**ŽILINSKÁ UNIVERZITA V ŽILINE**

FAKULTA RIADENIA A INFORMATIKY

**Systém pre geodetov**

SEMESTRÁLNA PRÁCA

Vypracoval: **Peter Szathmáry**

Študijná skupina: **5ZIS12**

Predmet: **Algoritmy a údajové štruktúry 2**

Cvičiaci: **Ing. Peter Jankovič, PhD.**

# Obsah

Obsah

[Obsah 1](#_Toc153748007)

[Zadania práce 2](#_Toc153748008)

[Návrh štruktúry 4](#_Toc153748009)

[Operácie údajovej štruktúry 5](#_Toc153748010)

[Vyhľadaj dáta 5](#_Toc153748011)

[Uprav dáta 5](#_Toc153748012)

[Vlož dáta 5](#_Toc153748013)

[Vymaž dáta 6](#_Toc153748014)

# Zadania práce

Problém semestrálnej práce nadväzuje na prvú semestrálnu prácu. Súkromná spoločnosť

vyhlásila súťaž na vytvorenie systému pre geodetov, ktorý bude použitý na špecializovanom PDA

zariadení. Firma sa chce na trhu presadiť s najlacnejším, spoľahlivým a zároveň rýchlym

zariadením.

Zariadenia musia umožňovať presné zameranie jednotlivých nehnuteľností podľa signálu

GPS a dát získaných z centrálneho systému, ktoré budú na zariadení uchovávané. Vytvorte

demonštračnú verziu softvéru pre informačný systém záznamov o nehnuteľnostiach. Pre

zjednodušenie tejto demo verzie programu bude systém evidovať pre každú parcelu/nehnuteľnosť

iba najmenší ohraničujúci pravouholník.

Pre každú nehnuteľnosť evidujte v dynamickom hešovacom súbore nasledovné údaje:

• súpisné číslo (celé číslo)

• identifikačné číslo nehnuteľnosti (jedinečné celé číslo)

• popis (reťazec max. dĺžky 15 znakov)

• záznam dvoch pozícií GPS ohraničujúcich nehnuteľnosť

• záznamy o parcelách na ktorých sa nachádza (max. 6 záznamov) - identifikačné čísla parciel

Pre každú parcelu evidujte v dynamickom hešovacom súbore nasledovné údaje:

• identifikačné číslo parcely (jedinečné celé číslo)

• popis (reťazec max. dĺžky 11 znakov)

• záznam dvoch pozícií GPS ohraničujúcich parcelu

• záznamy o nehnuteľnostiach na ktorých sa nachádza (max. 5 záznamov) - identifikačné čísla

nehnuteľností

Dáta sú zároveň vložene aj do Vami implementovaného Quad stromu.

V tejto demonštračnej verzii spoločnosť požaduje, aby boli k dispozícii nasledujúce operácie:

• **Vyhľadanie nehnuteľnosti v dynamickom hešovacom súbore** **podľa identifikačné čísla**

**nehnuteľnosti** - vyhľadajú a vypíšu sa všetky informácie o nehnuteľnosti vrátane podrobných

informácií o parcelách na ktorých sa nachádza.

• **Vyhľadanie parcely v dynamickom hešovacom súbore** **podľa identifikačné čísla parcely**

- vyhľadajú a vypíšu sa všetky informácie o parcele vrátane podrobných informácií

o nehnuteľnostiach ktoré sa na nej nachádzajú.

• **Pridanie** – na základe vstupných údajov pridá záznam do evidencie (užívateľ nezadáva

identifikačné číslo). Pomocou intervalového vyhľadávania a quad stromu sa automaticky

naplní zoznam prepojených nehnuteľností/parciel. Záznam je samozrejme pridaný aj do quad

stromu.

• **Vyradenie** – na základe **identifikačné čísla** vyradí záznam o nehnuteľnosti z evidencie.

Záznam je samozrejme vyradený aj z quad stromu.

• **Zmena** – na základe **identifikačné čísla nehnuteľnosti** umožní meniť jednotlivé údaje

o nehnuteľnosti (okrem identifikačného čísla). Pri zmene GPS sa automaticky aktualizujú aj

zoznamy prepojených parciel.Algoritmy a údajové štruktúry 2 2023/2024

S1, Strana 2

• **Zmena** – na základe **identifikačné čísla parcely** umožní meniť jednotlivé údaje o parcele

(okrem identifikačného čísla). Pri zmene GPS sa automaticky aktualizujú aj zoznamy

prepojených nehnuteľností.

Najfrekventovanejšou operáciou, ktorá sa vykonáva je **vyhľadávanie**, a preto je potrebné, aby

táto operácia pracovala s **najväčšou možnou rýchlosťou**. Po nájdení záznamu je potrebné zobraziť

všetky evidované informácie.

**Keďže použité zariadenie disponuje malou RAM pamäťou a nie je stále napájané, je**

**potrebné, aby čo najviac údajov bolo neustále uložených na pevnom disku, resp. pamäťovej**

**karte. Zabráňte zbytočnej redundancií dát na disku. Pre účely testovania implementujte aj**

**operáciu zobrazenia celého aktuálneho obsahu databázy v GUI aplikácie (vrátane prepojenia**

**blokov, preplňujúceho súboru, prípadne ďalších súborov) – obsah všetkých súborov aplikácie**

**bude možné v aplikácií sekvenčne vypísať tak, aby bolo vidieť, čo jednotlivé bloky obsahujú**

**(vrátane ich interných atribútov).** V semestrálnej práci je potrebné použiť **dynamické hešovanie**

**využívajúce preplňujúci súbor (nepoužívate priame hešovanie, využite Vami navrhnutú**

**funkciu)**. **Implementujte efektívny manažment prázdnych blokov v súboroch založené na ich**

**zreťazení.** V dokumentácii uveďte **presný počet** prístupov do súboru pri jednotlivých operáciách

(vrôznych situáciách). Nezabudnite na všeobecné požiadavky semestrálnych prác (napr. generátor

na naplnenie databázy...). Export do textových súborov nie je potrebný, avšak aplikáciu je možné

zavrieť a neskôr pokračovať v práci s uloženými dátami (quad strom sa musí tiež naplniť).

Dôležitou časťou je dôsledné oddelenie jadra aplikácie od GUI.

**Pre zisk 3 bodov za prvú priebežnú kontrolu je potrebné najneskôr v desiatom týždni**

**predviesť:**

• Naprogramované a otestované (pomocou generátora operácii) dynamické hešovanie,

teda musia byť funkčné operácie vlož (2 body), vyhľadaj (1 bod) na ľubovoľných dátach

(bodový zisk závisí aj na kvalite implementácie). Riešenie kolízie s využitím

preplňujúceho súboru ešte **nemusí** byť implementované.

**Pre zisk 3 bodov za prvú priebežnú kontrolu je potrebné najneskôr v jedenástom týždni**

**predviesť:**

• Naprogramované a otestované (pomocou generátora operácii) dynamické hešovanie,

teda musia byť funkčné operácie vlož (1 bod), vymaž (1 bod) a vyhľadaj (1 bod) na

ľubovoľných dátach (bodový zisk závisí aj na kvalite implementácie). Riešenie kolízie

s využitím preplňujúceho súboru **musí** byť už implementované.

**Pracujte každý samostatne!**

Ak študent nepreukáže funkčnosť použitých údajových štruktúr pri kontrole rozpracovania,

preukáže ju pri odovzdávaní semestrálnej práce.

# Návrh štruktúry

A diagram of a computer

Description automatically generated

# Operácie údajovej štruktúry

## Vyhľadaj dáta

1. 1 prístup pre načítanie hlavného bloku dát
2. Ak sa dáta nenašli v hlavnom bloku:
   1. Opakuj pokiaľ nenájdeš dáta v preplňujúcom súbore, alebo neprejdeš všetky preplňujúce súbory hlavného bloku:
      1. 1 prístup pre načítanie preplňujúceho bloku

## Uprav dáta

1. 1 prístup pre načítanie hlavného bloku dát
2. Počet prístupov podľa miesta nájdenia dát:
   1. Ak sa dáta našli v hlavnom bloku:
      1. 1 prístup pre zapísanie upraveného hlavného bloku
   2. Ak sa dáta nenašli v hlavnom bloku:
      1. Opakuj pokiaľ nenájdeš dáta v preplňujúcom súbore, alebo neprejdeš všetky preplňujúce súbory hlavného bloku:
         1. 1 prístup pre načítanie preplňujúceho bloku
         2. 1 prístup pre zapísanie upraveného preplňujúceho bloku

## Vlož dáta

1. vložené dáta sa vložia do hlavného bloku, ktorý existoval a v ktorom je miesto
   1. počet prístupov z metódy vyhľadaj
   2. 1 prístup do súboru pre načítanie bloku
2. vložené dáta sa vložia do hlavného bloku, ktorý neexistoval a v ktorom je miesto
   1. počet prístupov z metódy vyhľadaj
   2. 1 prístup pre vytvorenie bloku
   3. 1 prístup pre načítanie blok
3. Vložené dáta sa vložia do hlavného bloku, ktorý musel byť rozšírený na 2 synov bez využitia preplňujúcich súborov
   1. počet prístupov z metódy vyhľadaj
   2. 1 prístup pre načítanie bloku
   3. Počet prístupov podľa situácie pri vymazávaní rozšíreného bloku:
      1. Ak je blok na konci súboru:
         1. 1 prístup pre skrátenie súboru na novú dĺžku
      2. Ak blok nie je na konci súboru:
         1. 1 prístup pre zapísanie „vyprázdneného“ bloku
         2. Ak nie je zreťazenie voľných hlavných blokov prázdne:
            1. 1 prístup pre načítanie prvého zreťazeného voľného hlavného bloku
            2. 1 prístup pre zapísanie prvého zreťazeného voľného hlavného bloku
            3. 1 prístup pre zapísanie nového voľného hlavného bloku
   4. Počet prístupov podľa výsledku rozšírenia bloku:
      1. Ak nie je žiaden z nových blokov po rozšírení plný:
         1. 1 zápis pre každý neprázdny nový blok
      2. Ak je aspoň jeden z nových blokov po rozšírení plný:
         1. 1 zápis pre každý neprázdny rozšírený hlavný blok
         2. Opakuj od bodu c pokiaľ nenastane situácia d. i.
4. Vložené dáta sa vložia do hlavného bloku, v ktorom nebolo miesto a museli byť využité preplňujúce bloky
   1. 1 prístup pre načítanie hlavného bloku
   2. Počet prístupov podľa toho, či má hlavný blok už nejaký preplňujúci blok:
      1. Ak hlavný blok ešte nemá preplňujúci blok:
         1. 1 prístup pre vytvorenie nového preplňujúceho bloku
      2. Ak hlavný blok má preplňujúci blok:
         1. Pokiaľ nenájdeš voľný preplňujúci blok hlavného bloku:
            1. 1 prístup pre načítanie každého nového preplňujúceho bloku
         2. Ak sa našiel voľný blok:
            1. Pokračuj na bod 4
         3. Ak sa nenašiel voľný blok:
            1. 1 prístup pre vytvorenie nového preplňujúceho bloku
            2. 1 prístup pre zapísanie posledného preplňujúceho bloku
         4. 2 prístupy pre zapísanie hlavného a preplňujúceho bloku

## Vymaž dáta

1. Proces striasania
   1. Opakuj pokiaľ sa nenačítajú všetky preplňujúce bloky hlavného bloku:
      1. 1 prístup pre načítanie preplňujúceho bloku
   2. Pre každý prázdny preplňujúci blok hlavného bloku po znovu vložení dát:
      1. Počet prístupov podľa situácie pri vymazávaní preplňujúceho prázdneho bloku:
         1. Ak je blok na konci súboru:
            1. 1 prístup pre skrátenie súboru na novú dĺžku
         2. Ak blok nie je na konci súboru:
            1. 1 prístup pre zapísanie „vyprázdneného“ bloku
            2. Ak nie je zreťazenie voľných hlavných blokov prázdne:

1 prístup pre načítanie prvého zreťazeného voľného hlavného bloku

1 prístup pre zapísanie prvého zreťazeného voľného hlavného bloku

1 prístup pre zapísanie nového voľného hlavného bloku

* 1. 1 prístup pre zapísanie každého neprázdneho preplňujúceho bloku
  2. 1 prístup pre zapísanie hlavného bloku

1. Dáta, ktoré mali byť vymazané boli nájdené v hlavnom bloku, ktorý neostal prázdny a nebolo potrebné spustiť proces striasania
   1. 1 prístup pre načítanie hlavného bloku
   2. 1 prístup pre zapísanie hlavného bloku
2. Dáta, ktoré mali byť vymazané boli nájdené v hlavnom bloku, ktorý ostal prázdny a na konci súboru a nebolo potrebné spustiť proces striasania
3. 1 prístup pre načítanie hlavného bloku
4. Počet prístupov podľa situácie pri vymazávaní hlavného bloku:
   * 1. Ak je blok na konci súboru:
        1. 1 prístup pre skrátenie súboru na novú dĺžku
     2. Ak blok nie je na konci súboru:
        1. 1 prístup pre zapísanie „vyprázdneného“ bloku
        2. Ak nie je zreťazenie voľných hlavných blokov prázdne:
           1. 1 prístup pre načítanie prvého zreťazeného voľného hlavného bloku
           2. 1 prístup pre zapísanie prvého zreťazeného voľného hlavného bloku
           3. 1 prístup pre zapísanie nového voľného hlavného bloku
5. Dáta, ktoré mali byť vymazané boli nájdené v hlavnom bloku, ktorý neostal prázdny a bolo potrebné spustiť proces striasania:
6. Rovnako ako v bode 2.
7. Rovnako ako proces striasania (bod 1)
8. Dáta, ktoré mali byť vymazané boli nájdené v hlavnom bloku, ktorý ostal prázdny a na konci súboru a bolo potrebné spustiť proces striasania:
9. Rovnako ako v bode 3.
10. Rovnako ako proces striasania (bod 1)
11. Dáta, ktoré mali byť vymazané neboli nájdené v hlavnom bloku, ale v preplňujúcom bloku hlavného bloku a preplňujúci blok po vymazaní neostal prázdny a na konci súboru a nebolo potrebné spustiť proces striasania
    1. 1 prístup pre načítanie hlavného súboru
    2. Pokiaľ sa nenájde preplňujúci blok opakuj:
       1. 1 prístup pre načítanie preplňujúceho bloku
    3. 1 prístup pre zapísanie preplňujúceho bloku po vymazaní dát
12. Dáta, ktoré mali byť vymazané neboli nájdené v hlavnom bloku, ale v preplňujúcom bloku hlavného bloku a preplňujúci blok po vymazaní ostal prázdny a na konci súboru a nebolo potrebné spustiť proces striasania:
    1. 1 prístup pre načítanie hlavného súboru
    2. Pokiaľ sa nenájde preplňujúci blok opakuj:
       1. 1 prístup pre načítanie preplňujúceho bloku
    3. Počet prístupov podľa stavu predchádzajúceho a nasledujúceho zreťazeného preplňujúceho bloku aktuálneho preplňujúceho bloku:
       1. Ak preplňujúci blok nemá žiaden predchádzajúci ani nasledujúci preplňujúci blok:
          1. 1 prístup pre zapísanie hlavného bloku – bola mu vymazaná adresa jeho preplňujúceho bloku
       2. Ak preplňujúci blok nemá žiaden predchádzajúci blok, ale má nasledujúci blok:
          1. 1 prístup pre zapísanie hlavného bloku
          2. 1 prístup pre načítanie nasledujúceho bloku
          3. 1 prístup pre zapísanie nasledujúceho bloku
       3. Ak preplňujúci blok má predchádzajúci blok, ale nemá nasledujúci:
          1. 1 prístup pre načítanie predchádzajúceho bloku
          2. 1 prístup pre zápis predchádzajúceho bloku
       4. Ak preplňujúci blok má aj predchádzajúci aj nasledujúci preplňujúci blok:
          1. 1 prístup pre načítanie predchádzajúceho bloku
          2. 1 prístup pre zápis predchádzajúceho bloku
          3. 1 prístup pre načítanie nasledujúceho bloku
          4. 1 prístup pre zápis nasledujúceho bloku
    4. Počet prístupov podľa situácie pri vymazávaní preplňujúceho bloku:
       1. Ak je blok na konci súboru:
          1. 1 prístup pre skrátenie súboru na novú dĺžku
       2. Ak blok nie je na konci súboru:
          1. 1 prístup pre zapísanie „vyprázdneného“ bloku
          2. Ak nie je zreťazenie voľných preplňujúcich blokov prázdne:
             1. 1 prístup pre načítanie prvého zreťazeného voľného preplňujúceho bloku
             2. 1 prístup pre zapísanie prvého zreťazeného voľného preplňujúceho bloku
             3. 1 prístup pre zapísanie nového voľného preplňujúceho bloku
13. Dáta, ktoré mali byť vymazané neboli nájdené v hlavnom bloku, ale v preplňujúcom bloku hlavného bloku a preplňujúci blok po vymazaní ostal prázdny a na konci súboru a bolo potrebné spustiť proces striasania:
    1. Rovnako ako bod 7
    2. Rovnako ako proces striasania (bod 1)