

Závazné pokyny pro funkcionální projekty FLP 2018/2019

Libor Škarvada
iskarvada@fit.vutbr.cz

1 Požadavky na zdrojový text

Funkcionální část bude vypracována v jazyce Haskell. Pro opravování bude použit kompilátor GHC (Glasgow Haskell Compiler) s dalšími nástroji a knihovnami, které jsou součástí Haskell Platform. Pro vývoj doporučujeme použít nejnovější stabilní verzi Haskell Platform (Windows/Linux), nebo verzi dostupnou v repozitáři vaší distribuce (Linux). Své řešení můžete otestovat na serveru `merlin`, mějte ale na paměti, že zde není dostupná poslední verze Haskell Platform. Případné problémy způsobené rozdílnými verzemi softwaru budou řešeny samostatně a nikdo kvůli nim nepřijde o body (různé verze na linuxových distribucích atd.).

- Definice každé hodnoty na globální úrovni a každého typu a typové třídy začíná stručným a výstižným komentářem popisujícím danou hodnotu, typ nebo třídu.
- Definice hodnot obsahují typovou anotaci.
- Volte vhodné datové typy, a to i s ohledem na efektivnost algoritmu.
- Pokud možno se vyhýbejte parciálním funkcím, totální funkce jsou bezpečnější.
- Nepoužívejte hluboce zanořované konstrukce (if-výrazy, case, where apod).
- Nepoužívejte nestandardní funkce a akce, které obcházejí bezpečné otypování nebo obcházejí monadické zapouzdření vedlejších efektů – vesměs jsou to hodnoty obsahující ve svém jméně slovo „unsafe“.
- Neopakujte stejné části kódu, dodržujte princip DRY.
- Použijte nástroj `hlint`, který vás upozorní na nedostatky v kódu a na jeho části, které by šly napsat lépe.
- Pište přehledně, odsazujte jednotně a konsistentně, odsazujte mezerami, nedělejte řádky delší než 100 znaků.

Konkrétní požadavky budou specifikovány v příslušné variantě zadání.

2 Odevzdávání

Svá řešení odevzdávejte v předepsaném tvaru (zkomprimovaný archiv zip, viz Závazné pokyny pro projekty) prostřednictvím WIS do datového skladu předmětu FLP.

3 Implementace

Součástí řešení bude soubor **Makefile** (projekty budou překládány pomocí **gmake**), ve kterém lze nastavit případné parametry překladu. Pokud soubor pro sestavení cílového programu nebude obsažen nebo se na jeho základě nepodaří sestavit cílový program, nebude projekt hodnocen. Cílový program pojmenujte podle pokynů v příslušné variantě zadání.

Úvodní řádky **všech** zdrojových textů musí obsahovat název projektu, login a jméno autora.

Pokud není řečeno jinak, vytvořte program ve formě konzolové aplikace. Nepožaduje se grafické uživatelské rozhraní.

4 Samostatné zadání

Pokud máte zájem pracovat na něčem zajímavějším než jsou generická společná zadání, použijte variantu **custom**, formulujte zadání a kontaktujte cvičícího.

5 Konzultace

Budou probíhat osobní formou na cvičení a po předchozí domluvě mailem. Cvičící může rovněž v rámci svých časových možností poskytovat konzultace prostřednictvím fóra předmětu, které ovšem prioritně slouží pro komunikaci mezi studenty. Dále lze využít fakultní elektronickou poštu.

Konzultace k funkcionálním projektům v letním semestru 2019 poskytuje L. Škarvada.

6 Ostatní

Jazyk, jímž je napsán soubor README, komentáře v kódu, popisy testů a další texty, si student může zvolit mezi češtinou, slovenštinou a angličtinou.