

**Warning**

This page is located in archive.

## 4. domácí úloha: Paralelizace provádění databázových dotazů

- Předloha: `pdv-04database.zip` [<https://pdv.pages.fel.cvut.cz/pdv-private/hw/pdv-04database.zip>]
- Do BRUTE odevzdávejte zip archiv obsahující soubor `query.h`.

Vyhodnocování dotazů v databázových systémech často obnáší procházení velkého množství dat. Aby bylo vyhodnocování efektivní, databáze často odhadují jaká posloupnost operací vede k nejrychlejšímu výsledku. V této domácí úloze si zkusíte rozhodnout (a vyzkoušet), jaký přístup k vyhodnocení dotazů je nejvhodnější – při čemž se zaměříme na to, jak vyhodnocování dotazu co možná nejlépe paralelizovat.

Vaším úkolem bude vyhodnotit sadu dotazů na jednu tabulku. U každého dotazu bude Vaším úkolem rozhodnout, zda existuje alespoň jeden záznam v tabulce, pro který námi zadaná podmínka platí. My se Vás budeme ptát na pravdivostní hodnotu konjunkce (respektive disjunkce) těchto dílčích dotazů. V řeči SQL dotazů se tedy jedná (schématicky) o dotazy typu:

```
SELECT 1
WHERE
  EXISTS (SELECT * FROM data_table WHERE predicates[0](...)) AND
  EXISTS (SELECT * FROM data_table WHERE predicates[1](...)) AND
  EXISTS (SELECT * FROM data_table WHERE predicates[2](...)) AND
  ...
```

a

```
SELECT 1
WHERE
  EXISTS (SELECT * FROM data_table WHERE predicates[0](...)) OR
  EXISTS (SELECT * FROM data_table WHERE predicates[1](...)) OR
  EXISTS (SELECT * FROM data_table WHERE predicates[2](...)) OR
  ...
```

V případě, že chceme vyhodnocování takových dotazů paralelizovat, máme k dispozici dva základní dekompoziční přístupy:

1. Vyhodnocování jednoho poddotazu `exists select * from data_table where predicates[0](...)` přidělíme jednomu vláknu (tj. každý poddotaz bude zpracovávat právě jedno vlákno sekvenčním přístupem)
2. Zparalelizujeme vyhodnocování každého poddotazu (tj., vlákna si rozdělí řádky v tabulce, na kterých budou testovat pravdivost predikátu).

Vaším úkolem bude zamyslet se nad tím, který z těchto přístupů bude vhodnější pro který typ dotazu za předpokladu, že konjunkce (respektive disjunkce) je pravdivá. Následující otázky Vám mohou pomoci tento problém vyřešit:

1. Je pro daný typ dotazu (a danou očekávanou pravdivostní hodnotu `true`) důležité identifikovat alespoň jeden pravdivý dotaz v jedné vybrané tabulce? nebo
2. Je potřeba prohledávat více tabulek?

V některých situacích byste měli být schopní rozhodnout, jaká je pravdivostní hodnota dotazu bez toho, abyste museli prohledat otestovat všechny predikáty na všech záznamech v tabulce. V takové situaci je vhodné prohledávání ukončit.

Vaším úkolem je doimplementovat těla metod `is_satisfied_for_all` a `is_satisfied_for_any` v souboru `query.h`. Tyto metody mají za úkol rozhodnout, zda konjunkce (respektive disjunkce) dílčích dotazů je splněná.

[courses/b4b36pdv/tutorials/hw\\_04.txt](#) · Last modified: 2024/03/12 14:51 by [kafkamatt](#)