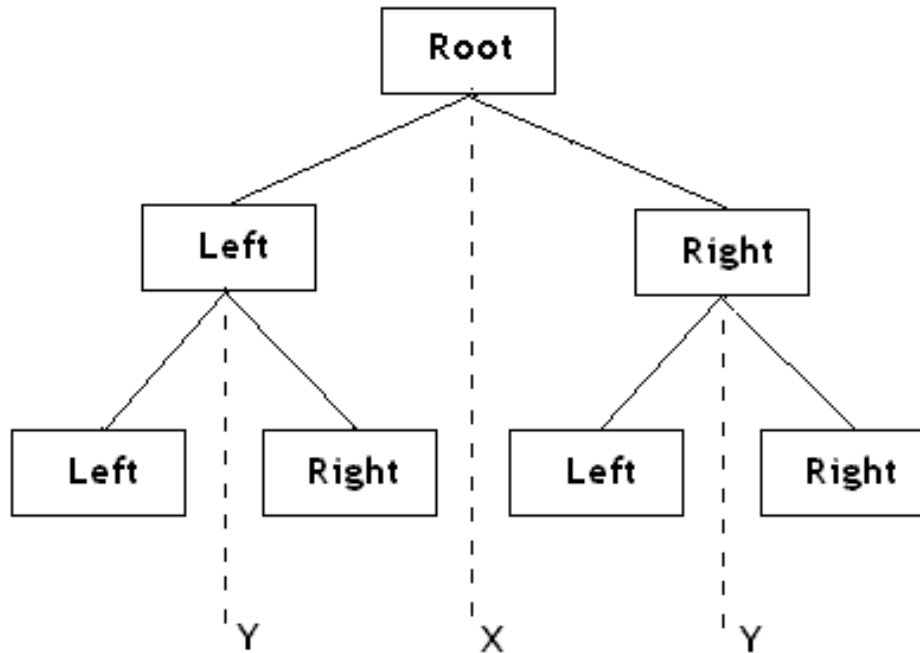


# KD-TREES



Aluno: Pedro Miguel Clemente Dias Moreira, n.º 10015

Estrutura de Dados e Algoritmos – 2º Ano Engenharia Informática

Escola Superior de Tecnologia e Gestão de Beja

Docente: José Jasnau Caeiro

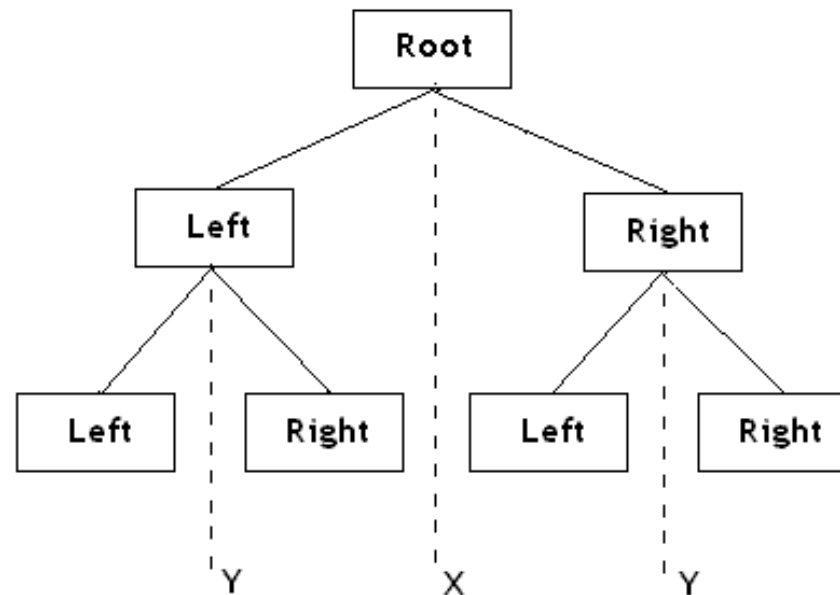
# KD-TREES

- **O que é?**
  - Estrutura de dados de partição do espaço destinadas a organizar dados num espaço k-dimensional-
- **Para que serve?**
  - Pesquisas envolvendo uma chave multidimensional. Estimação de densidades de probabilidades multi dimensionais, etc...
- **Balanceamento**
  - Para que a pesquisa de um elemento seja feita em  $O(\log n)$  a árvore deverá estar balanceada para não existir grande desequilíbrio na altura da mesma

# KD-TREES

## INSERÇÃO DE NÓS

A inserção é feita ordenada pela dimensão da chave que vai alternando de nível para nível, como é mostrado o exemplo abaixo.



# KD-TREES

## DESEMPENHO COMPUTACIONAL

- **INSERÇÃO DE NÓS**

- No caso do presente trabalho, visto que a árvore deve estar balanceada, ou seja ambos os lados de cada nó devem ter o número de nós equilibrado, a o desempenho computacional é  $O(n \log n^2)$

- **REMOÇÃO DE NÓS**

- Na remoção de nós o desempenho é  $O(n \log n)$

- **PESQUISA DO NÓ MAIS PRÓXIMO**

- No tipo de pesquisa apresentado o desempenho é  $O(n)$

# KD-TREES

## COMO UTILIZAR

### – CRIAR UMA ÁRVORE:

- `KDTree(tamanho, dimensões)`  
exemplo: `arvore = KDTree(1000, 3)`

### – INSERIR UM NÓ:

- `arvore.malloc(No(chave, valor))`  
exemplo:           `no = No((0,0,1), "Palavra")`  
                  `arvore.malloc(no)`

### – ELIMINAR UM NÓ:

- `arvore.freeNo(no)`

### – PESQUISAR O ELEMENTO MAIS PRÓXIMO:

- `mais_proximo = arvore.maisProximo(no)`

# BLOCOS

## CONCLUSÃO

- O trabalho deveria ter sido elaborado com base no tipo Adaptive Kd-Tree, pois julgo que seria mais simples efectuar as operações pretendidas, mas tal não foi possível dentro de tempo útil. Fica essa importante alteração para trabalho futuro.
- A parte da pesquisa do elemento mais próximo também pode ser bastante melhorada baseando-se na pesquisa em profundidade e em largura, tirando partido da alteração para uma Adaptive Kd-Tree. Nesta parte assumi que a pesquisa é feita pela distância do ponto no espaço a partir das suas coordenadas e não a distância dentro da árvore.
- Os gráficos também precisam ser melhorados, efectuando testes com valores mais elevados e obtendo a linha do gráfico uniforme, utilizando por exemplo o método dos mínimos quadrados.
- Outra parte a melhorar neste trabalho é o relatório, que apesar de ter sido feito em latex, não aproveita grande parte das suas potencialidades, como por exemplo a bibliografia.

**OBRIGADO**

**QUESTÕES?**

- Pedro Moreira, 10015 – [mail@pedromoreira.org](mailto:mail@pedromoreira.org)
- Estruturas de Dados e Algoritmos – 2º Ano Engenharia Informática
- Escola Superior de Tecnologia e Gestão de Beja
- Docente: José Jasnau Caeiro