



SP z predmetu Algoritmy a údajové štruktúry 1 - Prvá úroveň

Zadanie

Načítajte údaje o obciach, správnych obvodoch obcí s rozšírenou pôsobnosťou (ďalej len SOORP, SOORP predstavujú analógiu k okresom v Slovenskej republike) a krajoch Českej republiky do vami zvolených sekvenčných údajových štruktúr. Po načítaní teda budú načítané 3 sekvenčné štruktúry, 1 pre obce, 1 pre SOORP a 1 pre kraje.

Vstupné údaje nájdete v súbore **CR.csv**. Súbor je kódovaný v UTF-8. Popis jednotlivých stĺpcov spolu s prvými 10 záznamami môžete nájsť v súbore **hlavicka.pdf**. Vstupný súbor si môžete ľubovoľne upraviť avšak nesmú sa stratiť požadované údaje (nemôže teda „zmiznúť“ žiadna obec ani relevantný stĺpec, môžete si však súbory prekonvertovať do iného formátu alebo vopred odstrániť nepotrebné údaje). Ak vstupné údaje upravíte, popíšte v dokumentácii túto úpravu.

Pre všetky územné jednotky (pod územnou jednotkou rozumieme obec, SOORP a kraj) načítajte a evidujte nasledovné údaje:

- Názov územnej jednotky.
- Kód územnej jednotky.
- Názov územnej jednotky.

Pre obce ďalej načítajte a evidujte nasledovné údaje:

- Typ obce.
- Katastrálna výmera v hektároch.
- Počet obyvateľov (celkom, vo veku 0-14 rokov a vo veku 65 rokov a viac).
- Informácia o existencii kanalizácie (s prípadným napojením na ČOV).
- Informácia o existencii vodovodu.
- Informácia o plynofikácii.

Na vypracovanie tejto (a práve tejto) úrovne je možné využiť existujúcu knižnicu údajových štruktúr (napr. `std::vector`).

Naprogramujte univerzálny algoritmus, ktorý umožní spracovať tie údaje z údajovej štruktúry, ktoré spĺňajú zadaný predikát¹.

Pod spracovaním sa v našom prípade bude rozumieť vloženie údajov do inej sekvenčnej priamej štruktúry. Naplnenú priamu sekvenčnú štruktúru umožnite vypísať.

Budeme využívať predikáty, ktoré vrátia pravdu, ak:

- `startsWithStr`: názov danej územnej jednotky začína na zadaný reťazec.
- `containsStr`: názov danej územnej jednotky obsahuje zadaný reťazec.

Prvú úroveň teda môžeme konkretizovať na: Naprogramujte algoritmus, ktorý vyfiltruje obce/SOORP/kraje, ktorých názov zahŕňa/začína na odovzdaný reťazec, do inej sekvenčnej údajovej štruktúry, ktorú potom vypíšete. Dbajte na čo najväčšiu univerzálnosť algoritmu (bude sa využívať aj neskôr).

¹ Ako predikát sa označuje funkcia, ktorá preberá testovaný objekt a vracia pravdivostnú hodnotu v závislosti od toho, či daný objekt testom prešiel alebo nie.



Tipy

- Algoritmus zapúzdrite do samostatného objektu.
- Predikáty (v našom prípade `startsWithStr` resp. `containsStr`) riešte formou lambda funkcií (preferované) alebo virtuálnych metód.
- Do algoritmu posielajte namiesto údajovej štruktúry pár iterátorov (v tomto prípade `begin` a `end` z danej údajovej štruktúry). Aby ste nemuseli uvádzať presný typ iterátora, využite šablóny a štandardom definované metódy.
- Vstupný súbor si môžete upraviť ak nestratíte požadované informácie. Takúto úpravu potom popíšete v dokumentácii.

Hodnotenie

Demonštrácia funkčnosti (nepovinné)

- 5 bodov.
- Pri demonštrácii funkčnosti je nutné uvažovať s diakritikou!
- Prebieha v 6. týždni semestra.
- Získanie bodov nie je podmienené vypracovaním dokumentácie.
- Získanie bodov je kvôli kontrole originality podmienené nahraním práce do systému Moodle.

Pri obhajobe

- Pri obhajobe nie je nutné uvažovať s diakritikou.
- Údaje o územných jednotkách sú v osobitných údajových štruktúrach (podľa typu územnej jednotky). Načítanie každej štruktúry vyžaduje jediné otvorenie súboru a spracovanie každého riadku súboru maximálne raz – max 3b.
- Algoritmus je v na to dedikovanom samostatnom objekte (teda nie je v nečlenskej metóde alebo členskej metóde iného objektu, ktorý ho volá) – max 3b.
- V algoritme je možné bez úpravy kódu samotného algoritmu zmeniť predikát pomocou
 - virtuálnej metódy – max 2b.
 - lambda funkcie – max 3b.
- Algoritmus pracuje nad údajmi, ktoré sú zadané formou
 - údajovej štruktúry (napr. `std::vector`) – max 2b.
 - páru iterátorov – max 3b.

Dokumentácia

Dokumentácia sa odovzdáva spolu s finálnou verziou semestrálnej práce na konci semestra. Dokumentácia musí byť vypracovaná podľa zverejnených požiadaviek (dokument SP popis).

Okrem povinných častí doplňte dokumentáciu o nasledujúce časti:

- **UML diagram aktivít** Vami zostaveného **univerzálneho algoritmu** – max 2b.
- **Programátorská príručka**, teda ako používať Váš univerzálne navrhnutý algoritmus v tu popísaných prípadoch a ako postupovať v prípade jeho budúceho rozšírenia o iný predikát – max 1b.
- Úpravy vstupných súborov ak ste nejaké vykonali.