PRINCIPLES OF DATA ORGANISATION

Single Dimension-Based Indexing

MOTIVATION

- How to search effectively in more than one dimension?
  - Example: map, people, houses – find restaurants in a given proximity from a person
- Objects with coordinates (position) and extent
- How to represent spatial object in the database?
  - Reduction to 1 dimenion
  - Use full spatial data

One-dimensional embedding of spatial objects

- Data files are linear (1-D), which is not natural for spatial data (n-D)
- We would like to cluster together data to maintain locality
  - Data close to each other in nD are close to each other also in 1D
- We assume a space representable as a grid
  - Every object is in a particular position on the grid
  - Sufficiently granular

SPACE FILLING CURVES

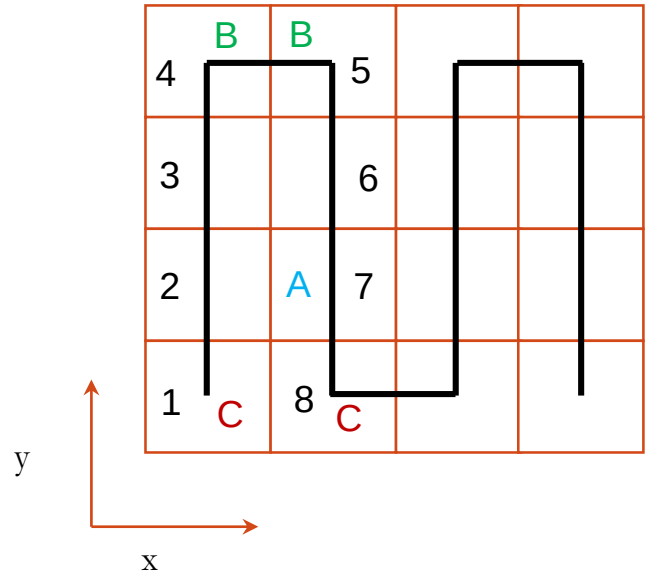
For now we are going to focus on 2D

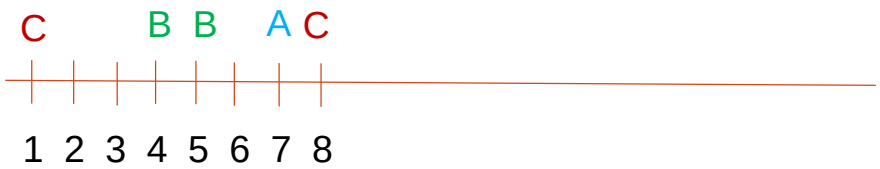
- A curve visiting cells of the grid representing the space; each cell is visited exactly once
- The points on the line are ordered thus giving the points in the space (grid) linear ordering
  - 2D => 1D
- With space filling curves one can implement file operations similarly to standard ordered files
- The goal is to find mapping preserving locality as best as possible

NAIVE REPRESENTATION

2D representation is projected into 1D so that points with smaller ***x*** coordinates precede those with larger ***x*** coordinates

- Good locality on the ***y*** axis

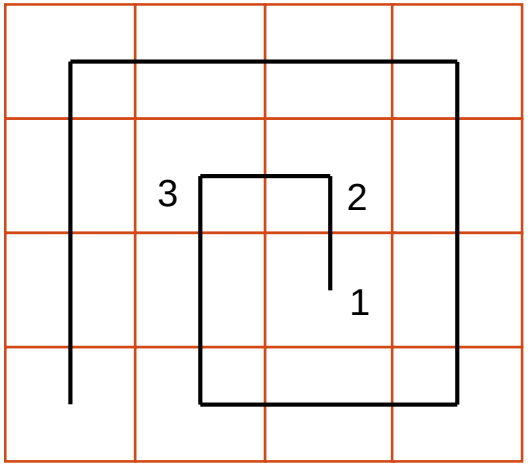




SPIRAL REPRESENTATION

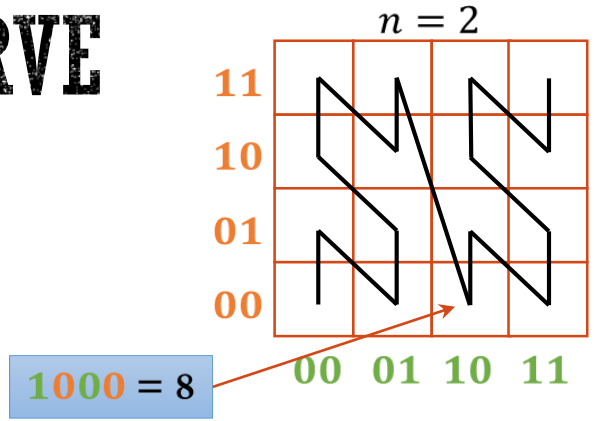
Favors objects in the middle of the grid but is not suitable for "moderately" off-centered objects

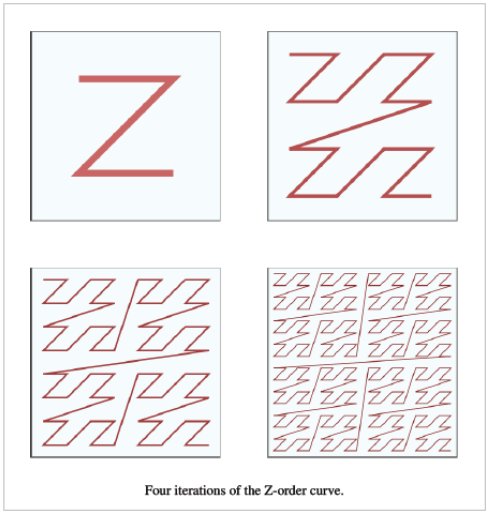
- Good locality in the center

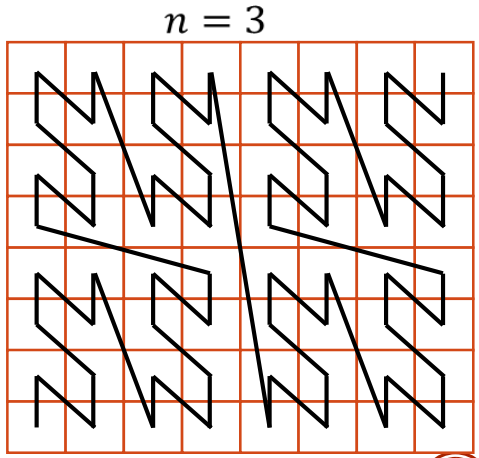


Z-CURVE/Z-ORDERING/PEAN CURVE

- Recursive representation, connecting points by z-order
- Address is formed by interleaving the bits in bit representations of x and y coordinates



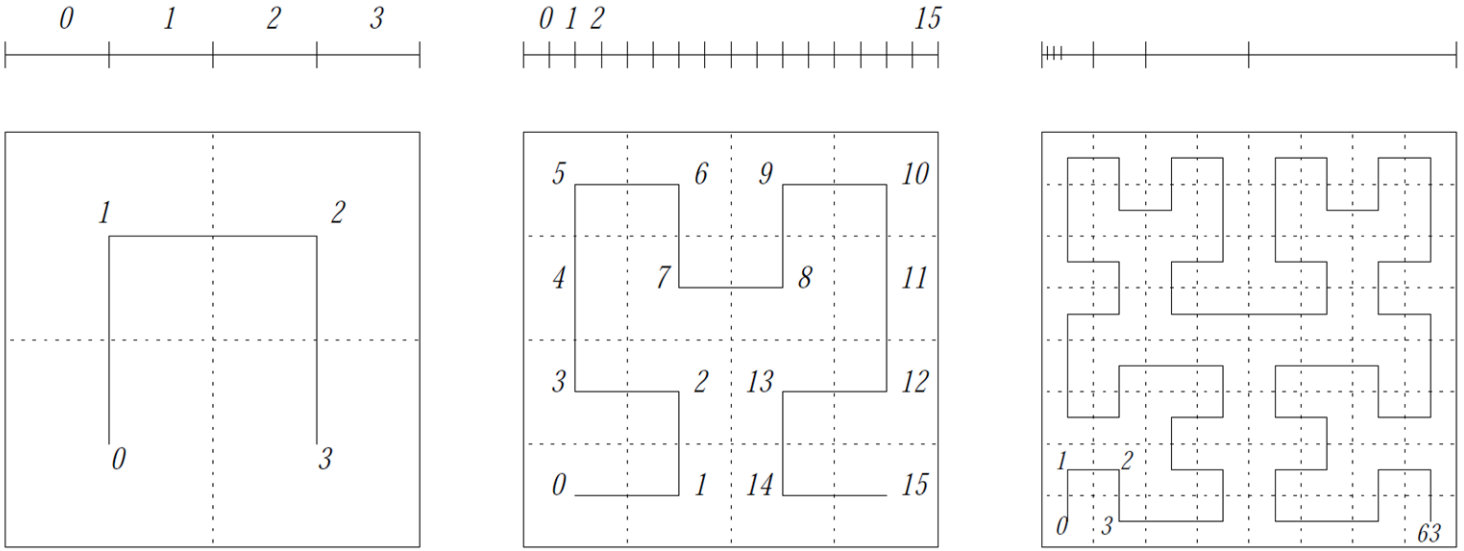

Four iterations of the Z-order curve.



https://en.wikipedia.org/wiki/Hilbert_curve#/media/File:Hilbert-curve_rounded-gradient-animated.gif

HILBERT CURVE

- Space is divided into four parts and their ordering is given by the "cup"-like curve
- Every square is divided into another four parts using another cup-like curve which needs to be rotated so that neighboring squares in higher level ordering are connected
- Most neighbors are preserved (but not all – e.g., the first and last)
  - Same for the z-curve



SINGLE DIMENSION-BASED INDEXING

- We can use B+ trees even without a space filling curve
- B+ trees are capable of storing multi-dimensional information in a form of an ordered tuple – compound (chained) search keys
  - The tuples are ordered first based on the first element, then on the second and so on (lexicographical order)
- The standard ordering of tuples in a B+ tree resembles naïve space-filling curve
- The way in which we define ordering on the tuples defines the type of space-filling curve

SINGLE DIMENSION-BASED INDEXING : EXAMPLE