

ACH2024 –

Algoritmos e Estruturas de Dados II

Prof. Helton Hideraldo Biscaro
heltonhb@usp.br

Dados espaciais

Quadtree para Pontos

Quadtree

- Espaço dividido em quadrantes (direções)



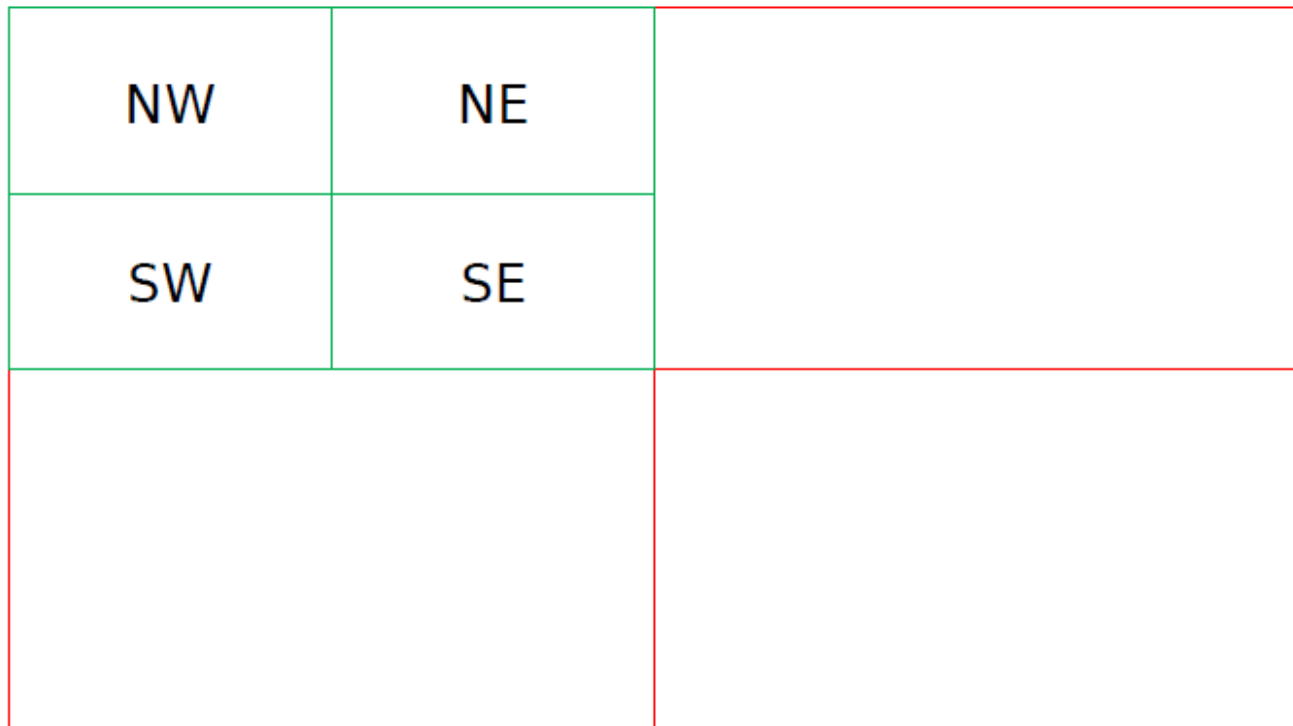
Quadtree

- Espaço dividido em quadrantes (direções)



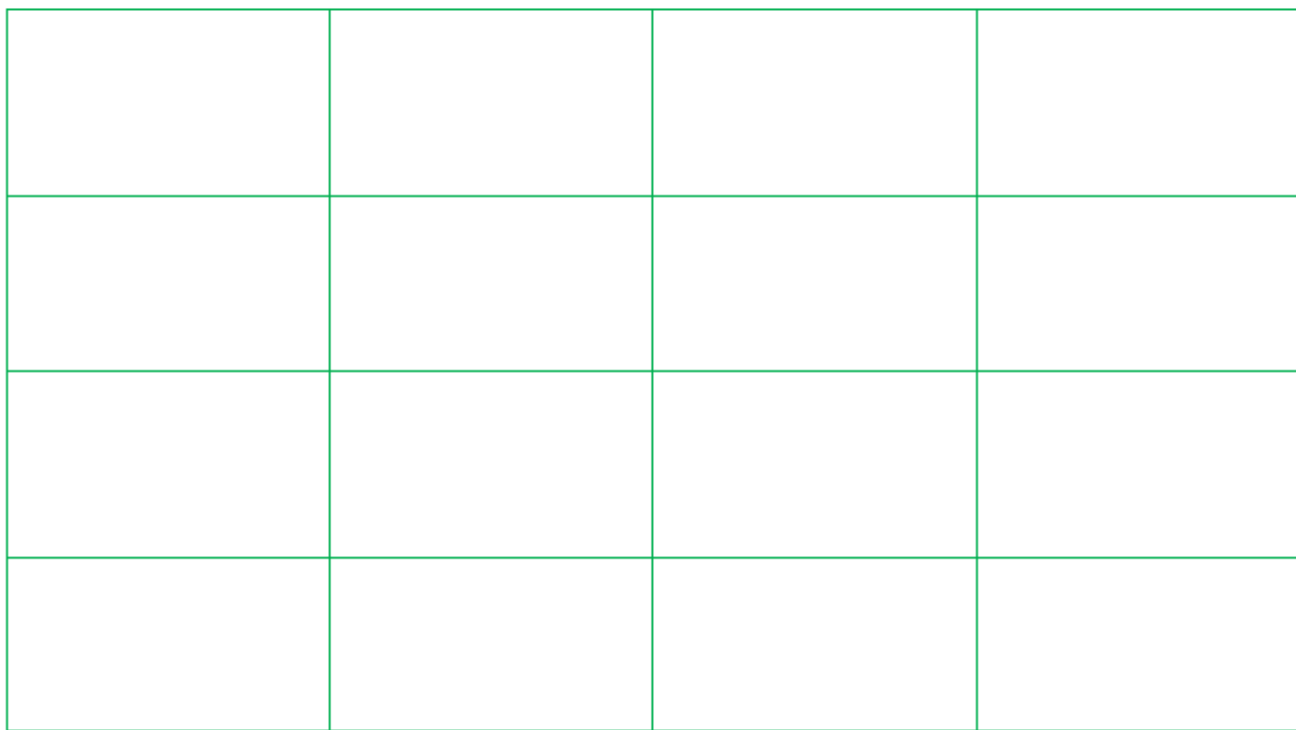
Quadtree

- Espaço dividido em quadrantes (direções)



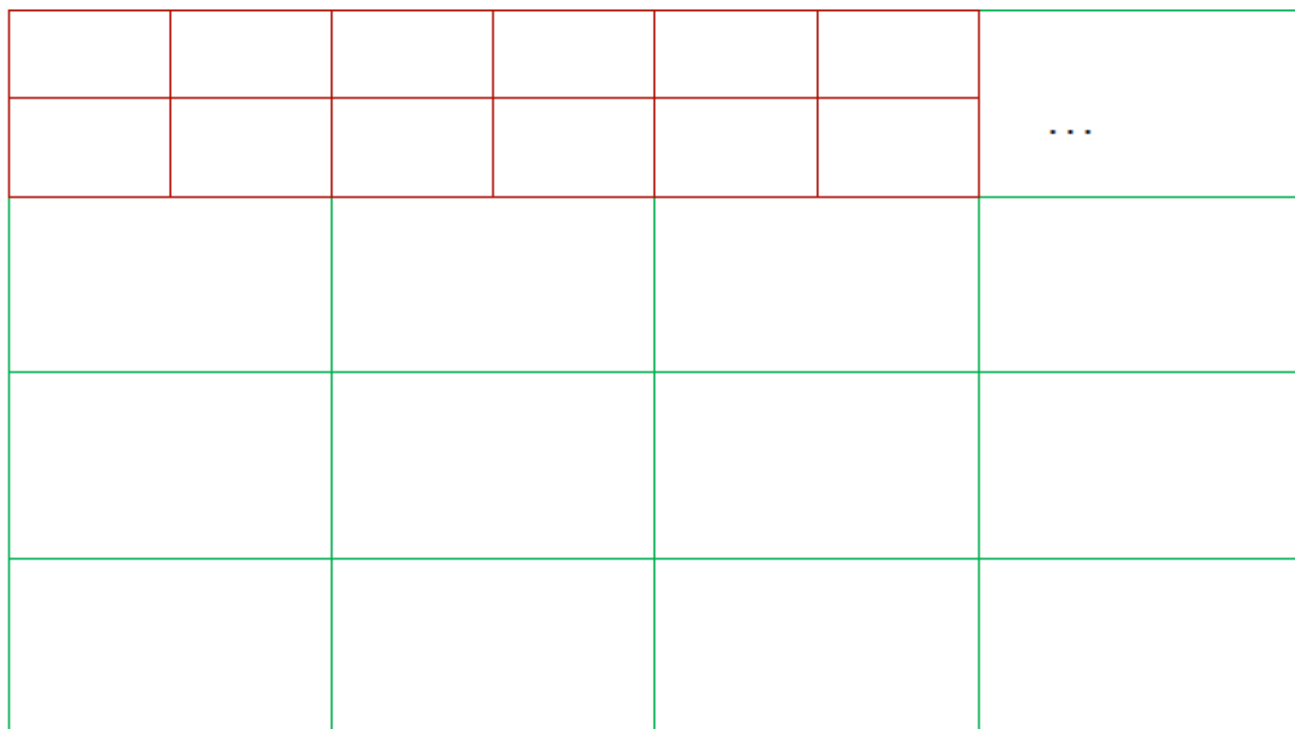
Quadtree

- Espaço dividido em quadrantes (direções)



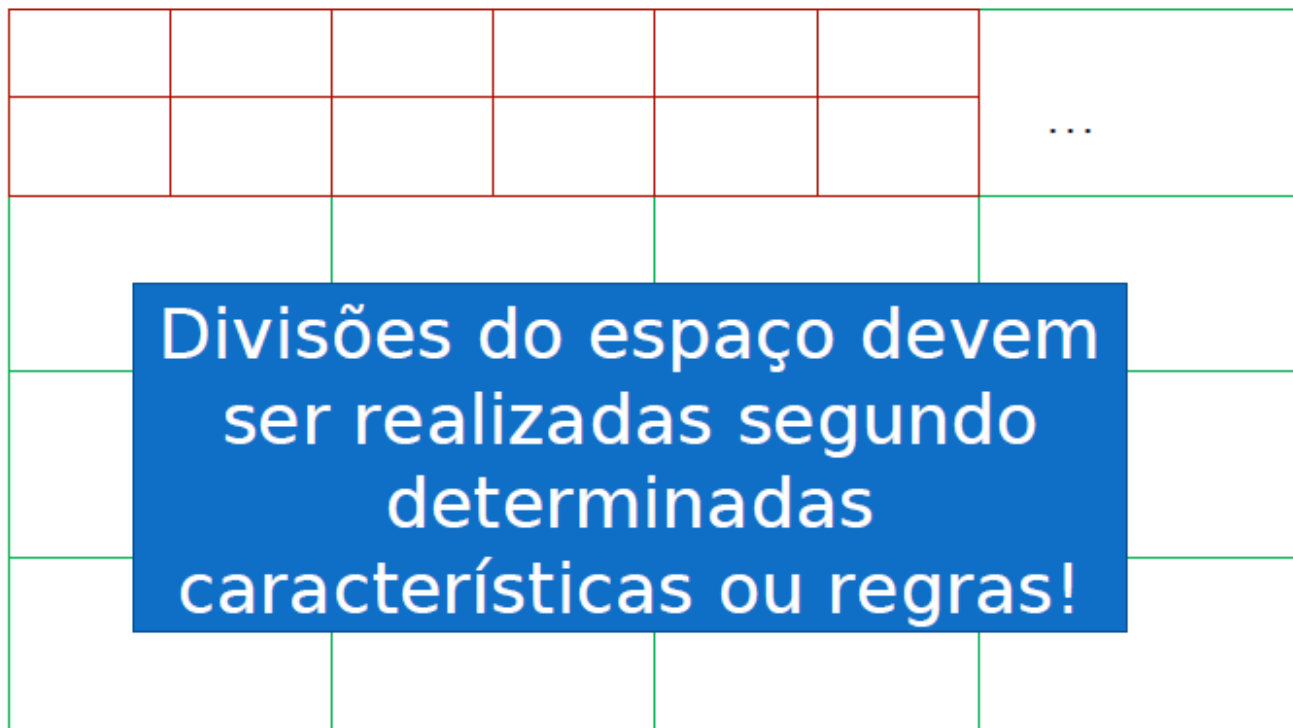
Quadtree

- Espaço dividido em quadrantes (direções)



Quadtree


- Espaço dividido em quadrantes (direções)



Point Quadtrees

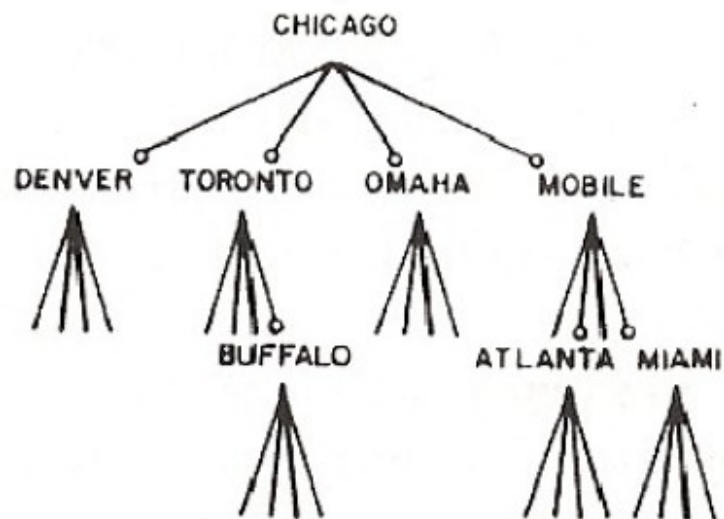
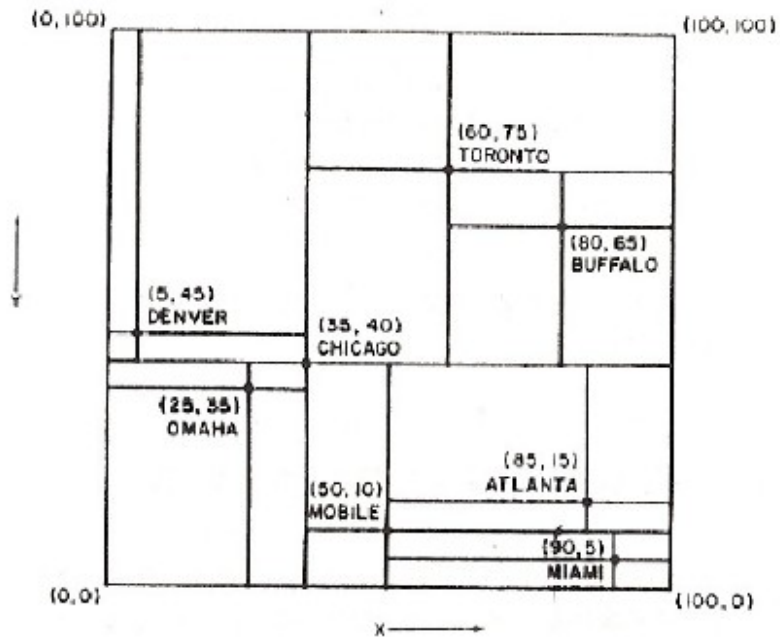
- Finkel e Benkley [1974]
- É implementada como uma generalização multidimensional de uma árvore binária de busca.
- Em duas dimensões cada ponto de dados é representado por um nó da quadtree: Tipo de dados com 7 campos:

- Primeiros 4 campos representam os 4 quadrantes (ou direções)
 - NW (noroeste)
 - NE (nordeste)
 - SW (sudoeste)
 - SE (sudeste)
- Coordenada X do ponto
- Coordenada Y do ponto
- Campo nome: contém informações descritivas sobre o nó, por exemplo, nome da cidade.

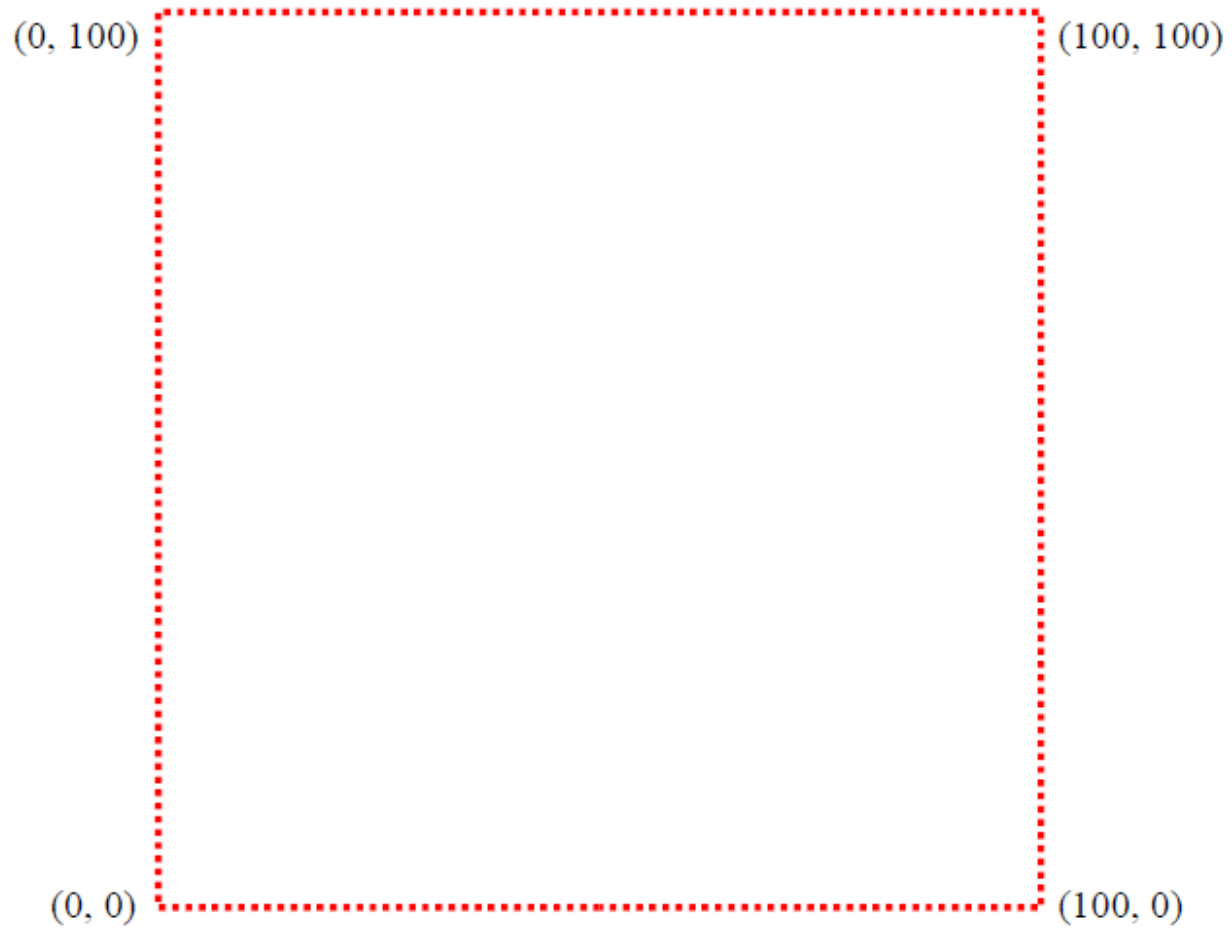


Se P é um ponteiro para um nó e I é um quadrante, então estes campos são referenciados SON(P, I)

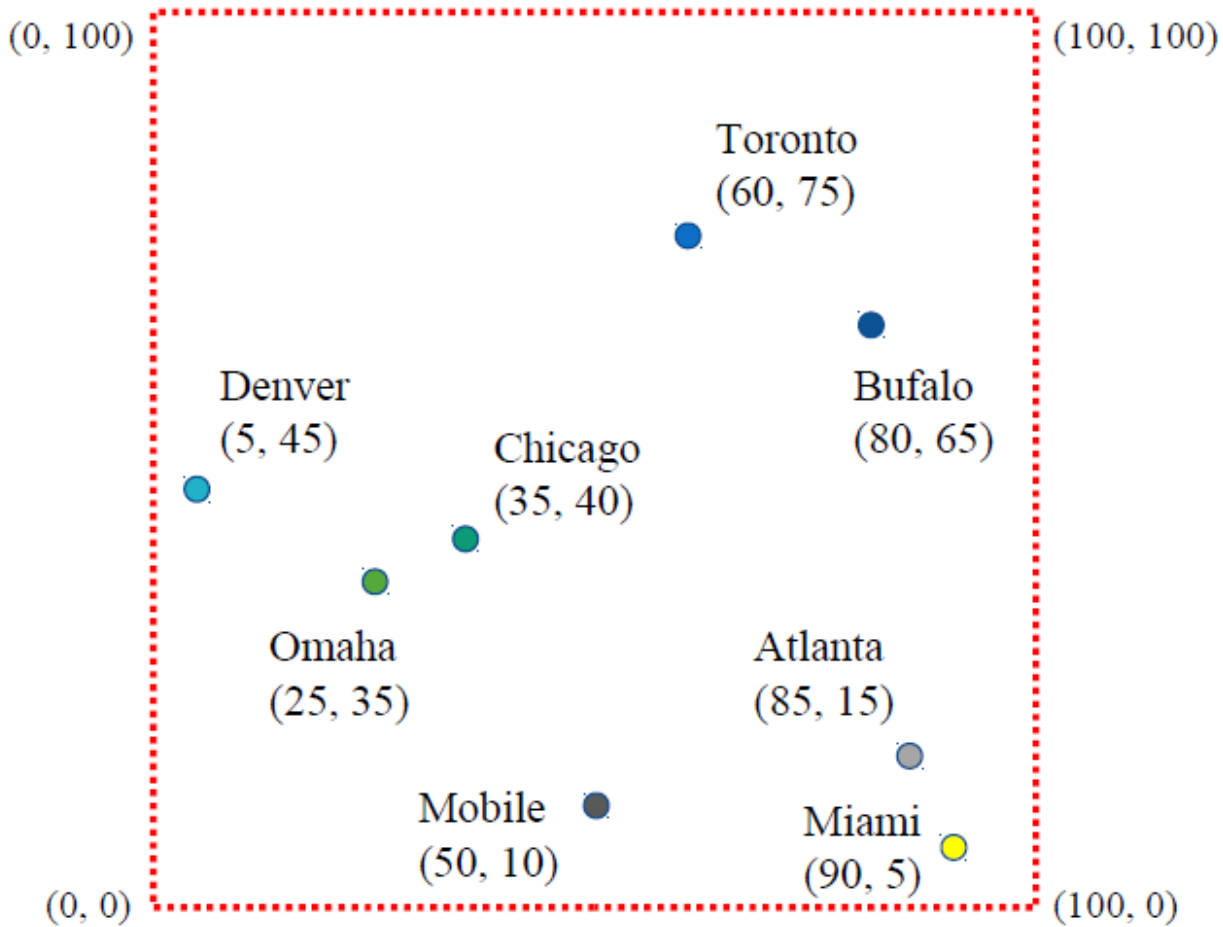
Point QuadTrees - Exemplo



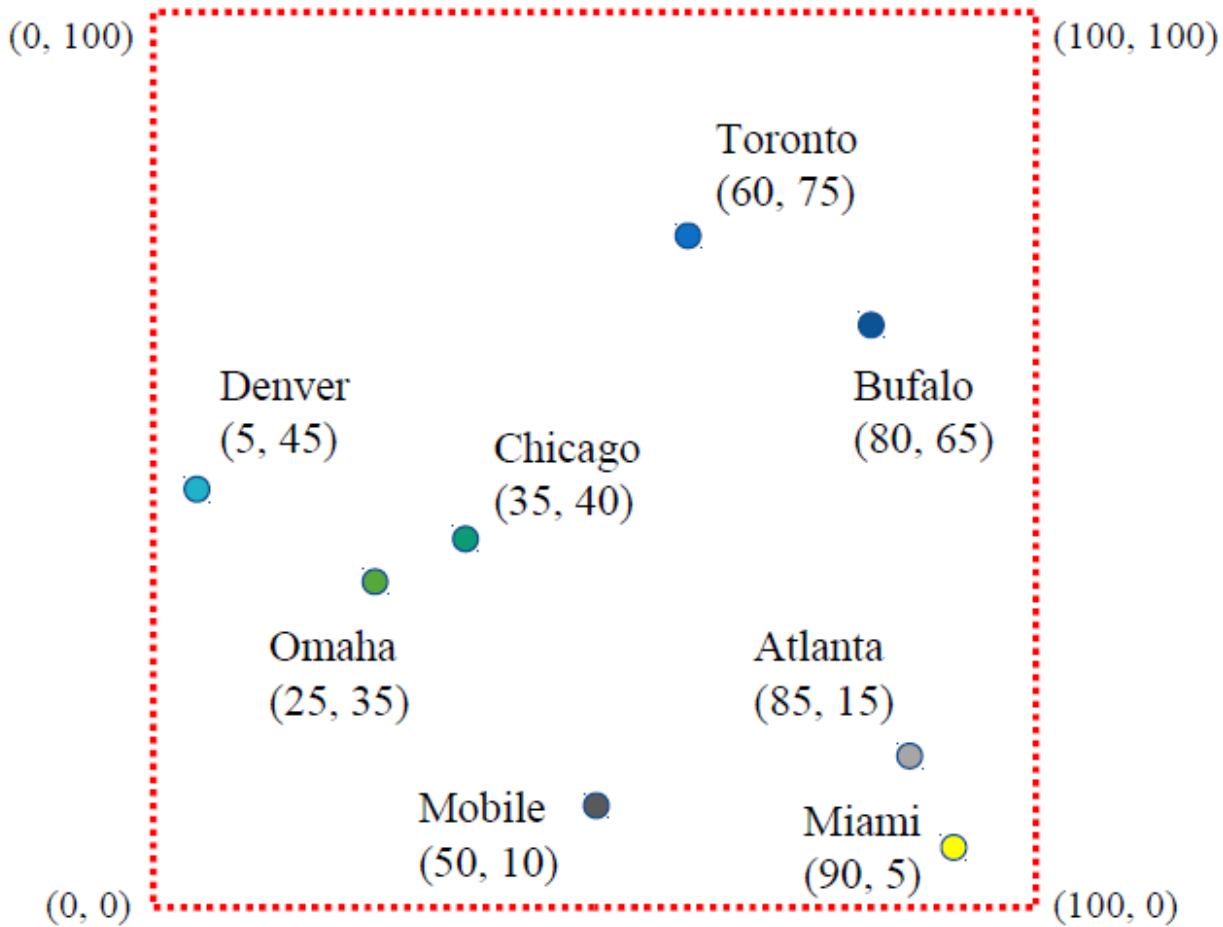
Point QuadTrees - Exemplo (Quadrante)



Point QuadTrees - Cidades (pontos)

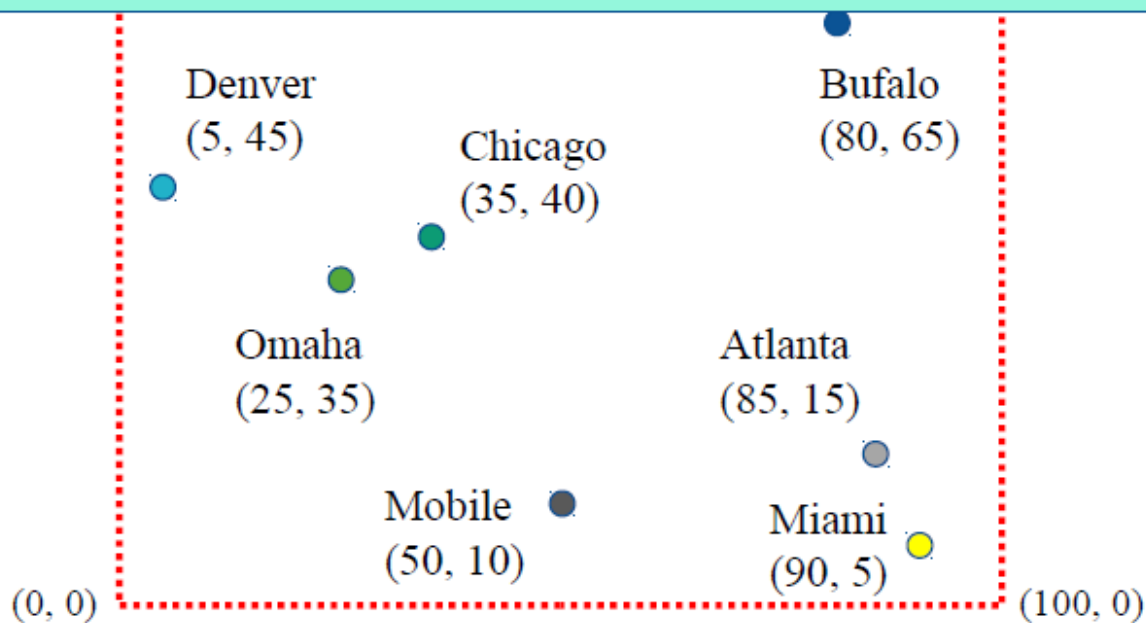


Point QuadTrees - Cidades (pontos)



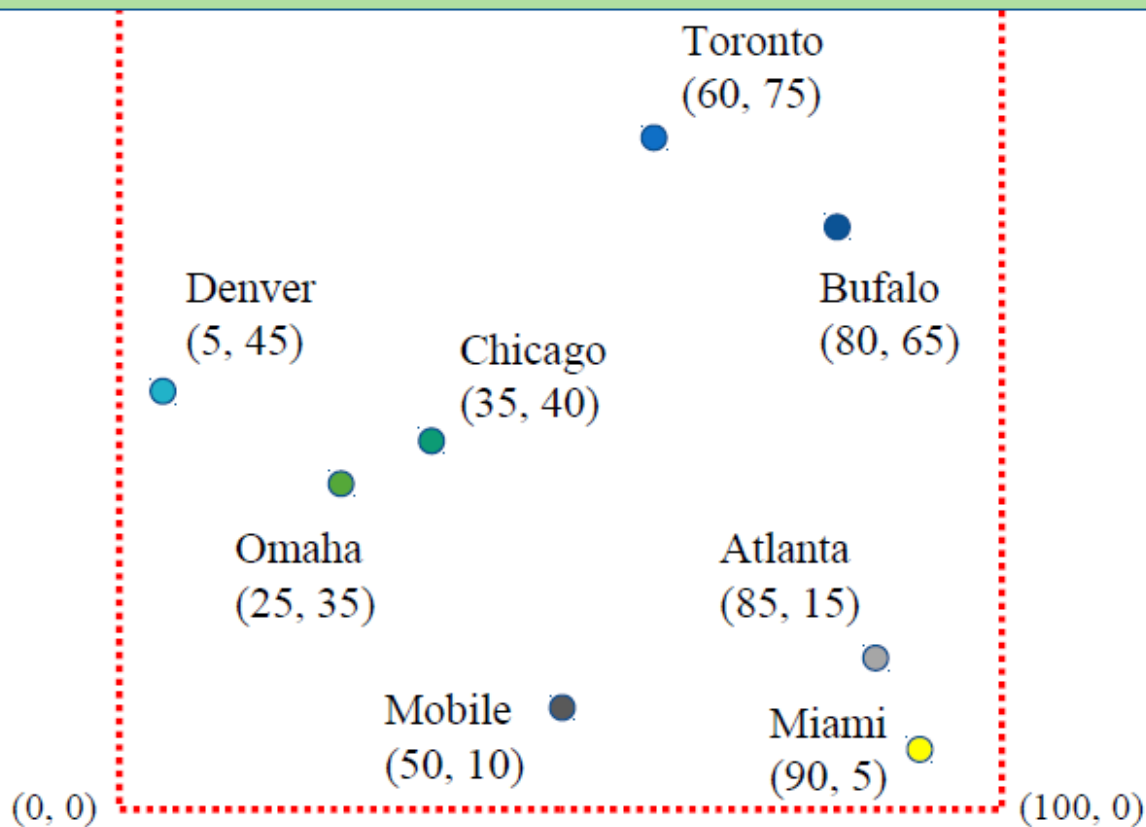
Point QuadTrees - Inserção

Registros são inseridos de forma semelhante à árvore binária de busca. A posição desejada é buscada de acordo com as coordenadas X e Y. Em cada nó uma comparação é feita e a subárvore apropriada (NE, NW, SW ou SE) é escolhida. Quando chega-se na base da árvore (filho nulo), foi encontrada a posição desejada para inserir o registro.

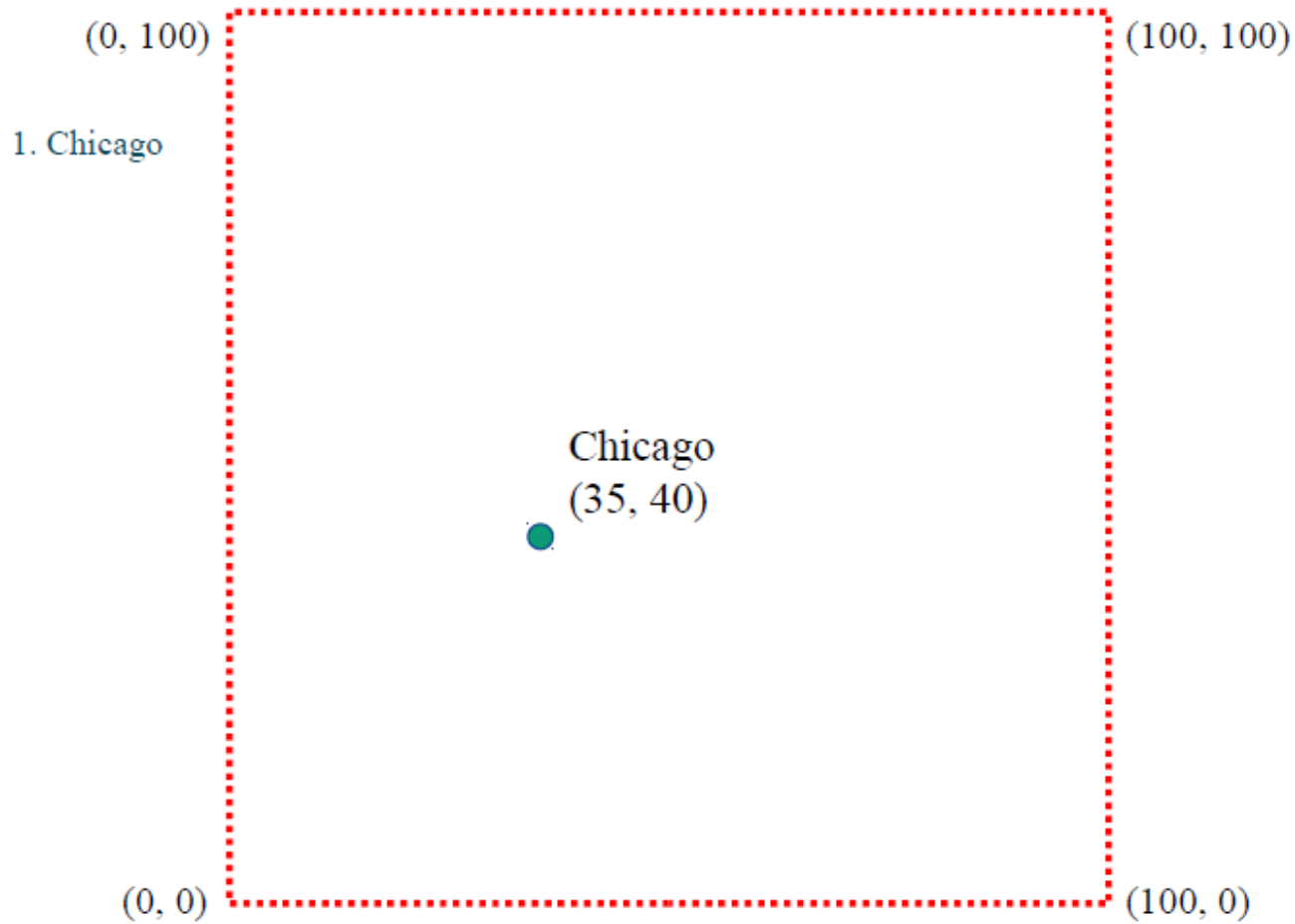


Point QuadTrees - Inserção

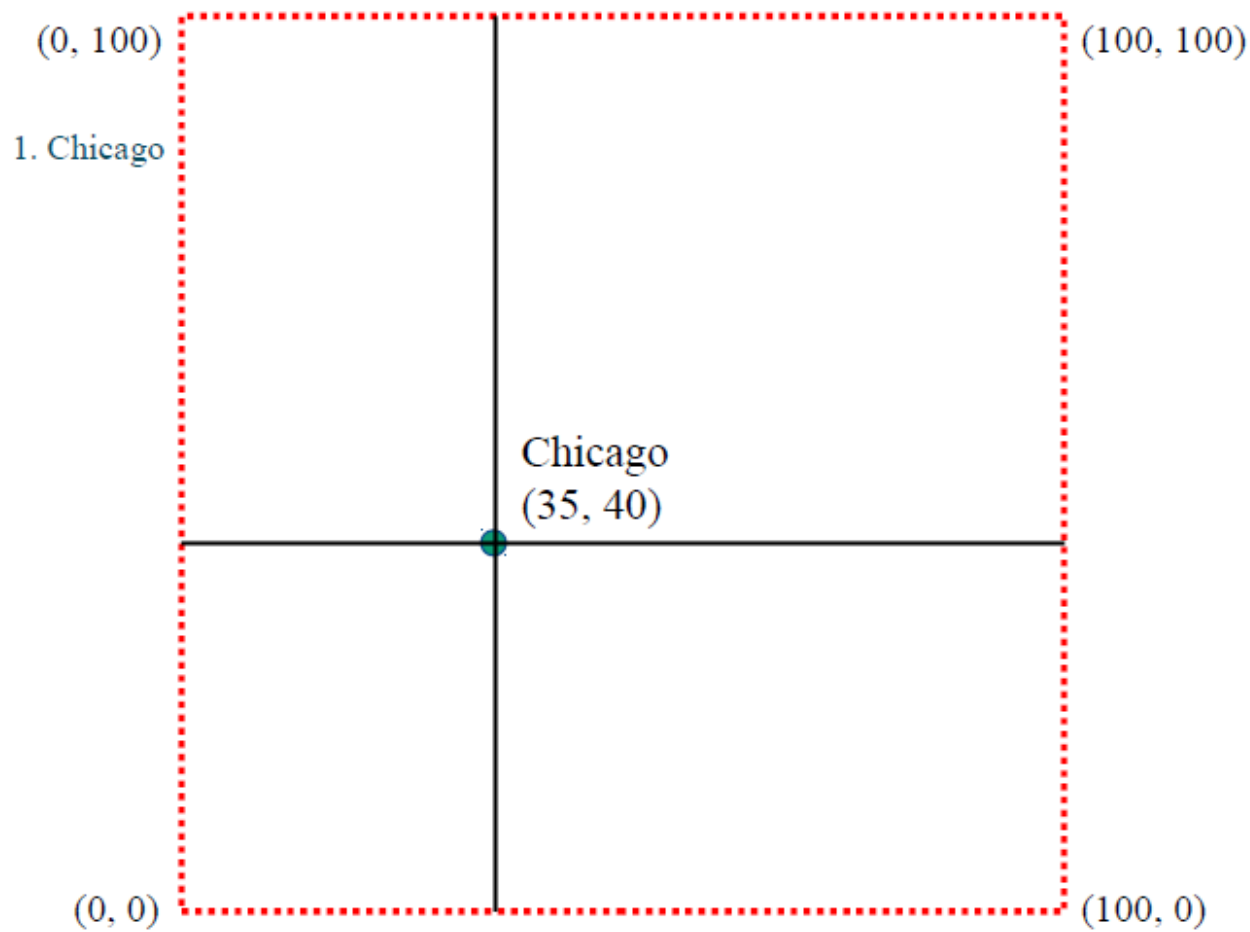
Inserção na ordem: Chicago, Mobile, Toronto, Buffalo, Denver, Omaha, Atlanta e Miami.



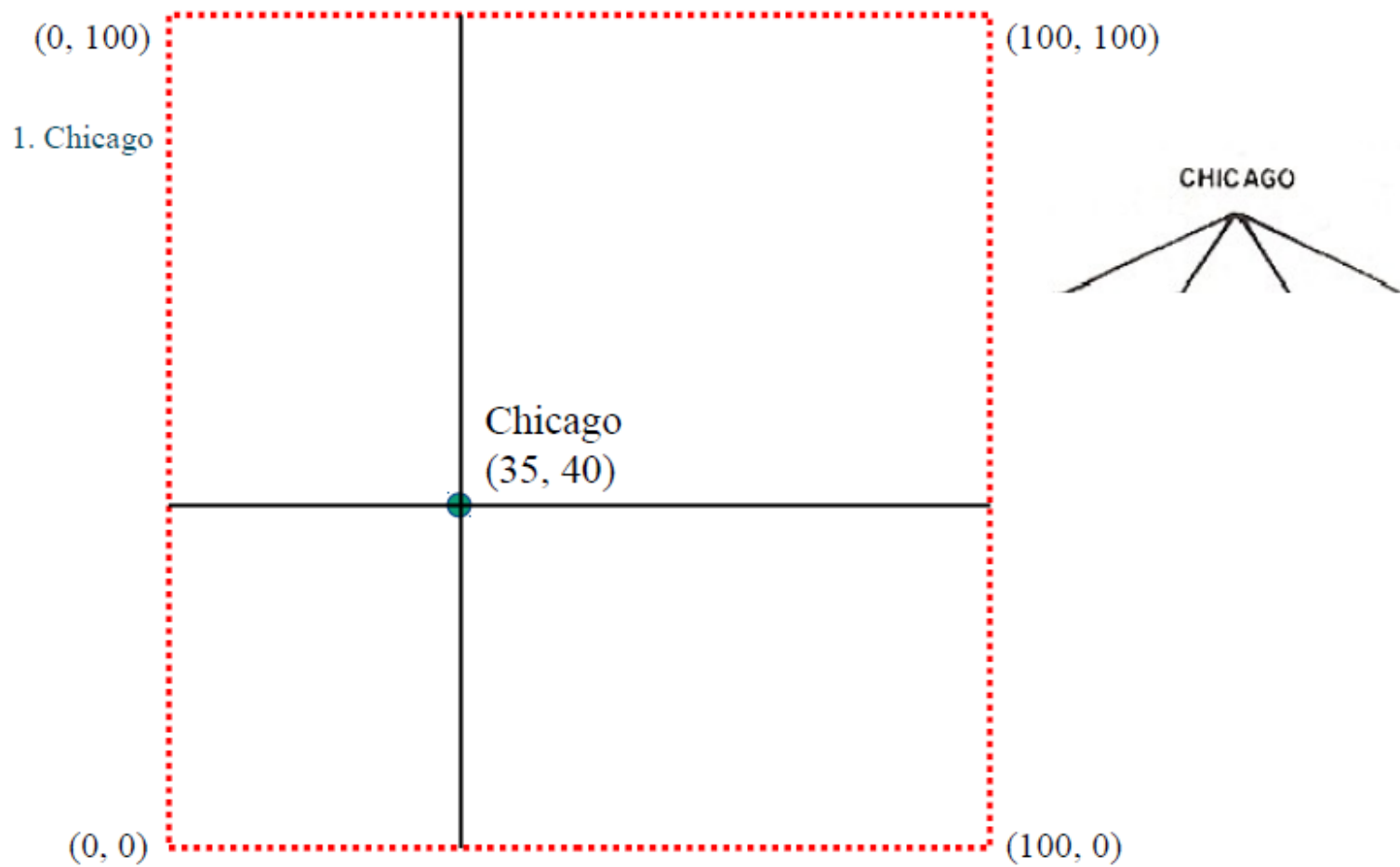
Point QuadTrees - Inserção



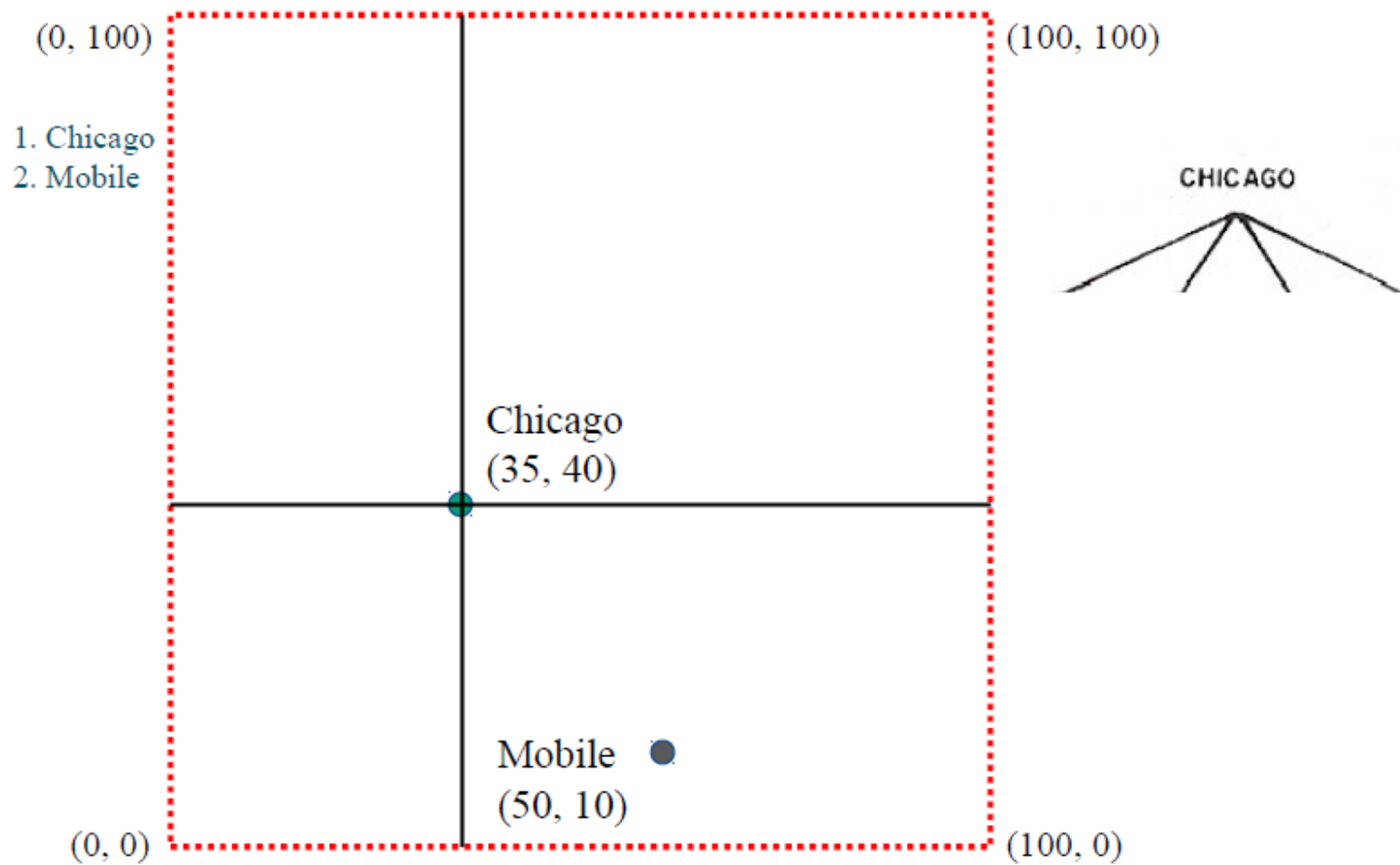
Point QuadTrees - Inserção



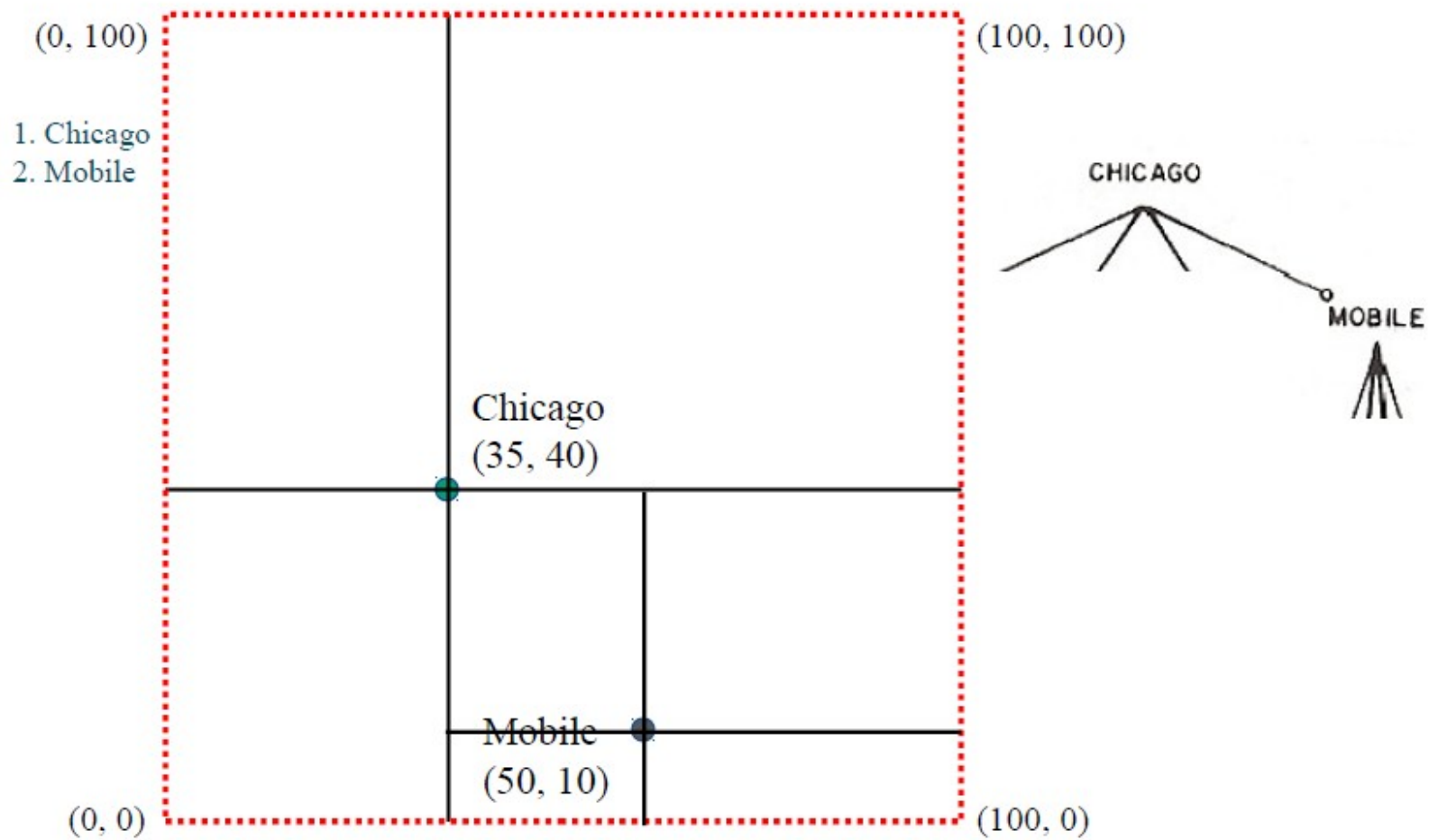
Point QuadTrees - Inserção



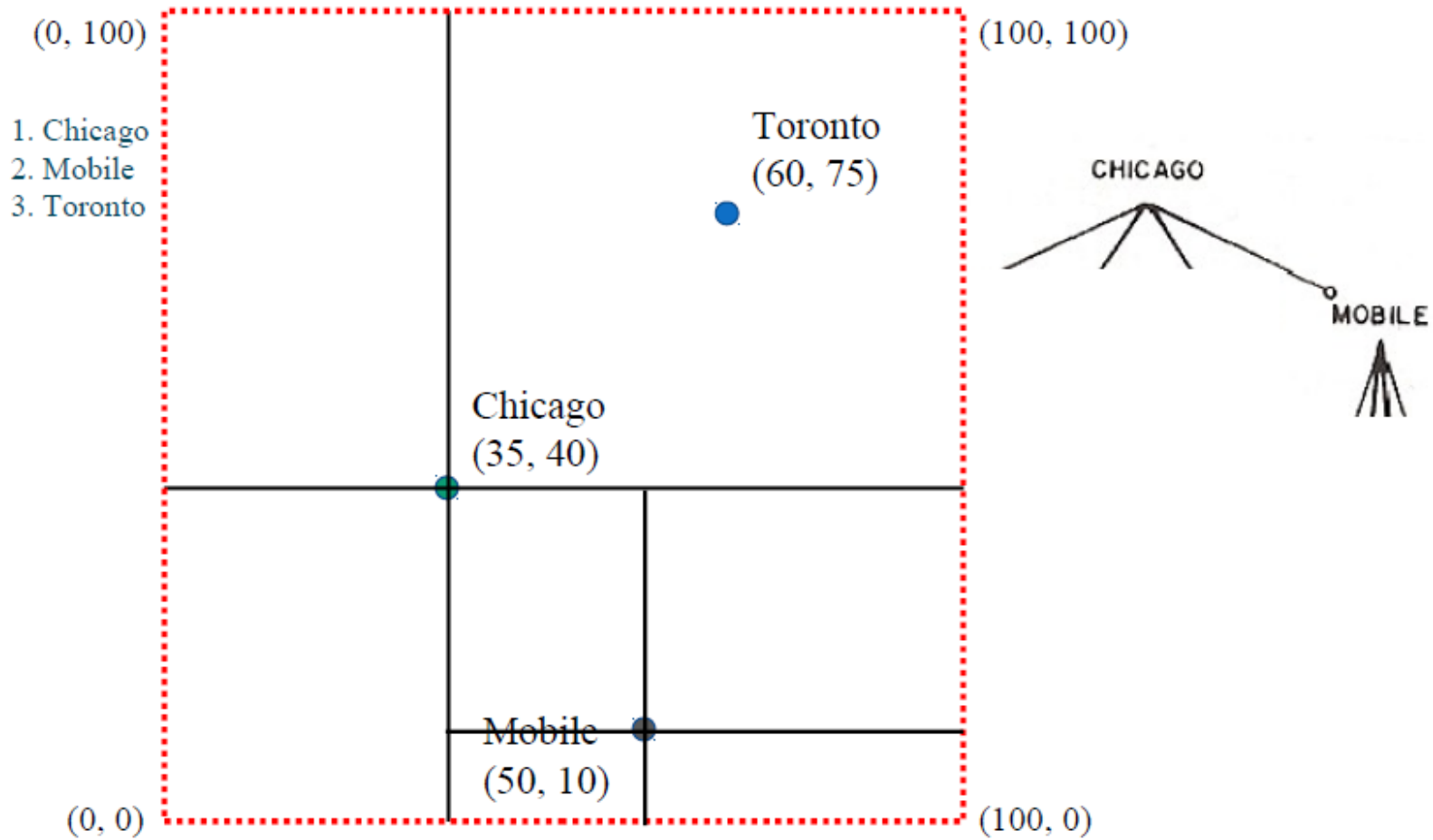
Point QuadTrees - Inserção



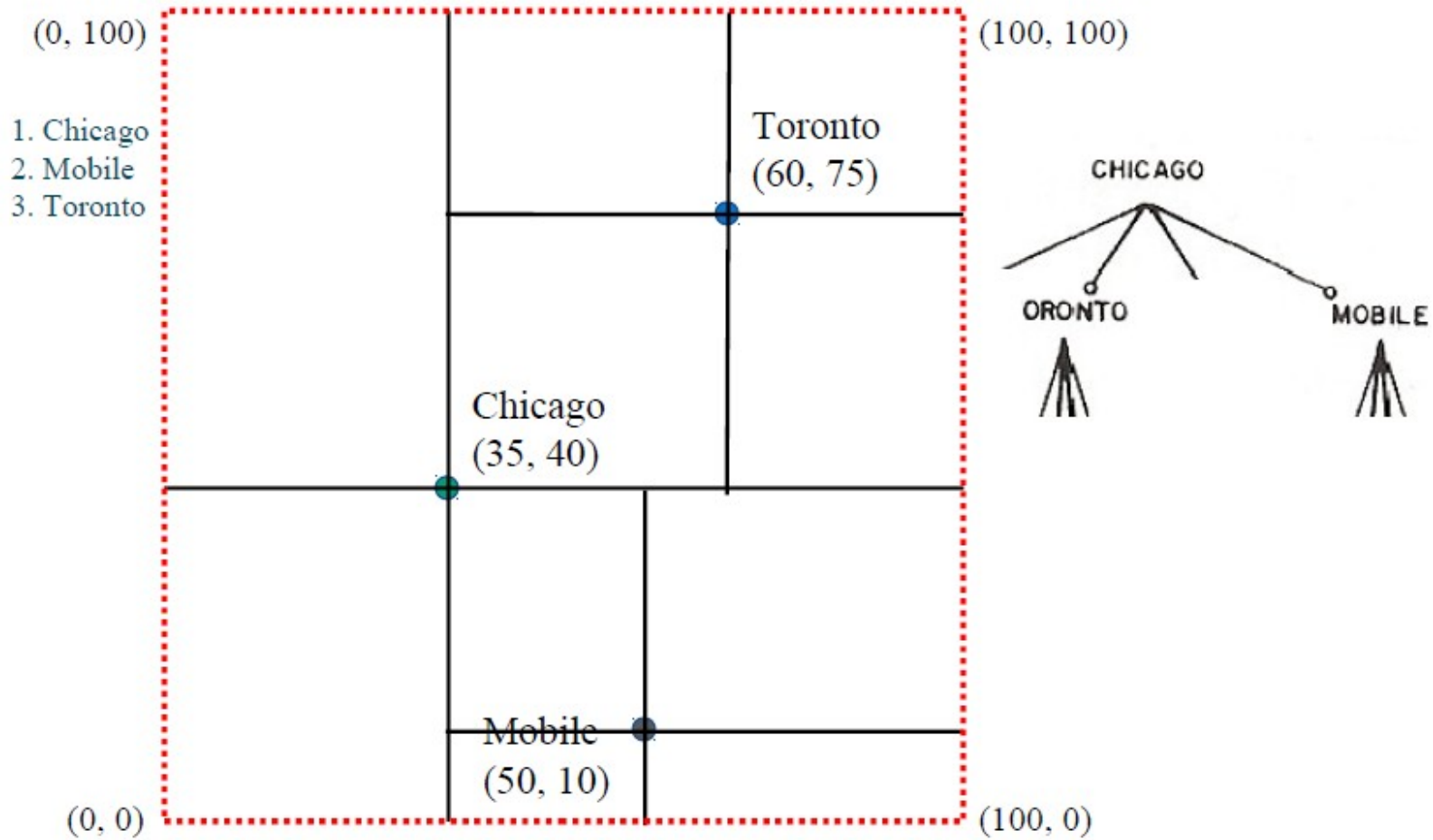
Point QuadTrees - Inserção



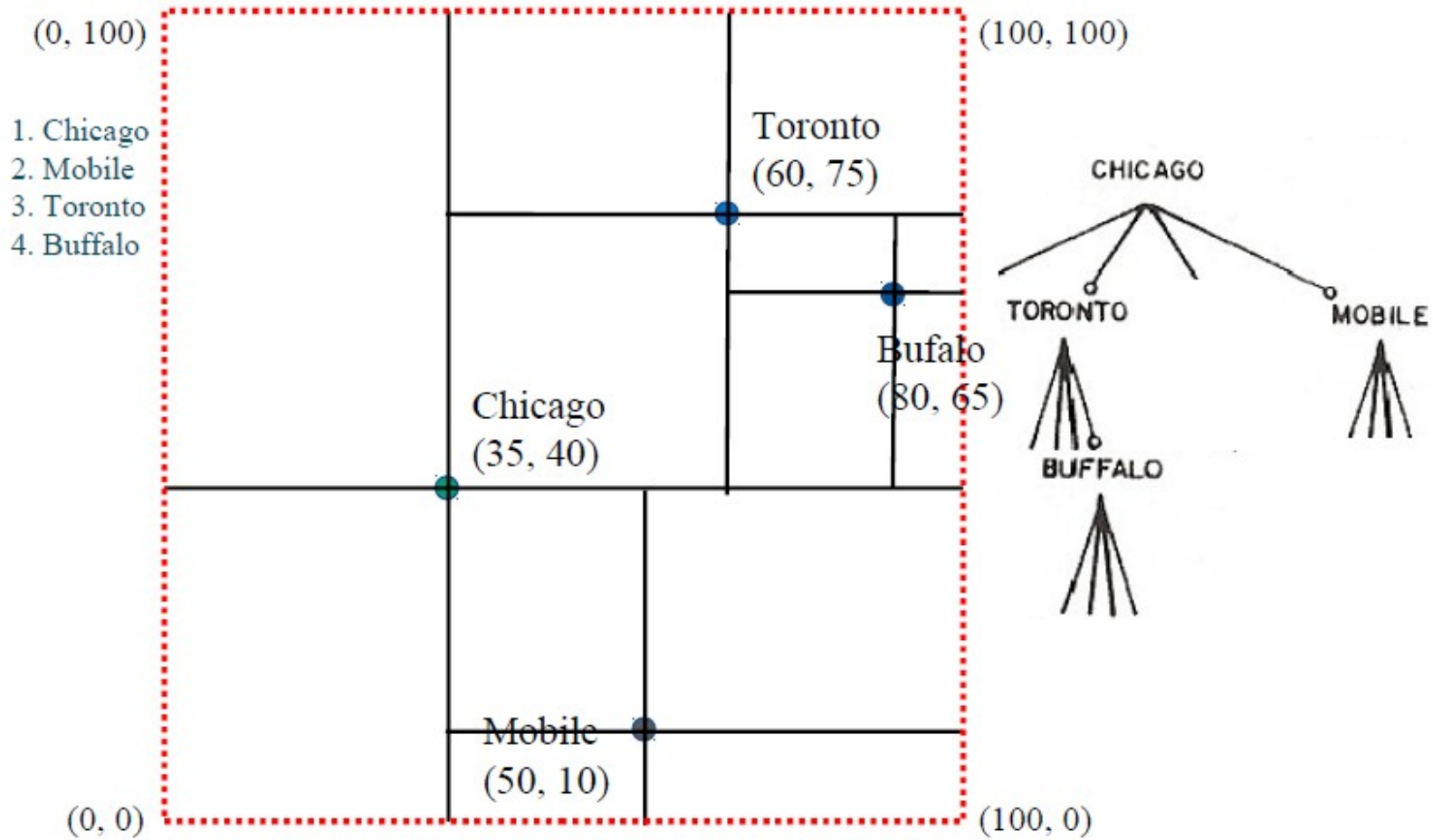
Point QuadTrees - Inserção



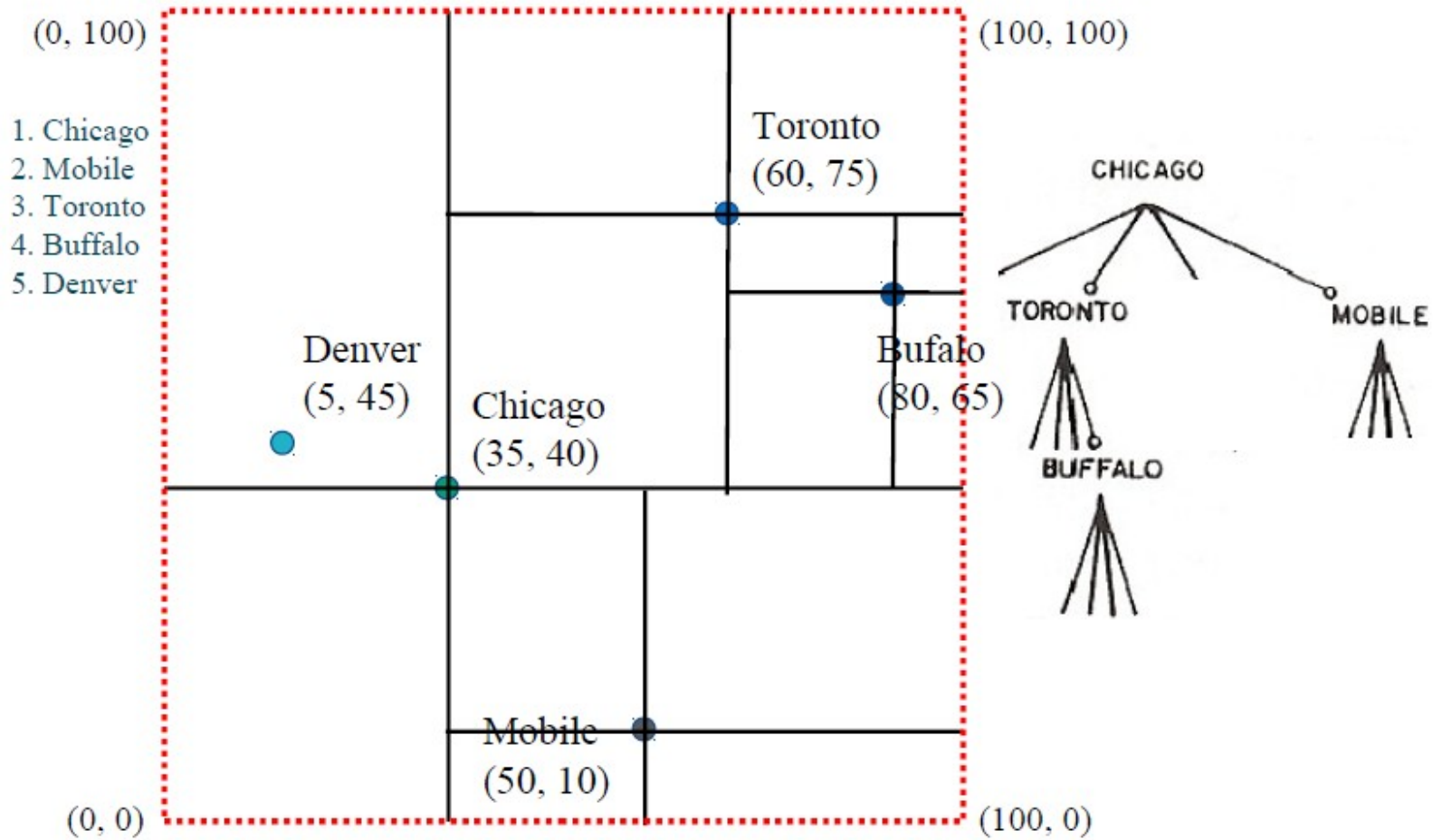
Point QuadTrees - Inserção



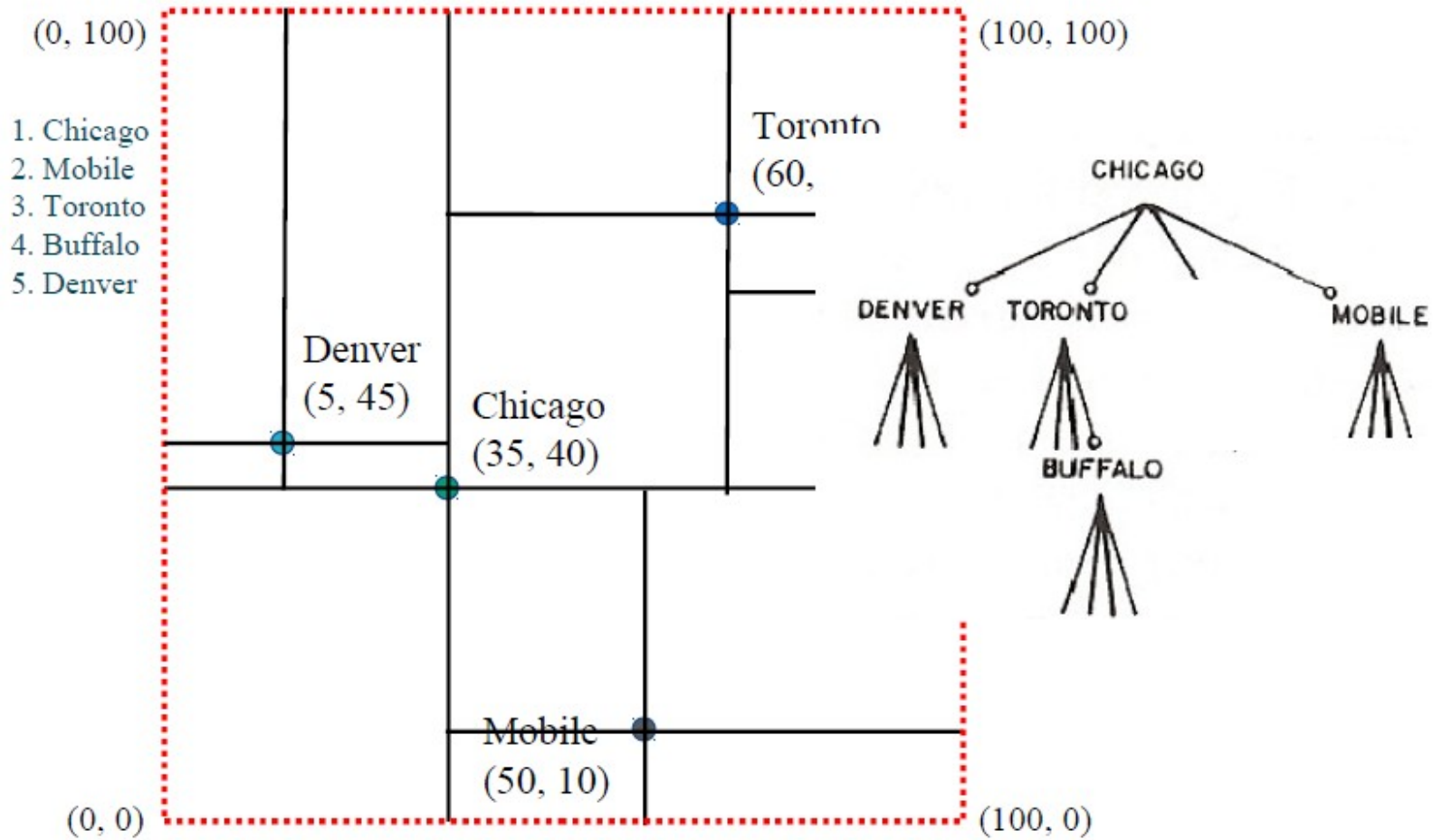
Point QuadTrees - Inserção



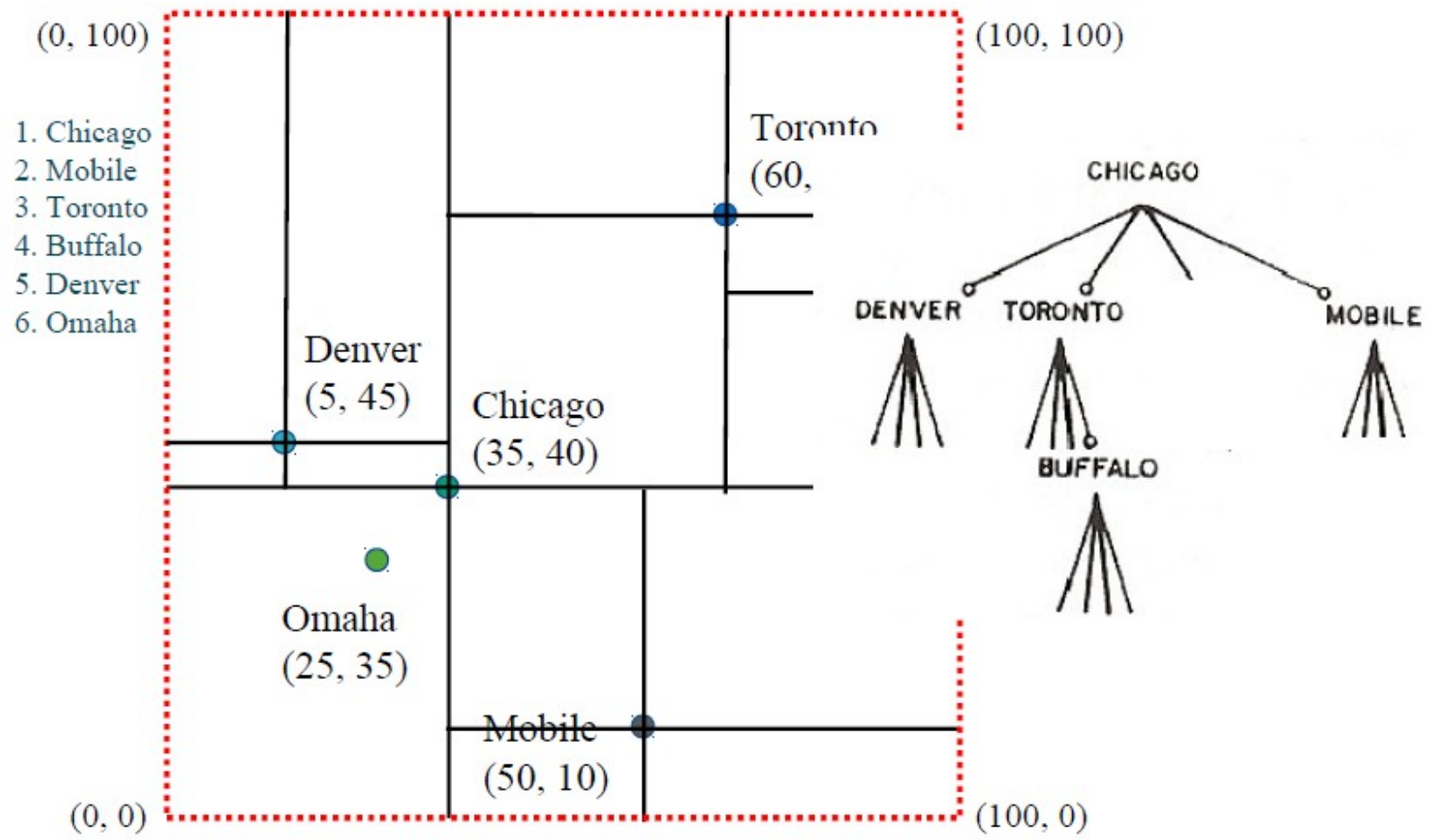
Point QuadTrees - Inserção



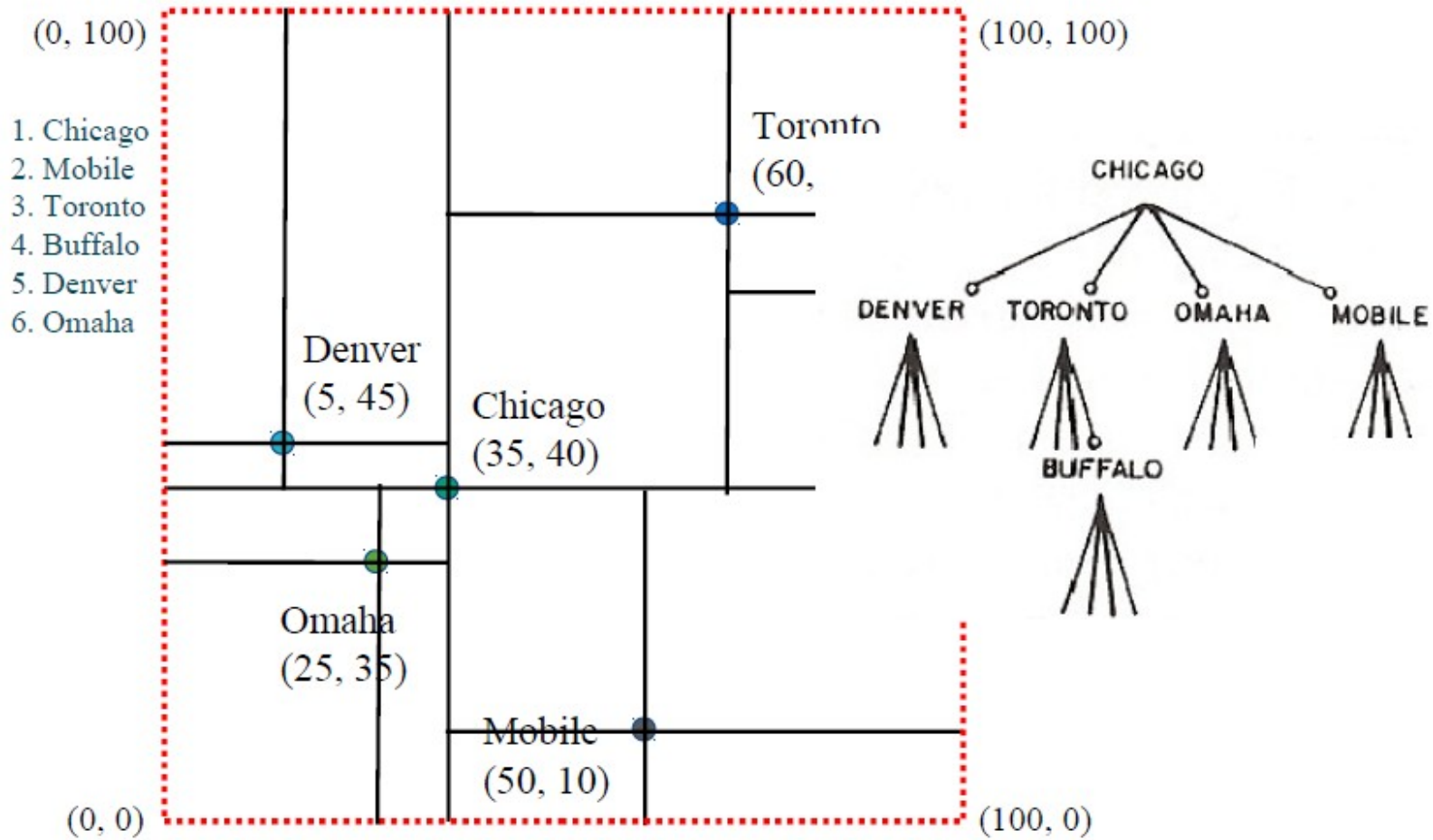
Point QuadTrees - Inserção



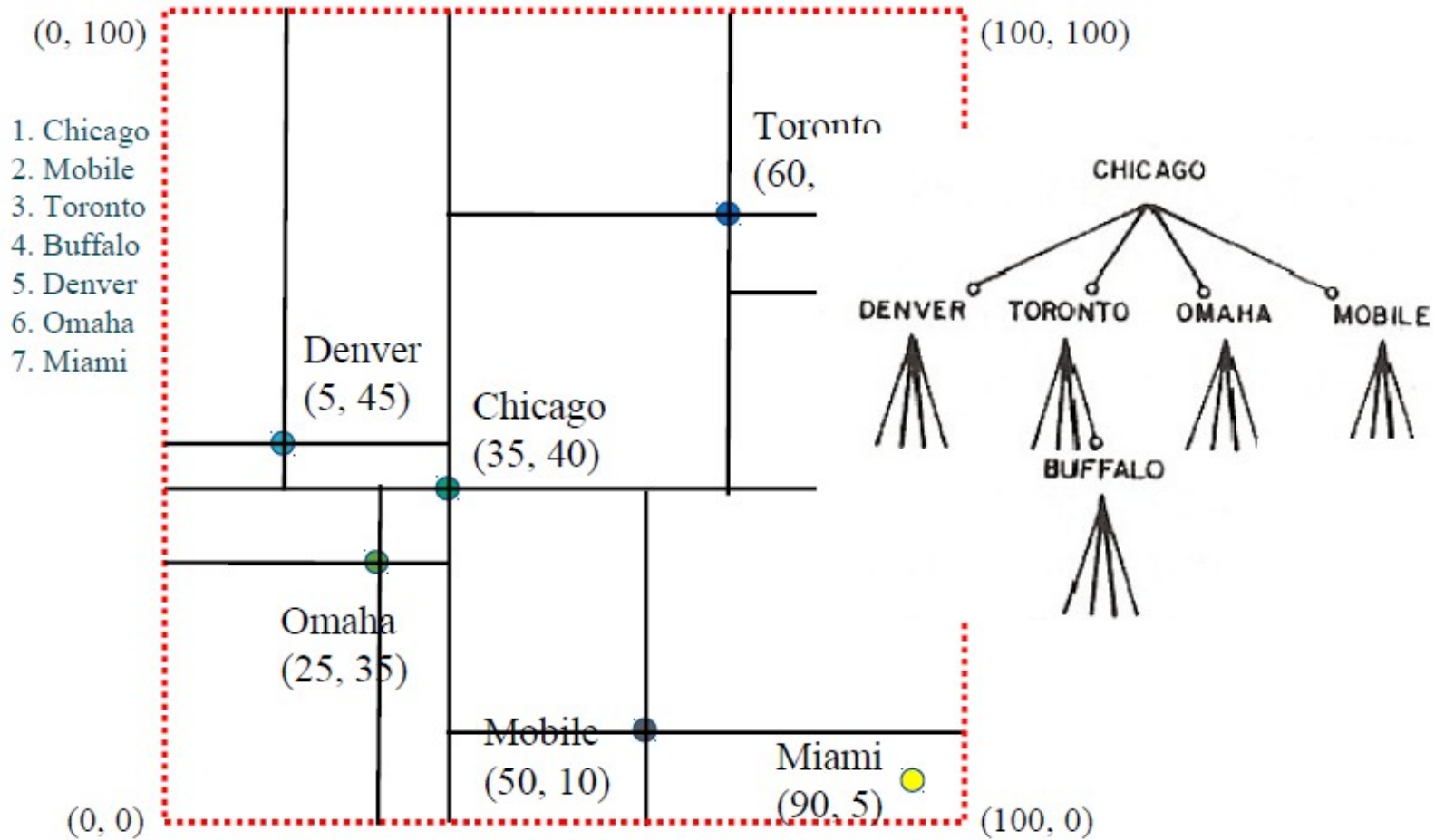
Point QuadTrees - Inserção



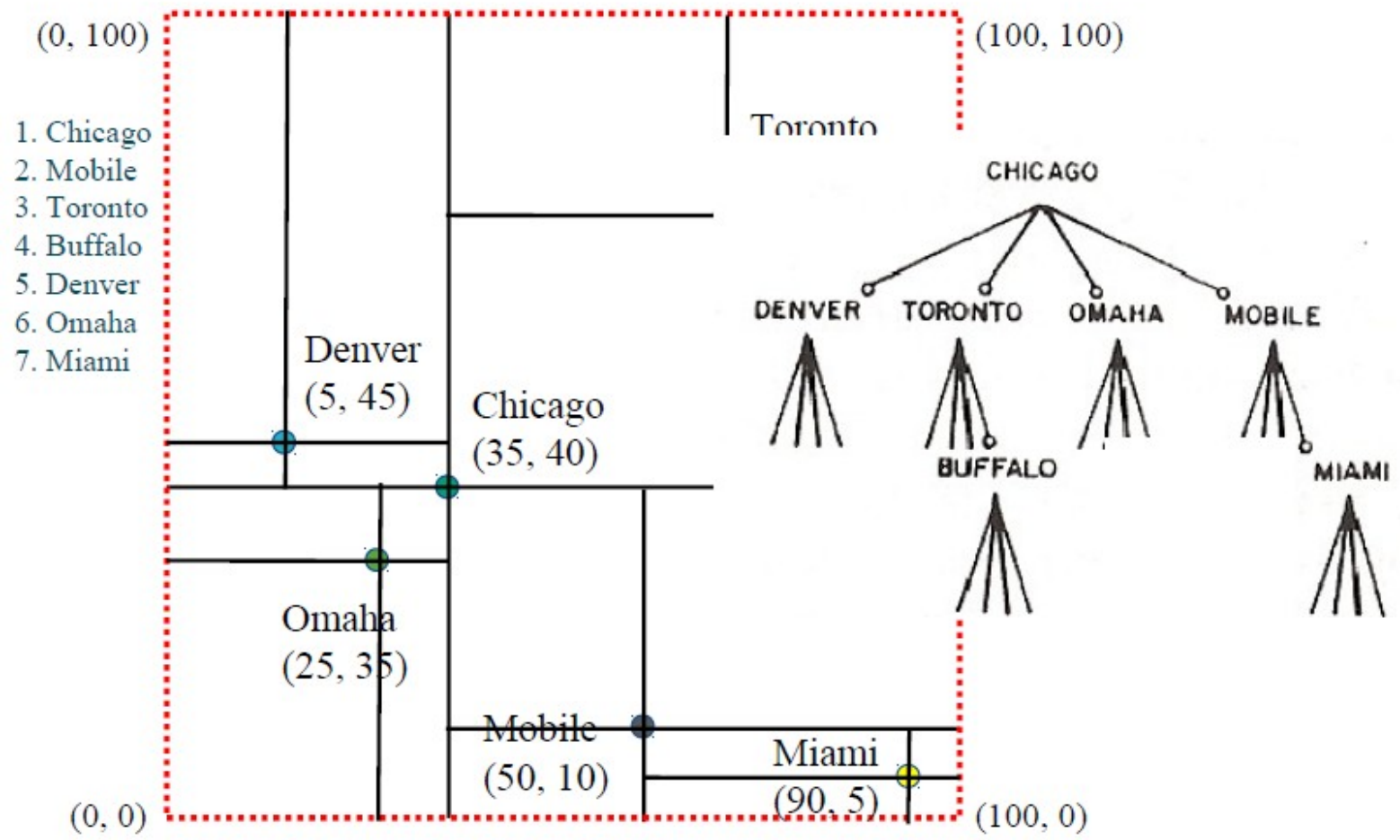
Point QuadTrees - Inserção



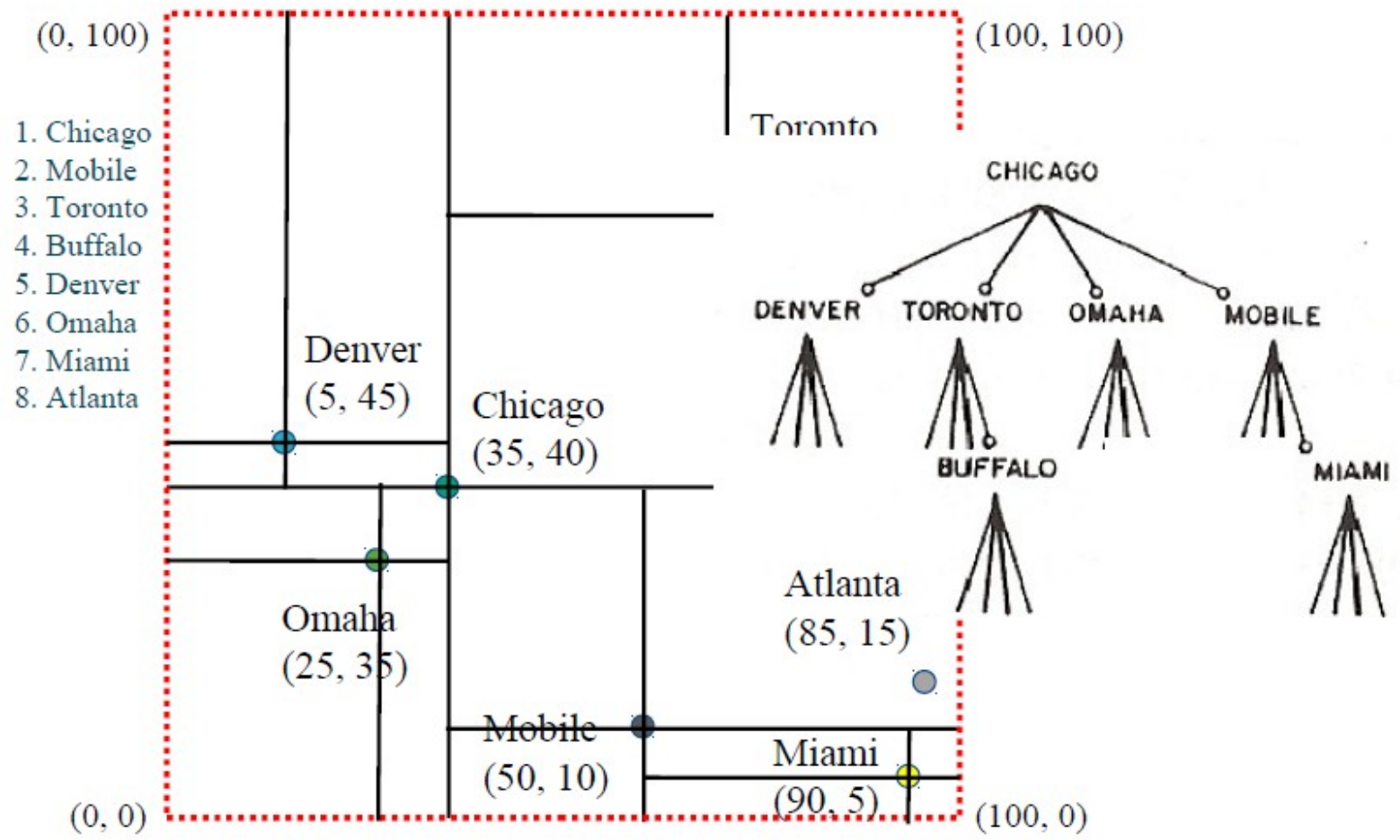
Point QuadTrees - Inserção



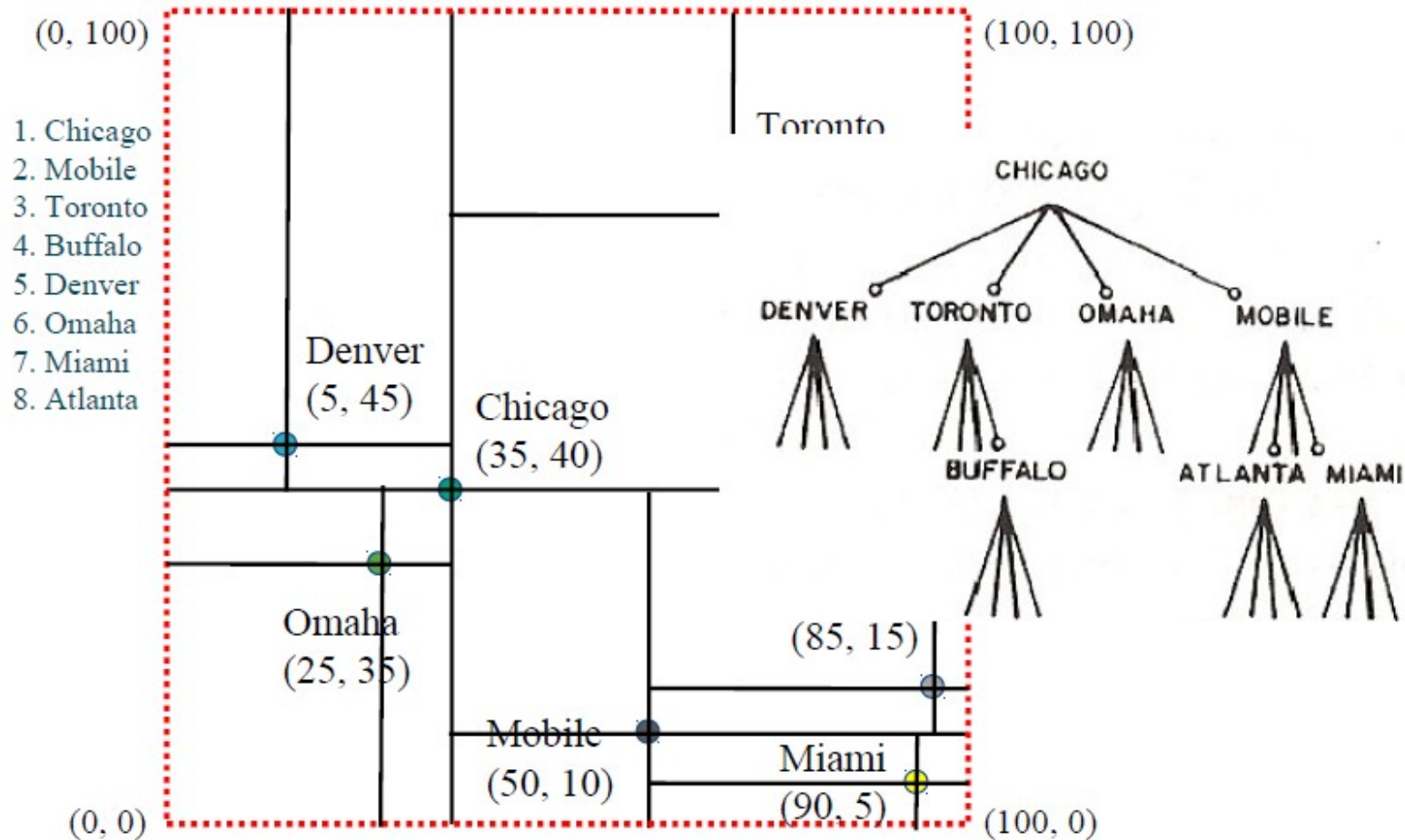
Point QuadTrees - Inserção



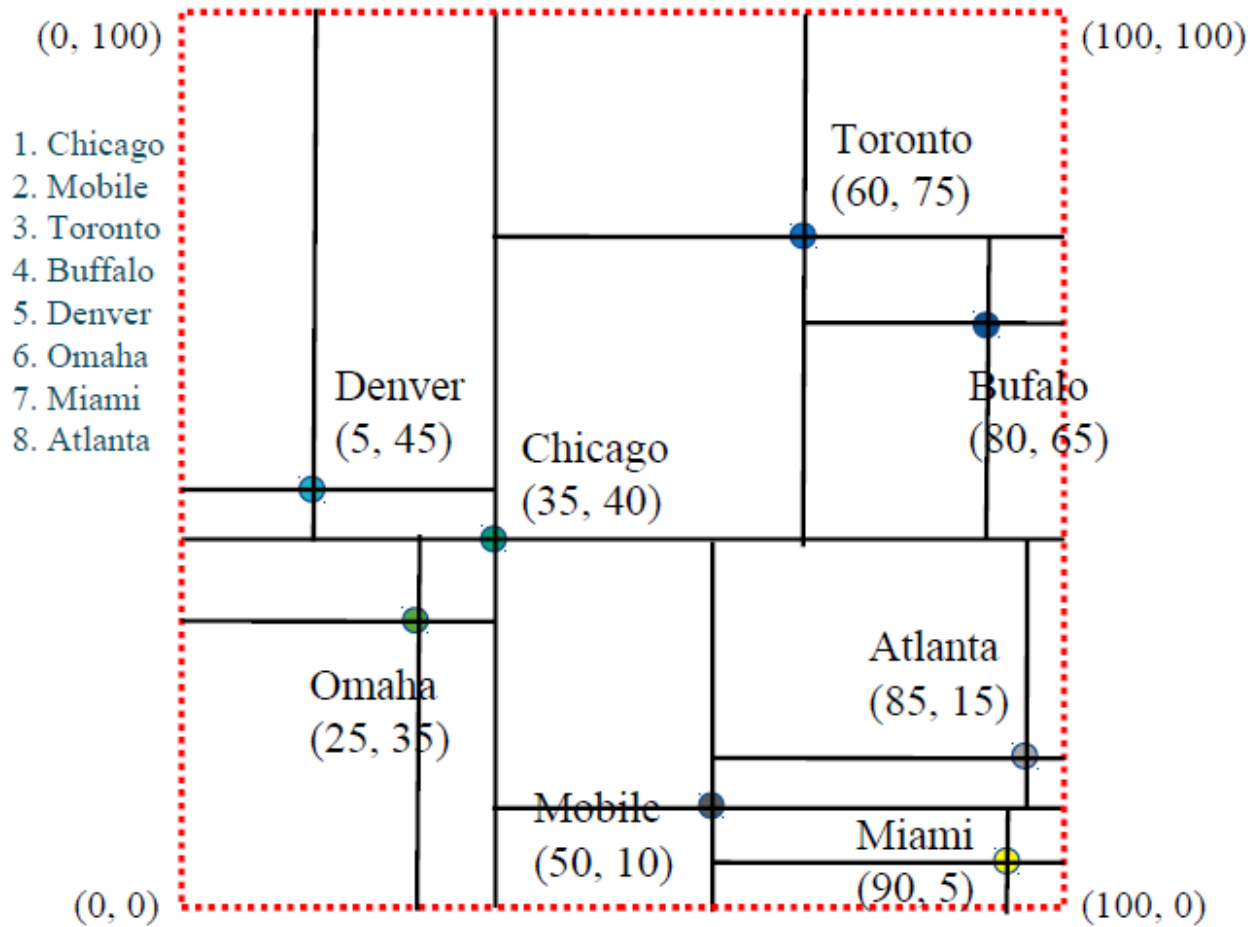
Point QuadTrees - Inserção



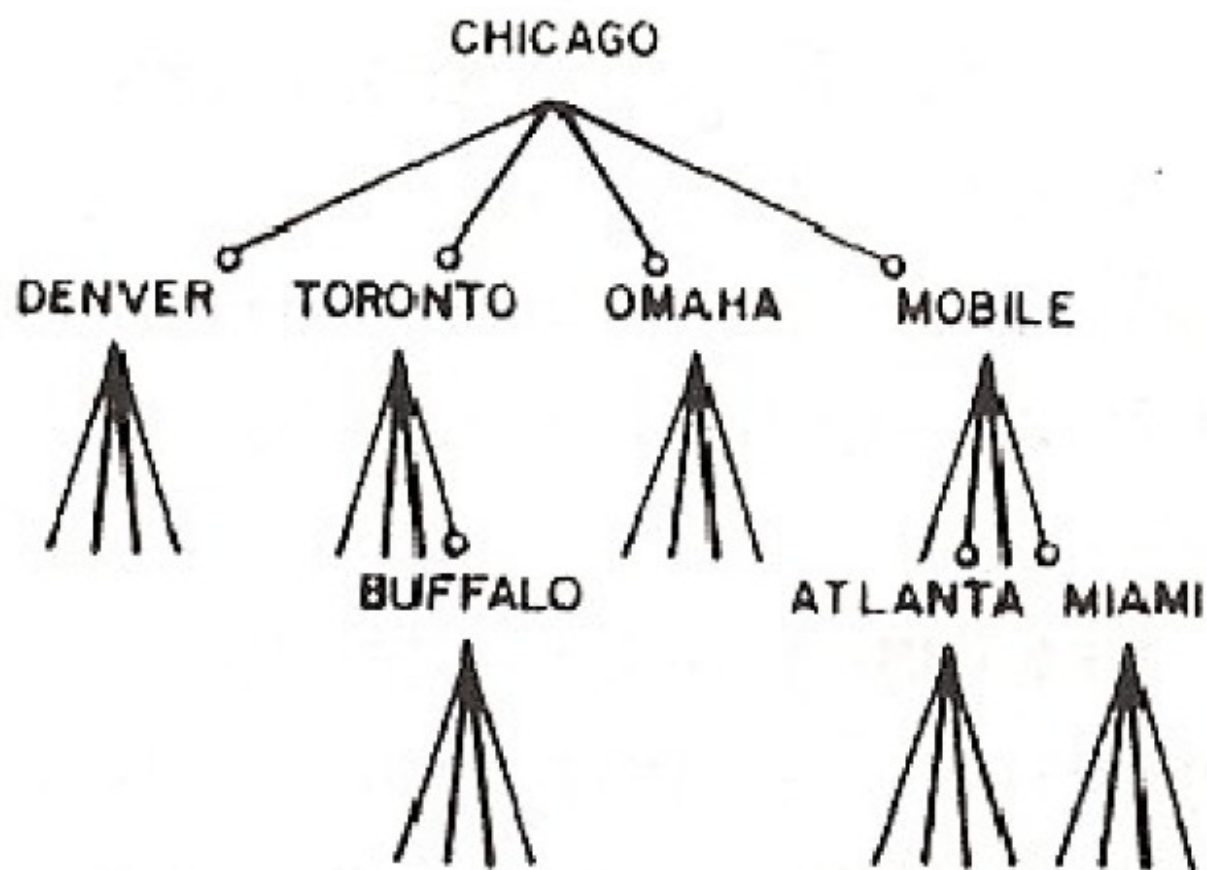
Point QuadTrees - Inserção



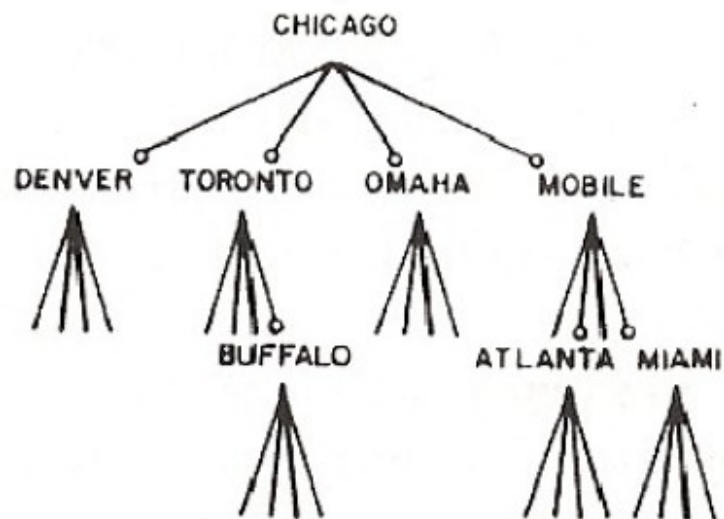
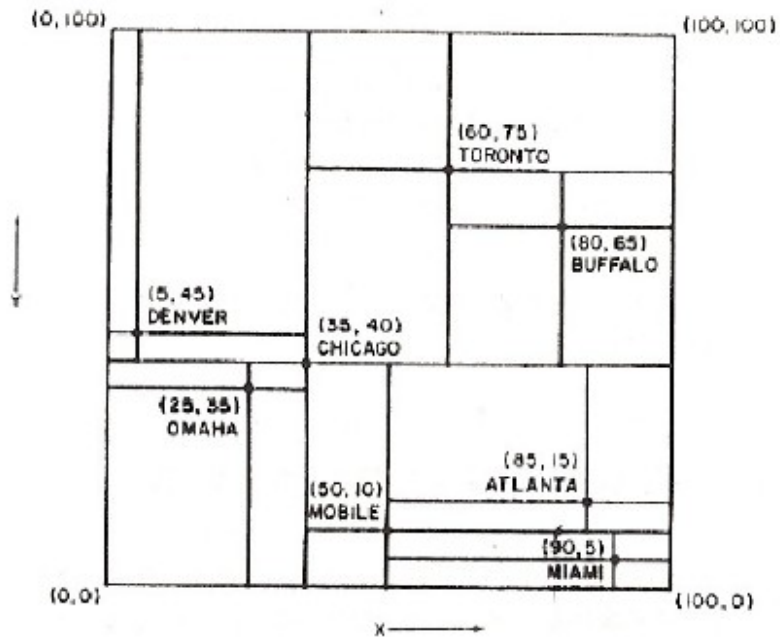
Point QuadTrees - Inserção



Point QuadTrees - Inserção

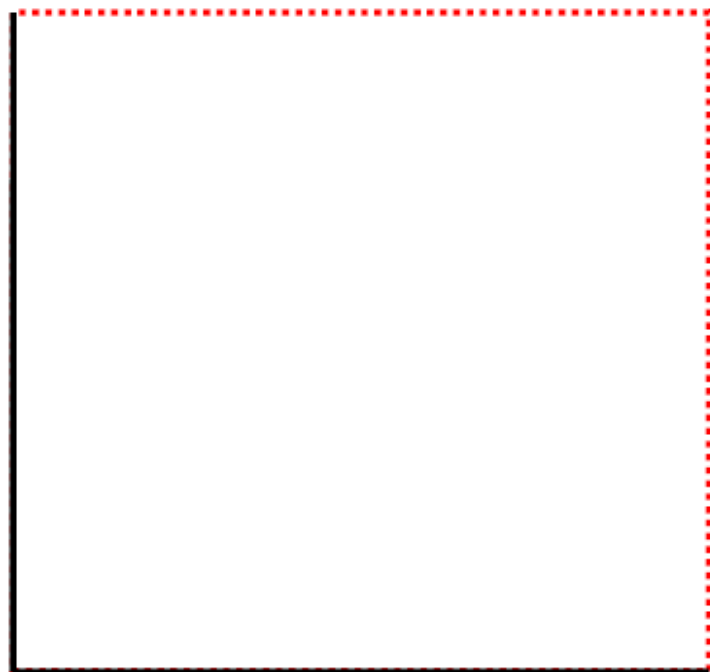


Point QuadTrees - Exemplo



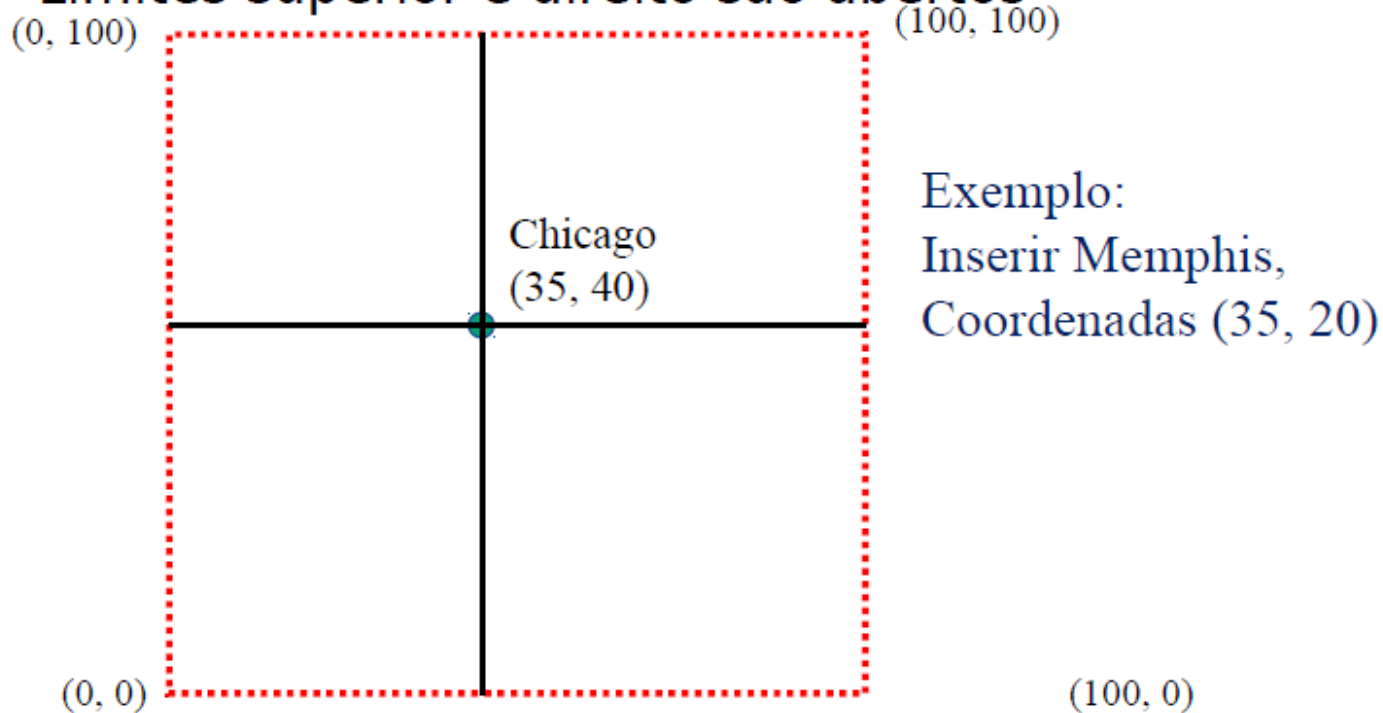
Ponto sobre a linha de um quadrante

- Regra adotada
 - Limites inferior e esquerdo de cada bloco são fechados
 - Limites superior e direito são abertos



Ponto sobre a linha de um quadrante

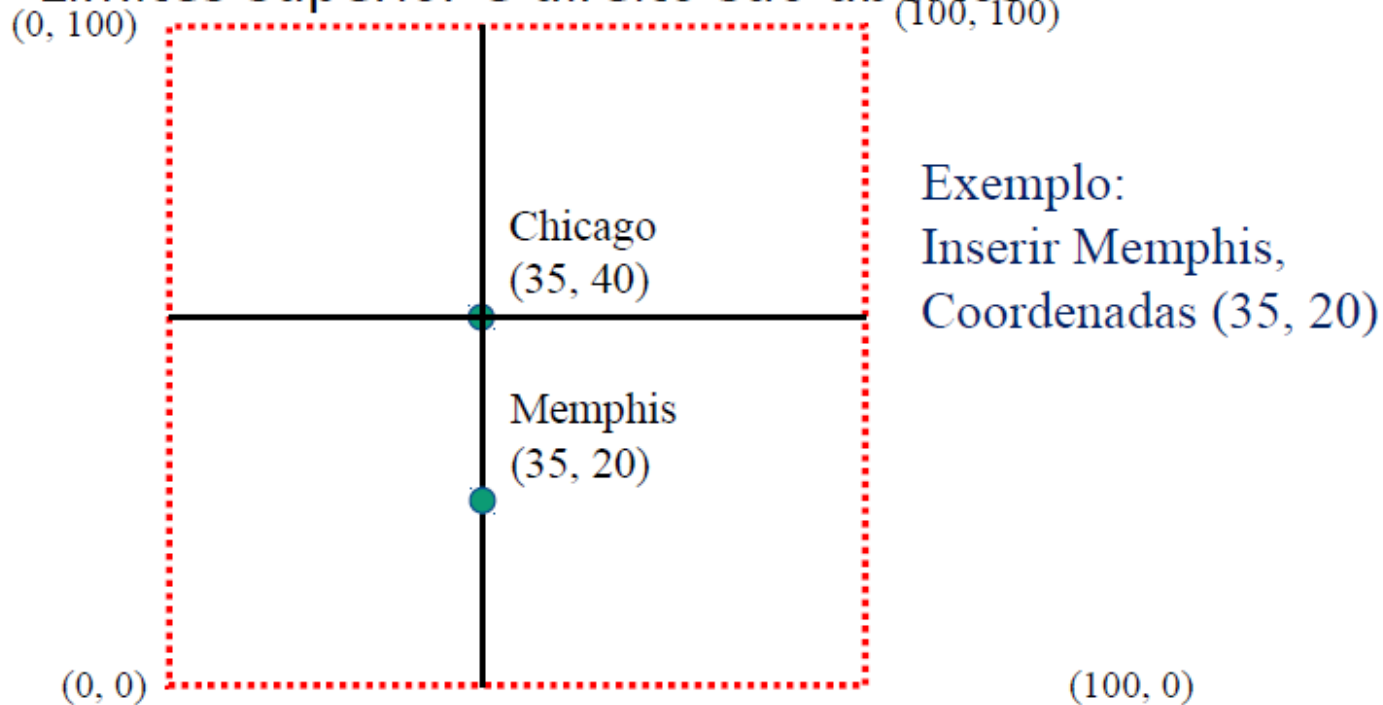
- Regra adotada
 - Limites inferior e esquerdo de cada bloco são fechados
 - Limites superior e direito são abertos



Ponto sobre a linha de um quadrante

- Regra adotada

- Limites inferior e esquerdo de cada bloco são fechados
- Limites superior e direito são abertos

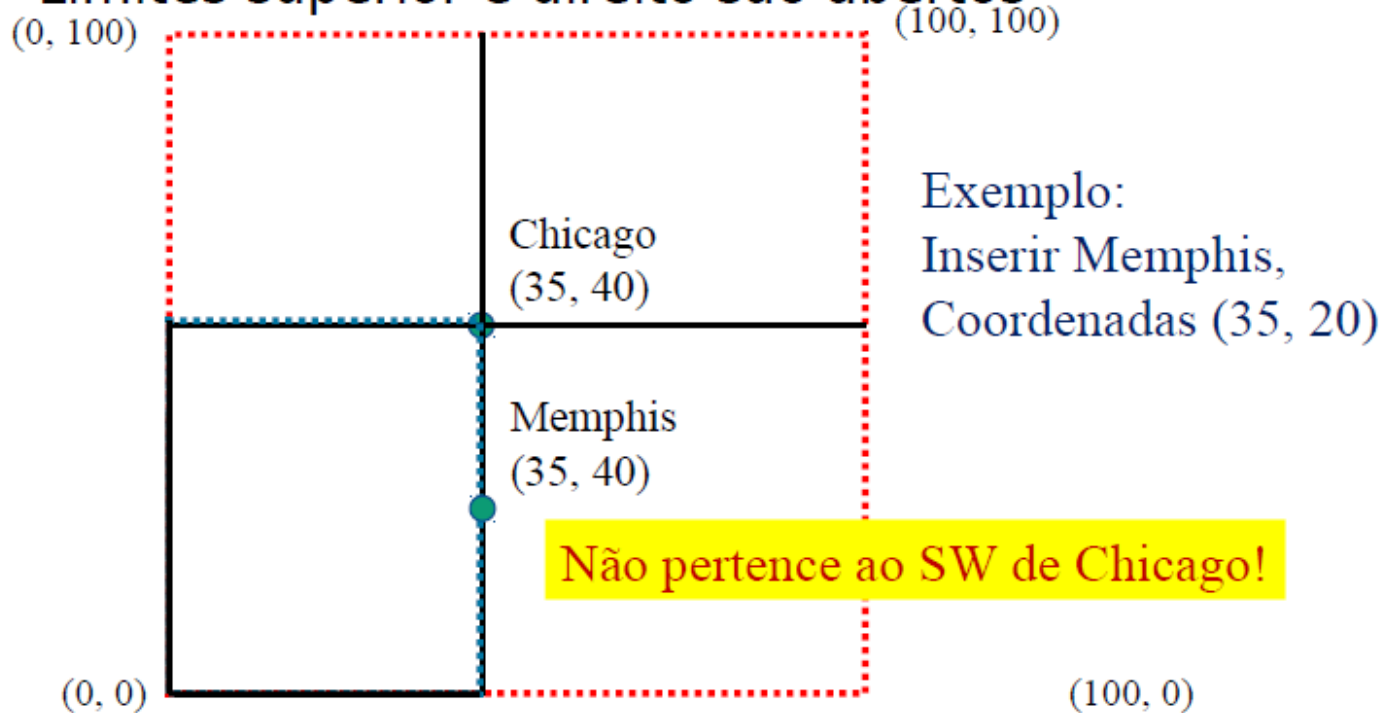


Ponto sobre a linha de um quadrante

- Regra adotada

- Limites inferior e esquerdo de cada bloco são fechados

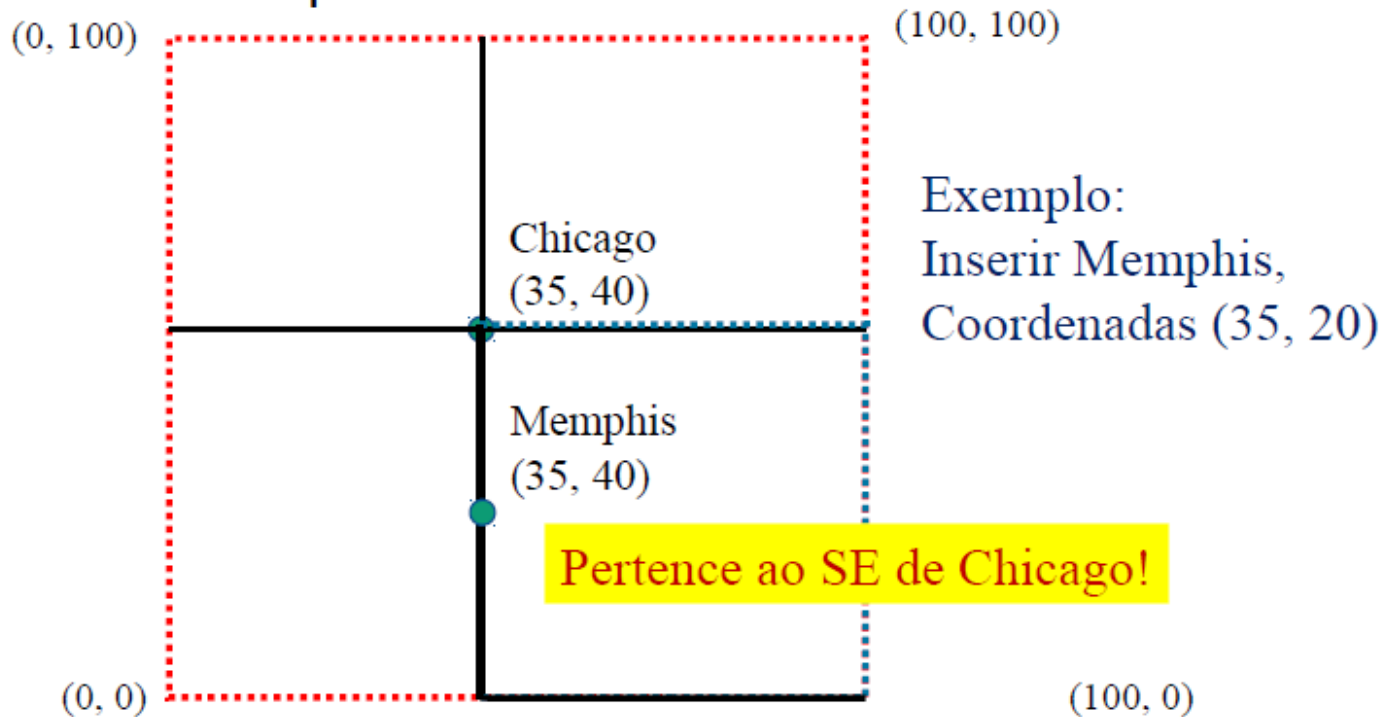
- Limites superior e direito são abertos



Ponto sobre a linha de um quadrante

- Regra adotada

- Limites inferior e esquerdo de cada bloco são fechados
- Limites superior e direito são abertos



Exercícios:

- Proponha a estrutura de um nó;
- Escreva um código de busca numa quadtree;

Algoritmo de inserção

```
procedure PT_INSERT(P,R);
/* Attempt to insert node P in the point quadtree rooted at node R. */
begin
  value pointer node P;
  reference pointer node R;
  pointer node F;
  quadrant Q;
  if null(R) then R ← P /* The tree at R is initially empty */
  else
    begin
      while not(null(R)) and not(EQUAL_COORD(P,R)) do
        begin
          F ← R; /* Remember the father */
          Q ← PT_COMPARE(P,R);
          R ← SON(R,Q);
        end;
      if null(R) then SON(F,Q) ← P; /* P is not already in the tree */
    end;
  end;
```

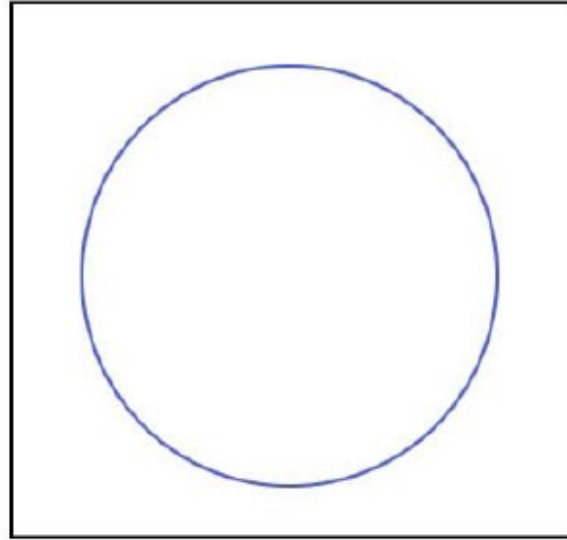
Exercício

- Mostre a QuadTree resultante após a inserção dos pontos:
 - (A, 51,30)
 - (B, 13,70)
 - (C, 81,40)
 - (D, 81,70)
 - (E, 02,25)
 - (F, 01,01)
 - (G, 99,99)
 - (H, 63,30)
 - (I, 70,67)
 - (J, 50,50)

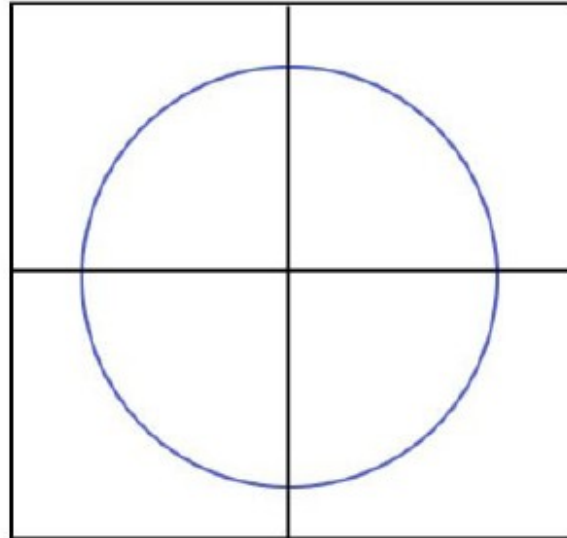
QuadTree para polígonos

- Uma QuadTree pode também ser utilizada para preencher polígonos
- Subdividir o espaço
 - Se o quadrante for homogêneo ou se o nível de detalhe foi alcançado, pára a divisão
 - Caso contrário, continue dividindo

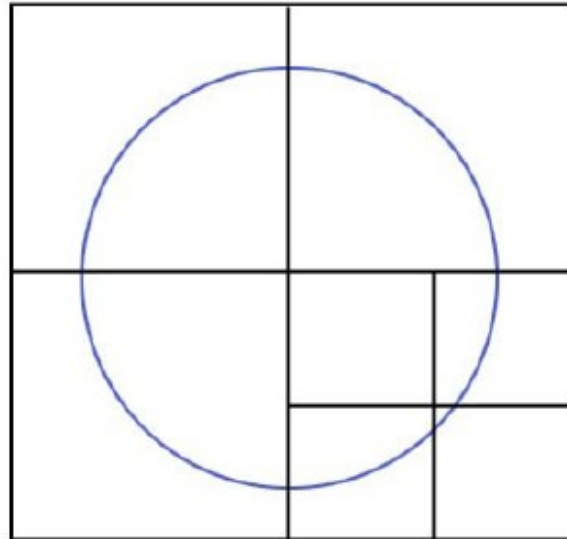
QuadTree para polígonos



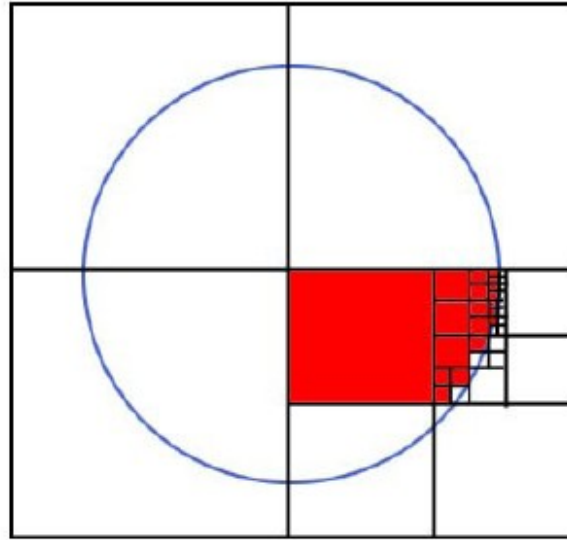
QuadTree para polígonos



QuadTree para polígonos

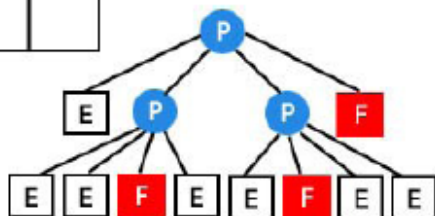
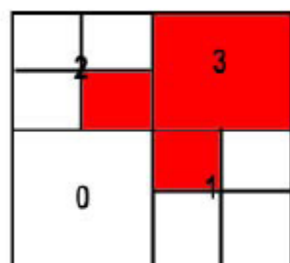


QuadTree para polígonos



QuadTree para polígonos

- As divisões são representadas na árvore como nós caso sejam quadrantes heterogêneos e como folhas caso sejam quadrantes homogêneos.

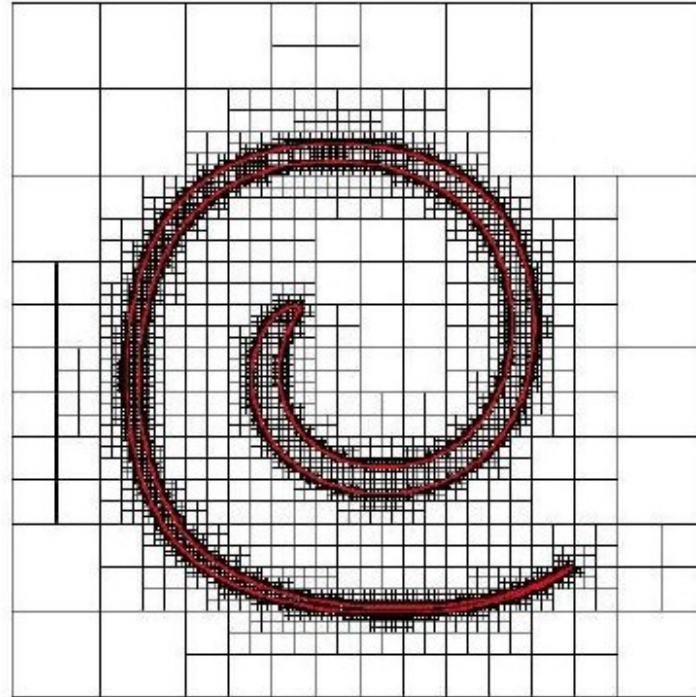


E = Empty

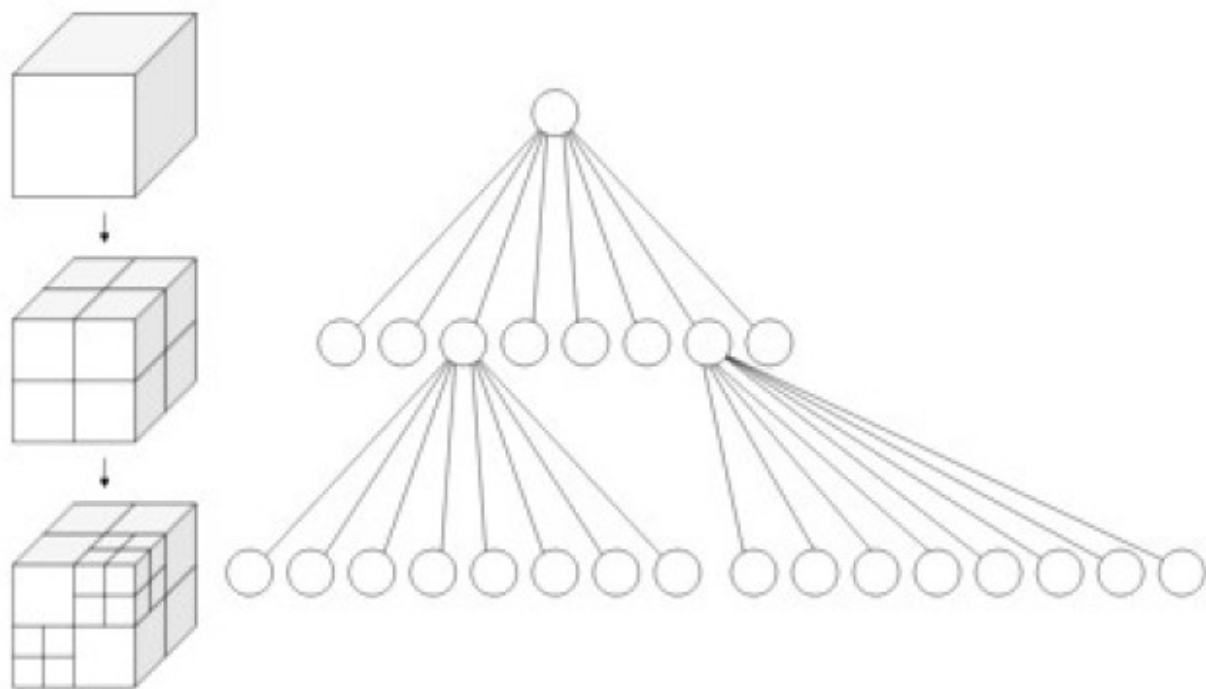
F = Full

P = Partially full

QuadTree para polígonos

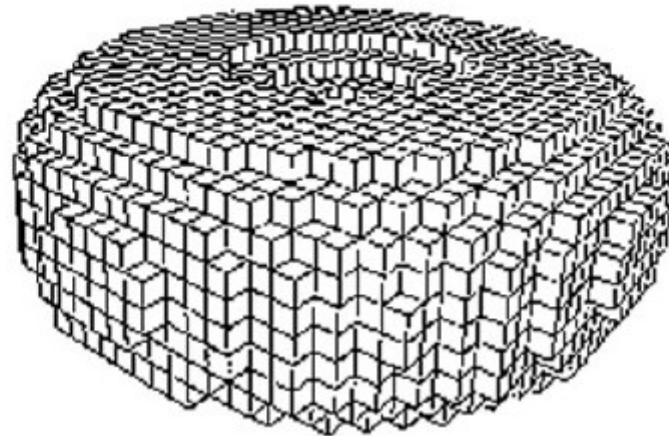


OctTree



Octree Para Poliedros

- A mesma ideia da QuadTree para polígonos



OctTree

