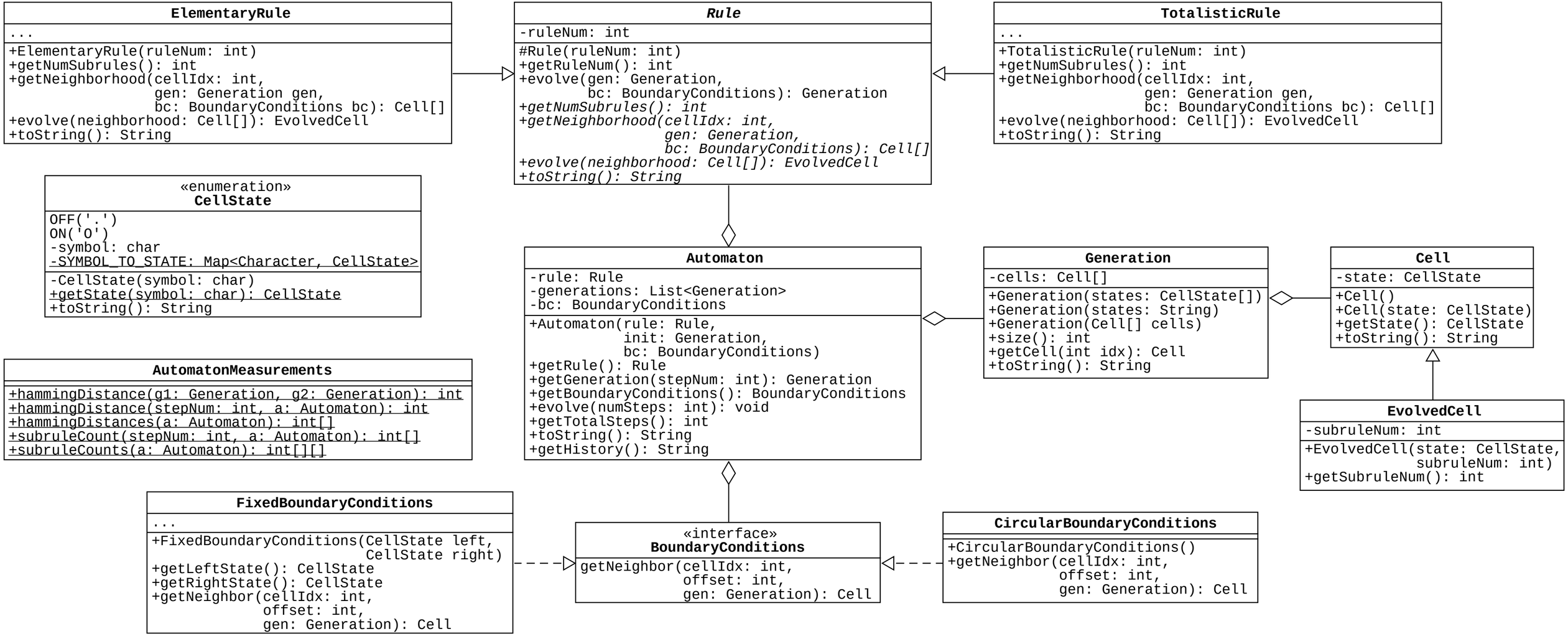
ElementaryRule
...
+ElementaryRule(ruleNum: int)
+getNumSubrules(): int
+getNeighborhood(cellIdx: int,
gen: Generation gen,
bc: BoundaryConditions bc): Cell[]
+evolve(neighborhood: Cell[]): EvolvedCell
+toString(): String
Rule
-ruleNum: int
#Rule(ruleNum: int)
+getRuleNum(): int
+evolve(gen: Generation,
bc: BoundaryConditions): Generation
+getNumSubrules(): int
+getNeighborhood(cellIdx: int,
gen: Generation,
bc: BoundaryConditions): Cell[]
+evolve(neighborhood: Cell[]): EvolvedCell
+toString(): String
TotalisticRule
...
+TotalisticRule(ruleNum: int)
+getNumSubrules(): int
+getNeighborhood(cellIdx: int,
gen: Generation gen,
bc: BoundaryConditions bc): Cell[]
+evolve(neighborhood: Cell[]): EvolvedCell
+toString(): String
«enumeration»
CellState
OFF('.')
ON('O')
-symbol: char
-SYMBOL_TO_STATE: Map<Character, CellState>
-CellState(symbol: char)
+getState(symbol: char): CellState
+toString(): String
Automaton
-rule: Rule
-generations: List<Generation>
-bc: BoundaryConditions
+Automaton(rule: Rule,
init: Generation,
bc: BoundaryConditions)
+getRule(): Rule
+getGeneration(stepNum: int): Generation
+getBoundaryConditions(): BoundaryConditions
+evolve(numSteps: int): void
+getTotalSteps(): int
+toString(): String
+getHistory(): String
Generation
-cells: Cell[]
+Generation(states: CellState[])
+Generation(states: String)
+Generation(Cell[] cells)
+size(): int
+getCell(int idx): Cell
+toString(): String
Cell
-state: CellState
+Cell()
+Cell(state: CellState)
+getState(): CellState
+toString(): String
EvolvedCell
-subruleNum: int
+EvolvedCell(state: CellState,
subruleNum: int)
+getSubruleNum(): int
AutomatonMeasurements
+hammingDistance(g1: Generation, g2: Generation): int
+hammingDistance(stepNum: int, a: Automaton): int
+hammingDistances(a: Automaton): int[]
+subruleCount(stepNum: int, a: Automaton): int[]
+subruleCounts(a: Automaton): int[][]
FixedBoundaryConditions
...
+FixedBoundaryConditions(CellState left,
CellState right)
+getLeftState(): CellState
+getRightState(): CellState
+getNeighbor(cellIdx: int,
offset: int,
gen: Generation): Cell
«interface»
BoundaryConditions
getNeighbor(cellIdx: int,
offset: int,
gen: Generation): Cell
CircularBoundaryConditions
+CircularBoundaryConditions()
+getNeighbor(cellIdx: int,
offset: int,
gen: Generation): Cell