11de8e -> b2a1ab

- Rename Method Ctrl+R
- Replace Magic Number with Symbolic Constant
- Dead Code

b2a1ab -> ffa4c9

- Extract Class
- Move Method Ctrl+M (Right -> Refactor -> Move)
- Extract Method Alt + Shift + M (Right -> Introduce -> Method)
- Introduce Parameter Object
- Replace Error Code with Exception

- create ParsedArgs (Extract Class)
- create InvalidInputException - for now empty (Replace Error Code with Exception)
- move methods to ParsedArgs (Move Method)
- first extract method to move it - isValidNumberOfArgs (Extract Method)
- same with pixel size
- pass whole arr to ParsedArgs at once
- ParsedArgs can be used as parameter itself - getters (Introduce Parameter Object)
    - or extract needed fields in method calls
- throw exception on first error - no true/false, throw immediately (Introduce Parameter Object)
- aggregate errors (just show idea, git checkout to actual)

ffa4c9 -> 85d9e8

- Extract Interface

- compose methods in main
    - colorize, prepareOutput, outputAscii, outputPng
- ass size() to FractalGrid to be used in colorize()
- AsciiImageConv extract interface colorFor() - only signature, regex from ctor
- Extract Class to hold current implementation

85d9e8 -> 8c6c69

- Replace Data Value with Object
- Primitive Obsession

- define the required interface - get x,y set x,y, wid, hei
- Matrix, ArrayBackedMatrix
- use current code for implementation - arrayBacked
- replace all occurences by Matrix -ImgConv, Ascii, Png, FractalGrid, RegexFract
- extract method symbolCount(rows, cols, pix) in Ascii

8c6c69 -> cf9506

- Refused Bequest
- Add Parameter
- Extract Superclass
- Pull Up Method
- Form Template Method

- need to know if dealing with Ascii or Png
- Png should not extend Ascii - validPixelSize was already checked, no need to do again
- ImageConverter is not abstract enough - we need to hack with side effects (Refused Bequest)
- refactor to also take the output file (Add Parameter)
    - could be even more abstract (-> OutputStream)
- extract AbstractImageConverter (Extract Superclass)
- extract methods in subclasses, then pull up (Pull Up Method)
    - convertStart, rowStart, writePixel, rowEnd, saveResult
- StringBuilder gridBuff and rowBuff to fields

cf9506 -> ac67d1

- Switch Statements
- Replace Type Code with Class
- Replace Type Code with State/Strategy

- create new Quadrant class
    - for now no behavior, only defined set of well named constants (Replace Type Code with Class)
- move addend methods to Quadrant (Replace Type Code with State/Strategy)

ac67d1 -> 242e26

- Temporary Field
- Replace Type Code with Subclasses
- Replace Constructor with Factory Method
- Push Down Method

- pass parent as parameter instead of individual fields
    - size has to be made into field
- extract method createChildren()
- children field not always used, calculate on demand
- create two subclasses based on isLeaf (Replace Type Code with Subclasses)
    - FragmentLeaf, FragmentSplittable
- move constructors to respective subclasses, keep isField
- push down writeChildrenTo(), child creation (not used by leaf), remove isField
- split Splittable into Intermediate and Root