# Datové struktury – semestrální práce B

Naším úkolem bylo :

Nad vybudovanou datovou strukturou graf ze semestrální práce A bylo za úkol vytvořit pomocnou strukturu Grid index, která bude uplatňovat přístup k uzlům sítě podle jejich sekundárních (multidimenzionálních) klíčů.

Struktura Grid index, která je aplikována na dvojdimenzionální data (klíče) sestává jednak ze dvou lineárních stupnic (řádkových vektorů) a jednoho Grid adresáře (dvourozměrného pole). Položky v Grid adresáři umožňují přímý přístup k jednotlivým uzlům sítě specifikovaným pomocí příslušných x,y-koordinátů.

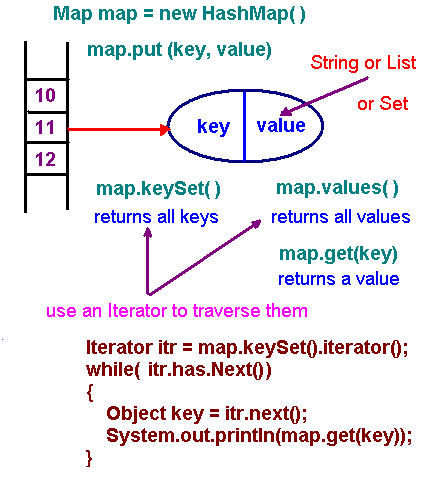
Při realizaci softwarové aplikace je požadováno, aby bylo umožněno:

• (bodové) vyhledávání uzlu podle hodnot koordinátů,

• intervalové vyhledávání uzlů v rámci zadaného obdélníkového segmentu.

## Datová struktura GridIndex

Základ semestrální práce je postaven na kolekci HashMap . Tuto kolekci jsem vybral z důvodu použití dvojdimenzionálních klíče který je složený z koordinátu X a Y. Do hashmapy jsem jako klíč ukládal prvek Point a jako hodnotu IMesto. Jelikož při metodě vkládání jsem potřeboval porovnávat dle koordinátu aby se dané města (prvky) mohli dobře vkládat. Také u metody vkládání zabezpečujeme, že na dané, kvadrantu bude pouze jeden prvek. Při vkládání nového města se rozhodujeme jestli se budou kvadranty řezat podle osy X nebo podle osy Y. Po rozhodnutí podle které osy se bude, spočítáme nový řez zeptáme se jestli na starém řezu na kterém se nacházíme není město s většími souřadnicemi které můžeme přesunout do nového řezu, když se město najde převedeme ho do nového kvadrantu když se nenajde necháme ho na stejném místě. Poté přidáme nový prvek kde se ptáme jestli prvek chce jít do nového řezu a nebo zůstat na řezu starém. Při zjištění že na daném řezu neleží žádné město vkládáme ho automaticky. Jelikož pracuji s koordináty zvolil jsem jako výchozí souřadnice [0,0].



|  |  |  |
| --- | --- | --- |
| **Hash table** | | |
| [**Type**](http://en.wikipedia.org/wiki/List_of_data_structures) | Unordered [associative array](http://en.wikipedia.org/wiki/Associative_array" \o "Associative array) | |
| **Invented** | 1953 | |
| [**Time complexity**](http://en.wikipedia.org/wiki/Time_complexity) **in**[**big O notation**](http://en.wikipedia.org/wiki/Big_O_notation) | | |
|  | Average | Worst case |
| **Space** | O(*n*)[[1]](http://en.wikipedia.org/wiki/Hash_table#cite_note-Cormen_et_al-1) | O(*n*) |
| **Search** | O(1) | O(*n*) |
| **Insert** | O(1) | O(*n*) |
| **Delete** | O(1) | O(*n*) |