



Fórum studentů MFF UK

Fórum pro všechny studenty matematicko-fyzikální fakulty UK, informatiky, fyziky i matematiky

[Přejít na obsah](#)

[Pokročilé hledání](#)

- [Obsah fóra](#) < [Informatika LS](#) < [Výuka LS 2. ročník](#) < [PRG005 Neprocedurální programování](#)
- [Změnit velikost textu](#)
- [Napsat e-mail](#)
- [Verze pro tisk](#)
- [FAQ](#)
- [Registrovat](#)
- [Přihlásit se](#)

Zkouška 2. 6. 2015 (Dvořák, Hric)

[Odeslat odpověď](#)

Příspěvek: 1 • Stránka 1 z 1

- [Ohlásit tento příspěvek](#)
- [Odpovědět s citací](#)

Zkouška 2. 6. 2015 (Dvořák, Hric)

od [Erim](#) » 2. 6. 2015 21:52

Prolog:

1) Definujte predikát `tranverse(+Strom,-OhodnocenýStrom)`, který zkopíruje strukturu stromu `Strom` do `OhodnocenýStrom` s tím, že ke každému vrcholu přidá číslo `N`, které znamená pořadí v `preOrder` průchodu a číslo `M`, které znamená pořadí v `postOrder` průchodu. Ideálně jedním průchodem stromem.

Kód: [Vybrat vše](#)

```
tranverse(t(t(nil,l,nil),v,t(nil,p,nil)),X).  
X = t(t(nil,l-2-1,nil),v-1-3,t(nil,p-3-2,nil))
```

2) Definujte predikát `atmost2(+seznam,-seznam2)`, který dostane seznam ohodnocených binárních proměnných a vrátí seznam všech možných ohodnocení stejných proměnných, které se od původního seznamu liší v maximálně 2 proměnných.

Kód: [Vybrat vše](#)

```
atmost2([x-true,y-false],X).
X = [[x-true,y-false],[x-false,y-false],[x-false,y-true],[x-true,y-true]]
```

Haskell:

3) Napište funkci, která převede slovo "abbbccac" na seznam, kde jsou po sobě jdoucí stejné znaky sjednoceny do dvojice (znak, početVýskytů). K reprezentaci výsledku použijte Either viz příklad:

Kód: [Vybrat vše](#)

```
rle::Eq a=>[a]->[Either a (a,Int)]
rle [a,b,b,b,c,c,a,c] = [Left a, Right (b,3), Right (c,2), Left a, Left c]
```

Napište, jakou definici bude mít inverzní funkce, která naopak převede kontrahovaný seznam na původní slovo a potom ji napište s pomocí concat a map.

4) Napište fold pro binární stromy. Napište funkci využívající váš fold, která pro daný binární strom spočítá počet jeho listů.

Kód: [Vybrat vše](#)

```
data BTree a = Nil | Vertex (BTree a) a (BTree a)
fold::b->(b->a->b->b)->BTree a->b
```

Velká úloha:

Hypergraf $H = (V, E)$, kde V je množina vrcholů a E je libovolná podmnožina vrcholů (ne nutně dvouprvková). Lineární hypergraf je pak hypergraf, jehož žádné 2 hrany se neprotínají ve víc než jedno bodě. Erdősova hypotéza říká, že každý hypergraf se dá hranově dobře (stejně obarvené hrany mají prázdný průnik) obarvit pomocí nejvýše $|V|$ barev. Napište predikát/funkci, který(á) generuje všechny lineární hypergrafy a otestujte pro každý, zda ho lze obarvit maximálně n barvami. U obecných hypergrafů je hledání takového obarvení NP-Úplný problém, ale pro lineární se předpokládá (nikdo Erdősovu hypotézu za 40 let nevyvrátil), že takové obarvení existuje. Použijte proto pro hledání toho obarvení nějakou heuristiku.

Já jsem jako heuristiku jednak setřídil hrany, které se chystám obarvit podle počtu vrcholů v nich obsažených sestupně a pak jsem ještě střídal použité barvy (místo abych jednou barvou barvil vše co můžu a měnil barvu až když nemůžu, tak jsem právě použitou barvu dával na konec seznamu pro barvení další hrany). Tyto heuristiky se mu líbily, neříkal, jestli doufal v lepší.

[Erim](#)

Matfyz(ák|ačka) level I

Příspěvky: 11

Registrován: 18. 12. 2014 12:28

Typ studia: Informatika Bc.

[Nahoru](#)

[Odeslat odpověď](#)

Příspěvek: 1 • Stránka 1 z 1

[Zpět na PRG005 Neprocedurální programování](#)

Přejít na:



Přejít

Kdo je online

Uživatelé procházející toto fórum: Žádní registrovaní uživatelé a 1 návštěvník

- [Obsah fóra](#)
- [Tým](#) • [Smazat všechny cookies z fóra](#) • Všechny časy jsou v UTC + 1 hodina

POWERED_BY

Český překlad – phpBB.cz