

**Fakulta riadenia a informatiky - Žilinská univerzita**

## **Semestrálna práca 2**

Algoritmy a údajové štruktúry 1

# 1. Zadanie

Vytvorenie aplikácie na podporu rozhodovania určeného pre biofarmára. Aplikácia slúži na hromadné spracovanie a analýzu štatistík z poľnohospodárstva. Navyše obsahuje modul, ktorý umožní biofarmárovi správne sa rozhodnúť pri budúcich investíciách.

## **Aplikácia musí podporovať nasledujúcu funkčnosť:**

1. Vyhľadanie obce na základe jej názvu. Pre každý rok z rozpätia rokov, ktoré bude možné zadať, vypíšte nasledujúce údaje o obci:
  - Celková výmera obce.
  - Výmera ornej pôdy.
  - Výmera ovocných sádov.
  - Podiel výmery vodných plôch voči celkovej výmere obce.
  - Podiel výmery lesných pozemkov voči celkovej výmere obce.
  - Podiel výmery zastavaných plôch a nádvorí voči celkovej výmere obce.
2. Zoradenie všetkých obcí na základe:
  - Celkovej výmery obce v danom roku (rok bude možné zadať).
  - Prírastku zastavaných plôch voči celkovej výmere obce v danom roku (rok bude možné zadať).
  - Zmeny podielu výmery ornej pôdy voči celkovej výmere obce medzi dvomi rokmi (roky bude možné zadať).

Obce umožnite zoradiť vzostupne aj zostupne. Pri zoradení obcí vždy vypíšte jej názov a príslušnú hodnotu triediaceho kritéria a-c.

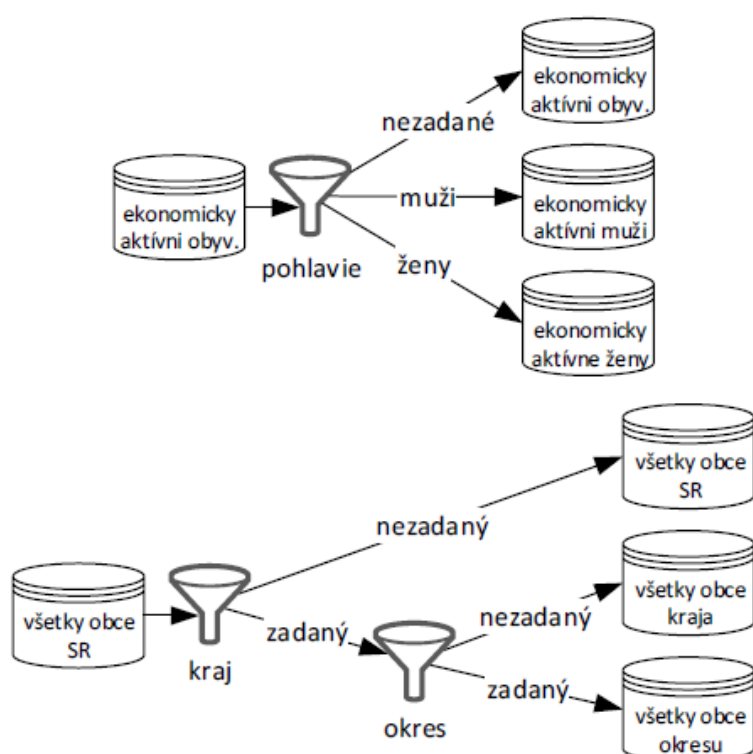
3. Vyhľadanie okresu, ktorý mal v danom roku (rok bude možné zadať):
  - Najväčšiu a najmenšiu celkovú výmeru ovocných sádov.
  - Najväčší a najmenší podiel celkovej výmery ornej pôdy okresu voči celkovej výmere ovocných sádov okresu.
  - Najväčší a najmenší podiel výmery lesov voči celkovej výmere okresu.

Uveďte názov okresu a hodnotu kritéria, na základe ktorej bol okres vyhľadaný.

4. Zoradenie všetkých okresov na základe:
  - Zmeny podielu celkovej výmery ovocných sádov okresu voči celkovej výmere okresu medzi dvomi rokmi (roky bude možné zadať).
  - Zmeny podielu celkovej výmery ornej pôdy okresu voči celkovej výmere okresu medzi dvomi rokmi (roky bude možné zadať).
  - Zmeny podielu celkovej výmery lesov okresu voči celkovej výmere okresu medzi dvomi rokmi (roky bude možné zadať).

Okresy umožnite zoradiť vzostupne aj zostupne. Pri zoradení okresov vždy vypíšte názov okresu a hodnotu triediaceho kritéria a-c.

5. Vyhľadanie krajov, pre ktoré v danom roku (rok je možné zadať) platí:
  - Obsahuje najviac a najmenej obcí, ktorých celková výmera lesa voči celkovej výmere obce je aspoň X% (X je možné zadať). Uveďte tieto obce v kraji spĺňajúcom kritérium a hodnotu podielu príslušných výmer v obciach.
  - Obsahuje aspoň X okresov, ktorých celková výmera lesa voči celkovej výmere okresu je aspoň Y% (X a Y je možné zadať). Pre každý kraj (môže ich byť viac) spĺňajúci toto kritérium uveďte tieto okresy a hodnotu podielu príslušných výmer.
  - Podiel celkovej výmery lesa kraja z celkovej výmery kraja je aspoň X% (X je možné zadať). Pre každý kraj (môže ich byť viac) spĺňajúci toto kritérium uveďte hodnotu príslušnej výmery.
6. Zoradenie obcí na základe počtu ekonomicky aktívneho obyvateľstva (obyvateľstvo vo veku 18 – 65 rokov). Umožnite zadať nasledujúce filtre:



7. Vytvorte textový súbor s dátami v nasledujúcom formáte:

ID obce	ID okresu	ID kraja	Počet ekonomicky aktívnych obyvateľov	Názov obce	Výmera ornej pôdy	Výmera ovocných sádov
---------	-----------	----------	---------------------------------------	------------	-------------------	-----------------------

Dáta v súbore budú oddelené znakom ; (bodkočiarka). Ekonomicky aktívne obyvateľstvo je počet obyvateľov vo veku 18-65 rokov. Pre vytvorenie súboru je potrebné zadať:

- Názov súboru.
- Rok, z ktorého sa budú exportovať údaje (rok má zmysel pre údaje o ornej pôde a ovocných sadoch, počet ekonomicky aktívnych obyvateľov použite všade rovnaký).
- Obce, ktoré sa majú do súboru zapísať (v zmysle filtra obcí definovaného v úlohe 6)

8. Biofarmár chce rozšíriť svoju výrobu tak, aby mohol pestovať zemiaky na nových plochách ornej pôdy a jablone na nových plochách ovocných sádov. Potrebuje nakúpiť, resp. prenajať si pôdu vo vybraných obciach zadaného okresu tak, aby počet ekonomicky aktívnych obyvateľov (obyvatelia vo veku 18-65 rokov) vo vybraných obciach zadaného okresu bol **čo najvyšší**, pričom zemiaky bude pestovať na najviac 20 % celkovej výmery ornej pôdy v zadanom okrese a jablone na najviac 10 % celkovej výmery ovocných sádov v zadanom okrese. Predpokladá sa, že farmár si bude môcť kúpiť/prenajať celú výmeru ornej pôdy aj celú výmeru ovocných sádov vo vybraných obciach určených riešením.

Riešenie pomocou duálnej vsúvacej heuristiky so zadaným lokálnym kritériom: Odstráň z riešenia obec z dosiaľ nespracovaných obcí, ktorá **má najmenší počet obyvateľov**.

## 2. Návrh aplikácie

### 2.1. Rozbor použitých údajových štruktúr

Do aplikácie som sa rozhodol implementovať nasledujúce štruktúry:

- **Array** - pole
- **ArrayList** - zoznam implementovaný poľom
- **LinkedList** - jednostranne zreťazený zoznam
- **Heap** – prioritný front implementovaný haldou
- **ExplicitStack** – explicitný zásobník
- **Treap** – vyvážený binárny vyhľadávací strom
- **UnsortedSequenceTableNoUnique** – neutriedená sekvenčná tabuľka, ktorá pri vkladaní nekontroluje unikátnosť kľúčov, takže má rýchle vkladanie

Využitia jednotlivých štruktúr:

**Array** - pre rýchly prístup k údajom napr. pre uchovávanie výmer obcí v jednotlivých rokoch som použil pole polí.

**ArrayList** – pre uchovávanie obcí v tabuľke v ktorej sú kľúčmi názvy obcí. V prípade rovnakých názvov sú tieto obce v ArrayListe

**LinkedList** – zoznamy cez ktoré sa iba iteruje a nie je potrebný rýchly prístup k jednotlivým prvkom.

**Heap** – pre odstraňovanie obcí s najmenším počtom obyvateľov z množiny obcí v algoritme heuristiky

**ExplicitStack** – pre vypísanie prvkov v opačnom poradí než ako sú usporiadané

**Treap** – pre rýchly prístup k prvkom (obciam, okresom, krajom) na základe kľúča (id, názov)

**UnsortedSequenceTableNoUnique** – pre triedenie prvkov na základe nejakého pravidla pomocou komparátorov

### 2.2. Zložitosť jednotlivých operácií

#### 2.2.1. Načítanie vstupných dát

Pre jednoduchšie spracovanie som si vstupné dáta upravil odstránením diakritiky a odstránením nepotrebných údajov. Taktiež som presunul každý údaj na nový riadok pre jednoduchšie načítavanie. Štruktúry vstupných súborov sú popísané v súbore vstupne\_subory\_struktura.txt.

**Načítanie krajov:** zo súboru kraje\_id sa načítajú všetky kraje (id kraja, názov kraja) a vložia sa do tabuľky (Treap) s kľúčmi id kraja a tabuľky (Treap) s kľúčmi názov kraja.

**Načítanie okresov:** zo súboru okresy\_id sa načítajú všetky okresy (id okresu, id kraja, názov okresu) a vložia sa do tabuľky (Treap) s kľúčmi id okresu a tabuľky (Treap) s kľúčmi názov okresu.

**Načítanie obcí:** zo súboru obce\_okresy\_id sa získa id obce, názov obce a okres do ktorého patrí, vytvorí sa objekt Obec a vloží sa do tabuľky (Treap) s kľúčmi id obce a do tabuľky (Treap) s kľúčmi názov obce. Zo súboru vymera\_raw sa načítajú údaje o výmerách obcí. Najprv sa obec s daným id musí nájsť ( $n \cdot \log(n)$ ) a potom sa načítajú jej údaje. Zo súboru obyvateľstvo\_raw sa načítajú údaje o obyvateľstve obcí, okresov a krajov. Najprv sa obec/okres/kraj s daným id musí nájsť ( $n \cdot \log(n)$ ) a potom sa načítajú jej údaje.

$n$  – počet krajov,  $o$  – počet okresov,  $p$  – počet obcí

**Výsledná zložitosť:**  $O(n \cdot (2 \cdot n \cdot \log(n)) + o \cdot (2 \cdot o \cdot \log(o)) + p \cdot (2 \cdot p \cdot \log(p)) + p \cdot (p \cdot \log(p)) + (n+o+p) \cdot ((n+o+p) \cdot \log(n+o+p)))$

### 2.2.2. Zložitosť jednotlivých funkčností zo zadania

1. Najprv sa musí vyhľadať okres podľa názvu ( $\log(n)$ ) kvôli duplicitným názvom obcí a ďalej sa musí vyhľadať obec podľa názvu a okresu ( $\log(o)$ ). Získaný objekt Obec má všetky údaje o výmerách uložené v poli takže prístup k nim má zložitosť  $O(1)$

$n$  – počet okresov,  $o$  – počet obcí

**Výsledná zložitosť:**  $O(\log(n) + \log(o))$

2. Obce v neutriedenej tabuľke sa zoradia algoritmom Quicksort ( $n \cdot \log(n)$ ) od najmenšej po najväčšiu. Na porovnávanie podľa jednotlivých kritérií (a - c) sa používajú potomkovia triedy Comparator, ktoré sa pošlú triediacemu algoritmu. V prípade že bol zvolený výpis vzostupne, tak sa vypíšu. Ak bol zvolený výpis zostupne, tak sa povkladajú do zásobníka a vypíšu sa v opačnom poradí.

**Výsledná zložitosť:**  $O(n \cdot \log(n) + n)$

3. V prípade, že sa hľadajú okresy s najmenšou a najväčšou hodnotou prvý krát, tak sa prehľadajú všetky okresy a nájde sa minimum a maximum ( $O(n)$ ). Tieto hodnoty sa uložia do poľa pre daný rok. V prípade, že sa neskôr hľadá rovnaký rok, tak už sa ako výsledok vráti táto uložená hodnota ( $O(1)$ ).

**Výsledná zložitosť:**  $O(n)$

4. Okresy v neutriedenej tabuľke sa zoradia algoritmom Quicksort ( $n \cdot \log(n)$ ) od najmenšieho po najväčší. Na porovnávanie podľa jednotlivých kritérií (a - c) sa používajú potomkovia triedy Comparator, ktoré sa pošlú triediacemu algoritmu. V prípade že bol zvolený výpis vzostupne, tak sa vypíšu. Ak bol zvolený výpis zostupne, tak sa povkladajú do zásobníka a vypíšu sa v opačnom poradí.

**Výsledná zložitosť:**  $O(n \cdot \log(n) + n)$

5. a) Pre každý kraj prejde všetky okresy a v nich všetky obce a vytvorí neutriedenú tabuľku v ktorej sú kľúče kraj a dáta zoznam obcí z daného kraja, ktoré spĺňajú podmienku. Pomocou algoritmu Quicksort túto tabuľku zoradí a vypíše kraj na prvom a poslednom mieste spolu s obcami, ktoré spĺňajú podmienku.

$n$  – počet krajov,  $o$  – počet okresov v kraji,  $p$  – počet obcí v okrese

**Výsledná zložitosť:**  $O((n \cdot o \cdot p) + (n \cdot \log(n)) + n \cdot (o \cdot p))$

b) Pre každý kraj zistí koľko obsahuje okresov s danou výmerou lesa a ak je ich počet väčší ako zadaná hodnota tak ho vypíše spolu s okresmi spĺňajúcimi danú podmienku.

$n$  – počet krajov,  $o$  – počet okresov v kraji

**Výsledná zložitosť:**  $O((n*o) + (n*o))$

c) Pre každý kraj zistí či spĺňa danú podmienku a ak áno, tak ho vypíše.

$n$  – počet krajov

**Výsledná zložitosť:**  $O(n)$

6. Nájdenie kraja / okresu podľa zadaného názvu z tabuľky ( $O(\log(n))$ ). Nájdenie obcí, ktoré patria do daného kraja / okresu ( $O(o)$ ). Utriedenie obcí podľa počtu obyvateľov algoritmom Quicksort ( $o*\log(o)$ ) od najmenšieho po najväčší. Na porovnávanie sa používajú potomkovia triedy Comparator, ktoré sa pošlú triediacemu algoritmu. V prípade že bol zvolený výpis vzostupne, tak sa vypíšu. Ak bol zvolený výpis zostupne, tak sa povkladajú do zásobníka a vypíšu sa v opačnom poradí.

$n$  – počet krajov / okresov,  $o$  – počet obcí v kraji / okrese

**Výsledná zložitosť:**  $O((\log(n)) + o + (o*\log(o)) + 2*o)$

7. Pomocou funkcionality z bodu 6 sa vyberú požadované obce a zapíšu sa do zvoleného súboru s požadovanými údajmi

$n$  – počet krajov / okresov,  $o$  – počet obcí v kraji / okrese

**Výsledná zložitosť:**  $O((\log(n)) + o + (o*\log(o)) + 2*o)$

8. Pomocou funkcionality z bodu 6 sa vyberú požadované obce, ktoré majú nenulové výmery ovocných sádov a ornej pôdy a vložia sa do prioritného frontu (Heap) s prioritou počet obyvateľov. Následne sa z tohto frontu odoberajú obce s najnižším počtom obyvateľov pokiaľ nie sú splnené štrukturálne podmienky z modelu úlohy.

$n$  – počet krajov / okresov,  $o$  – počet obcí v kraji / okrese

**Výsledná zložitosť:**  $O((\log(n)) + o + (o*\log(o)) + 2*o + 2*o*\log(o) + o)$

### 2.2.3. Vzorce použité pre výpočty

#### **Podiel výmery vodných plôch voči celkovej výmere obce**

(Výmera vodných plôch / celková výmera obce) \* 100

#### **Podiel výmery lesných pozemkov voči celkovej výmere obce**

(Výmera lesných pozemkov / celková výmera obce) \* 100

#### **Podiel výmery zastavaných plôch a nádvorí voči celkovej výmere obce**

(Výmera zastavaných plôch a nádvorí / celková výmera obce) \* 100

#### **Prírastok zastavaných plôch voči celkovej výmere obce**

$$(((\text{Výmera zastavaných plôch za zadaný rok} / \text{celková výmera obce za zadaný rok}) - (\text{Výmera zastavaných plôch za predch. rok} / \text{celková výmera obce za predch. rok})) / (\text{Výmera zastavaných plôch za predch. rok} / \text{celková výmera obce za predch. rok})) * 100$$

**Zmeny podielu výmery ornej pôdy voči celkovej výmere obce medzi dvomi rokmi**

(Podiel výmery v prvom roku - podiel výmery v druhom roku) \* 100

**Podiel celkovej výmery ornej pôdy okresu voči celkovej výmere ovocných sádov okresu**

(Výmera ornej pôdy okresu / výmera ovocných sádov okresu) \* 100

**Podiel výmery lesov voči celkovej výmere okresu**

(Výmera lesov okresu / celková výmera okresu) \* 100

**Zmena podielu celkovej výmery ovocných sádov okresu voči celkovej výmere okresu medzi dvomi rokmi**

(Podiel v prvom roku - podiel v druhom roku) \* 100

**Zmena podielu celkovej výmery ornej pôdy okresu voči celkovej výmere okresu medzi dvomi rokmi**

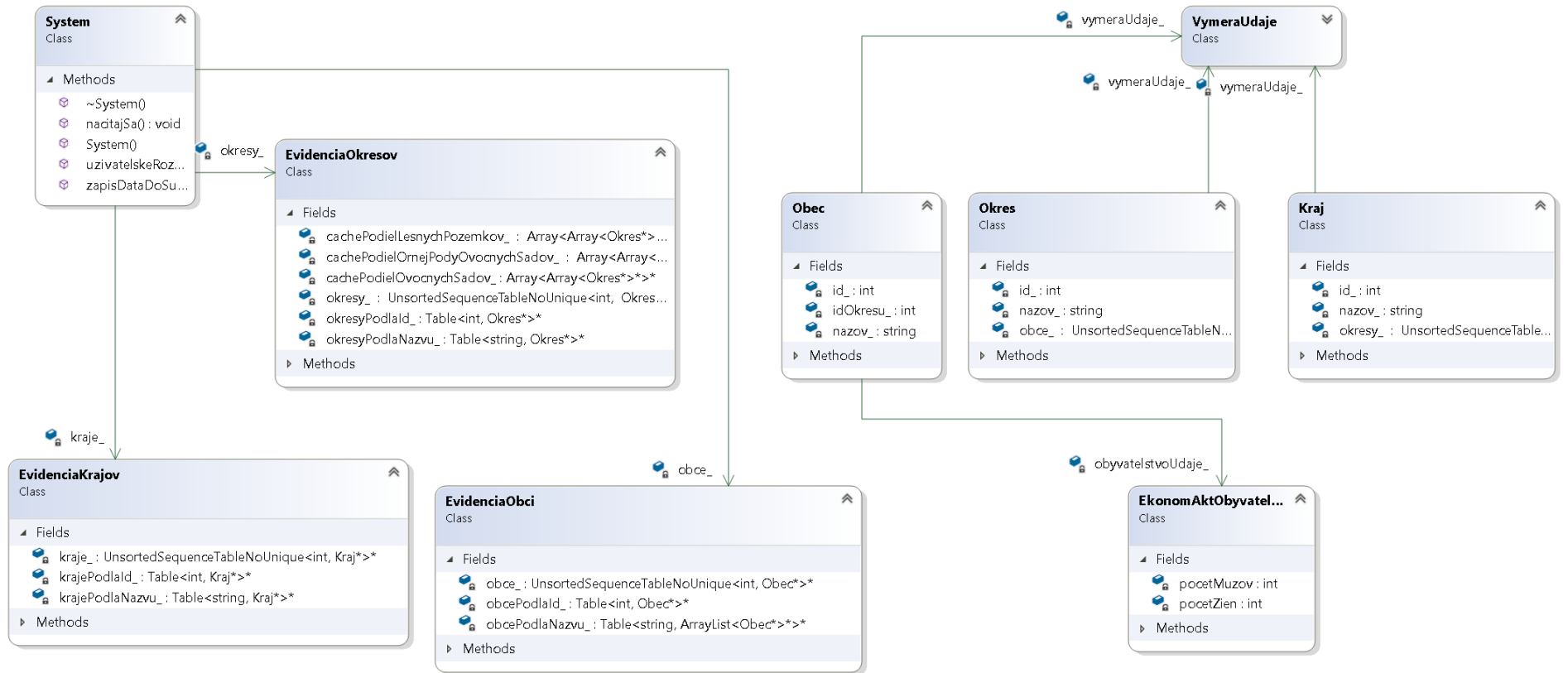
(Podiel v prvom roku - podiel v druhom roku) \* 100

**Zmena podielu celkovej výmery lesov okresu voči celkovej výmere okresu medzi dvomi rokmi**

(Podiel v prvom roku - podiel v druhom roku) \* 100



## 2.3. Diagram tried



### **3. Používateľská príručka**

Ovládanie je zabezpečené pomocou konzoly. Po spustení aplikácie a načítaní vstupných dát si môže užívateľ vybrať čo chce robiť z úvodného menu.

## Obsah

1. Zadanie .....	2
2. Návrh aplikácie.....	5
2.1. Rozbor použitých údajových štruktúr .....	5
2.2. Zložitosti jednotlivých operácií .....	5
2.2.1. Načítanie vstupných dát .....	5
2.2.2. Zložitosti jednotlivých funkčností zo zadania .....	6
2.2.3. Vzorce použité pre výpočty .....	7
2.3. Diagram tried.....	9
3. Používateľská príručka .....	10