznam) zastávek takovou, že platí:

- První zastávkou je ta, pro niž platí predikát `trip_start/1`. V kostře se jedná o Hlavní nádraží (atom `hlavni_nadrazi`). Můžete předpokládat, že tento predikát splňuje jen jedna zastávka, a tedy tento uspěje právě jednou.
- V seznamu se nachází právě ty zastávky, které existují v dopravní síti, a každá právě jednou. Jinými slovy: pokud platí `stops(X)`, `tour(Y)`, pak Y je permutací X. Zejména tedy první zastávka **není** zopakována na konci seznamu.
- Mezi každými dvěma po sobě jdoucími zastávkami musí existovat přímý spoj. Řečí predikátů: pro každé dva sousední prvky seznamu musí uspět `connexion/2`.
- Mezi poslední a první zastávkou musí existovat přímý spoj.

Existuje-li vícero takovýchto posloupností, musí predikát postupně (tj. ve `swipl` po stisknutí středníku) objevit každou z nich. Pakliže taková posloupnost neexistuje, predikát neuspěje (tj. `swipl` vypíše na dotaz `false`), a žádná oslavná jízda se nekoná.

¹Až na trasu mezi náměstími Svobody a Šilingrovým, ta je totiž lemována příjemným množstvím podniků.

Predikát má být použitelný v módu ?; měl by tedy nejen dokázat najít vhodné trasy, ale také umět ověřit, zda zadaná trasa vyhovuje výše popsaným podmínkám.

Váš predikát musí fungovat obecně, tedy na jakékoli dopravní síti popsané predikáty `c/2` a `trip_start/1`. Rozhodně nebude uznáno řešení, jež v sobě bude mít natvrdo naprogramované všechny trasy vyhovující zadání.

Zobrazení výsledků

Graficky

Jelikož se může v průběhu práce na úloze hodit názorně vidět, co váš predikát počítá, máte v kostře k dispozici predikát `dump_dot/1`, jenž výsledek vyexportuje do formátu `dot`.

Argumentem predikátu je prefix názvu souboru zadaný jako řetězec (například `dump_dot("trip_")`). Výsledkem je na první pohled jen `true.`, ale jako vedlejší efekt jsou vytvořeny soubory se jmény `<prefix><n>.dot`. Ty následně můžete programem `sfdp` převést na obrázek, v němž je vykreslená celá dopravní síť a v ní červeně zvýrazněný `n`-tý výsledek predikátu `tour/2`. Program `sfdp` je k dispozici minimálně na Linuxu v balíku `Graphviz` a je možné jej spustit na fakultních strojích `nymfe`. Případně můžete využít některé online kreslírky pro `dot`².

Příklady použití `sfdp`:

- `sfdp -O -Tsvg tour.dot` vytvoří vektorový obrázek `tour.dot.svg`
- `sfdp -Tsvg tour.dot > brno.svg` vytvoří vektorový obrázek `brno.svg`
- `sfdp -O -Tpng tour.dot` vytvoří rastrový obrázek `tour.dot.png`

V interpretru

Výsledkem výpočtu je seznam, který je poměrně dlouhý, a `swipl` ho proto nezobrazí celý. Toto chování můžete obejít například použitím vestavěného predikátu `print/1`. Dotaz poté bude vypadat následovně:

```
?- tour(X), print(X).
```

Druhá možnost, jak si od interpretru vyžádat vypsání celého seznamu, je použitelná pouze v případě, že vám `swipl` nabízí dotázat se na další řešení. Potom stačí stisknout místo středníku nebo tečky klávesu `w`.

Odevzdávání a bodování

Na rozdíl od předchozích velkých úkolů budete tentokrát své dílo vkládat do odevzdávacího systému. Odevzdávejte jeden soubor s příponou `.pl`, který obsahuje okomentovaný zdrojový kód. Do komentářů slovně popište, jak jste při řešení příkladu postupovali, a případně také uveďte zdroje, kterými jste se inspirovali. Komentáře nemusíte psát anglicky. Čas na vypracování úkolu je do středy **20. prosince** (včetně).

Za celou úlohu můžete dostat dva body. První bod je za funkčnost kódu, druhý bod je za jeho čitelnost, smysluplnost a slovní popis. Je možné získat i desetinné body, pokud bude řešení jen částečně správné. Oba body uděluje cvičící.

²Např. <http://viz-js.com>. Drobnou nepříjemností je, že výsledek není tak hezký, jako při použití `sfdp`, pro ilustraci je však dostatečný.

Příloha: Plánek sítě

Aby se vám zadaná síť hromadné dopravy lépe představovala, níže najdete její grafické vyobrazení.

