



3. Text, shell a regulární výrazy



Zpracování textu v shellu a regulární výrazy

Domácí příprava

Zadání úlohy

Materiály

Domácí příprava na další cvičení

Zpracování textu v shellu a regulární výrazy

Domácí příprava

Nastudujte **použití regulárních výrazů a nástroje pro zpracování textu** (alespoň zběžně `grep`, `sed` a `tr`).

Zadání úlohy

Vytvořte skript v jazyce Bash podle následujících požadavků:

- bude uložen v souboru s příponou `.sh`.
- jeho funkce bude určena následujícími přepínači:
 - `-h` vypíše stručnou nápovědu a ukončí skript. Nápovědu můžete vypsát na standardní nebo chybový výstup.
 - `-a` vypíše na standardní výstup všechny soubory PDF z aktuálního adresáře. Poznáte je podle přípony `.pdf`; na velikosti písmen nezáleží. Každý soubor bude na samostatném řádku, řazení podle abecedy.
 - `-b` přečte text ze standardního vstupu a vypíše na standardní výstup všechny řádky začínající číslem, bez tohoto čísla. Uvažujte pouze celá čísla v dekadickém zápisu, která mohou začínat znaménkem `+` či `-`. Případné desetinné tečky/čárky už jsou součástí dalšího textu a budou vypsány.

Nápověda: Použijte nástroj `sed`.

- `-c` přečte text ze standardního vstupu a najde v něm všechny věty. Ty pak vypíše na standardní výstup každou na nový řádek. Za větu považujte nejdelší textový řetězec mezi velkým písmenem a znakem ukončujícím větu (`!.?`), který neobsahuje

další znak ukončující větu. Věty typu "Král Karel I. Veliký se narodil 2. dubna." nebudou součástí testovacích skriptů. Bude-li věta rozdělena na více řádků, skript nahradí znak konce řádku mezerou (nápopověda: použijte nástroj `tr`).

- `-d <prefix>` (nepovinné) přečte zdrojový kód jazyka C ze standardního vstupu a provede změnu názvů souborů v direktivě `#include` ve všech souborech tak, že `<prefix>` přidá na začátek názvu souboru. Uvažujte obě varianty direktivy `#include` – jak pro soubory z aktuálního adresáře (`"foo.h"`) tak pro systémové knihovny (`<foo.h>`). Například pro `-d lib/`, bude `#include "foo.h"` změněno na `#include "lib/foo.h"` . Změny se provedou i uvnitř komentářů a řetězců.
- Pokud není zadán platný přepínač, skript vypíše nápopovědu a skončí s návratovým kódem 1, v ostatních případech skončí s návratovým kódem 0.
- V případě zadání více platných přepínačů, skript vykoná funkci podle prvního z nich.

Materiály

- [Základy regulárních výrazů](#)

Domácí příprava na další cvičení

Předpokládáme, že máte základní znalosti jazyka C a víte, jak funguje překlad ze zdrojových kódů jazyka C do binární spustitelné aplikace (v obecném případě, kdy je zdrojových souborů více).

Dále byste měli mít alespoň minimální povědomí o použití překladače `gcc` a jeho [základních parametrech](#).

Nastudujte si použití nástroje `make` pro překlad programu v jazyku C/C++: [make](#).