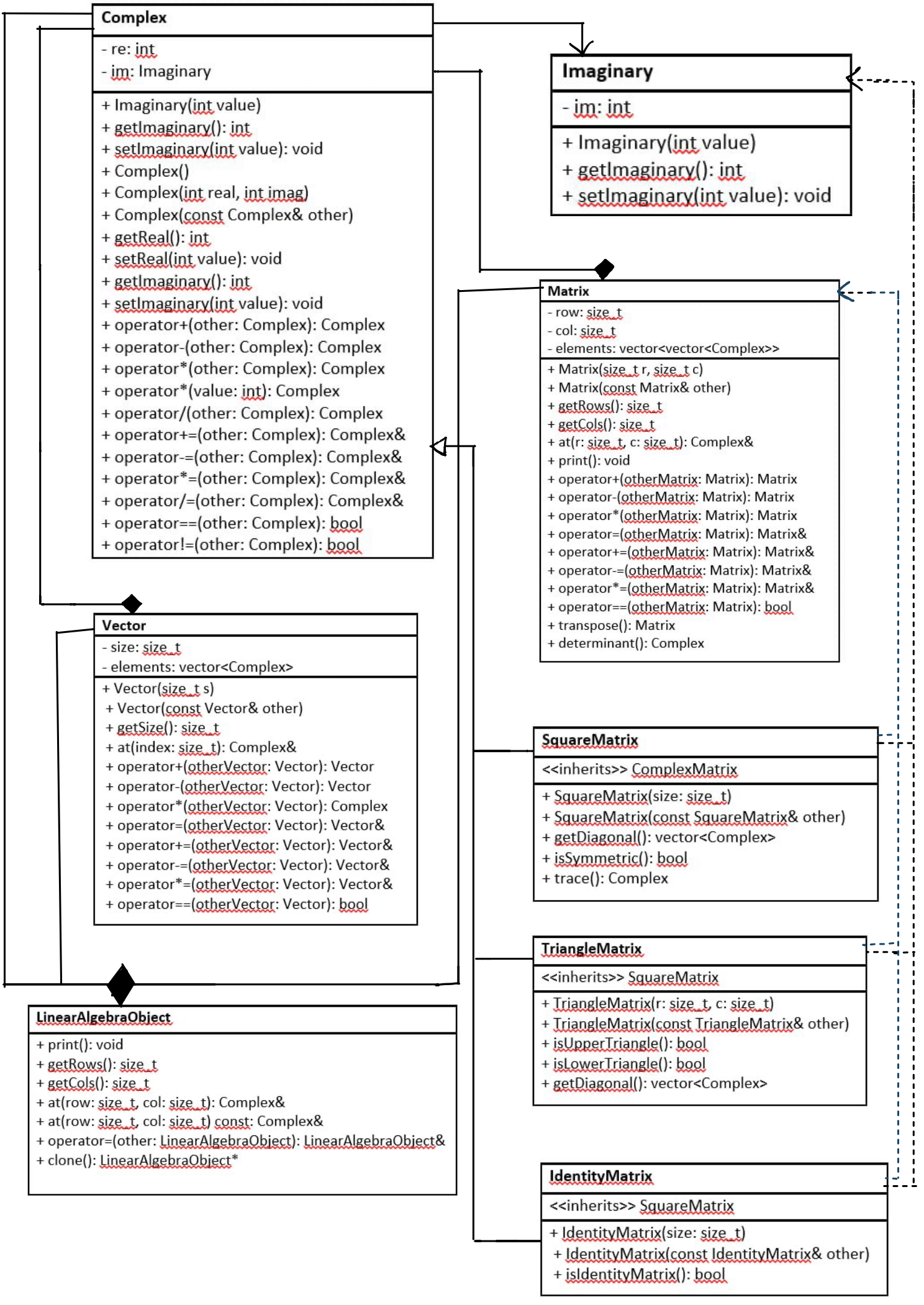
Complex
- re: int
- im: Imaginary
+ Imaginary(int value)
+ getImaginary(): int
+ setImaginary(int value): void
+ Complex()
+ Complex(int real, int imag)
+ Complex(const Complex& other)
+ getReal(): int
+ setReal(int value): void
+ getImaginary(): int
+ setImaginary(int value): void
+ operator+(other: Complex): Complex
+ operator-(other: Complex): Complex
+ operator*(other: Complex): Complex
+ operator*(value: int): Complex
+ operator/(other: Complex): Complex
+ operator+=(other: Complex): Complex&
+ operator-=(other: Complex): Complex&
+ operator*=(other: Complex): Complex&
+ operator/=(other: Complex): Complex&
+ operator==(other: Complex): bool
+ operator!=(other: Complex): bool
Imaginary
- im: int
+ Imaginary(int value)
+ getImaginary(): int
+ setImaginary(int value): void
Matrix
- row: size_t
- col: size_t
- elements: vector<vector<Complex>>
+ Matrix(size_t r, size_t c)
+ Matrix(const Matrix& other)
+ getRows(): size_t
+ getCols(): size_t
+ at(r: size_t, c: size_t): Complex&
+ print(): void
+ operator+(otherMatrix: Matrix): Matrix
+ operator-(otherMatrix: Matrix): Matrix
+ operator*(otherMatrix: Matrix): Matrix
+ operator=(otherMatrix: Matrix): Matrix&
+ operator+=(otherMatrix: Matrix): Matrix&
+ operator-=(otherMatrix: Matrix): Matrix&
+ operator*=(otherMatrix: Matrix): Matrix&
+ operator==(otherMatrix: Matrix): bool
+ transpose(): Matrix
+ determinant(): Complex
Vector
- size: size_t
- elements: vector<Complex>
+ Vector(size_t s)
+ Vector(const Vector& other)
+ getSize(): size_t
+ at(index: size_t): Complex&
+ operator+(otherVector: Vector): Vector
+ operator-(otherVector: Vector): Vector
+ operator*(otherVector: Vector): Complex
+ operator=(otherVector: Vector): Vector&
+ operator+=(otherVector: Vector): Vector&
+ operator-=(otherVector: Vector): Vector&
+ operator*=(otherVector: Vector): Vector&
+ operator==(otherVector: Vector): bool
SquareMatrix
<<inherits>> ComplexMatrix
+ SquareMatrix(size: size_t)
+ SquareMatrix(const SquareMatrix& other)
+ getDiagonal(): vector<Complex>
+ isSymmetric(): bool
+ trace(): Complex
TriangleMatrix
<<inherits>> SquareMatrix
+ TriangleMatrix(r: size_t, c: size_t)
+ TriangleMatrix(const TriangleMatrix& other)
+ isUpperTriangle(): bool
+ isLowerTriangle(): bool
+ getDiagonal(): vector<Complex>
IdentityMatrix
<<inherits>> SquareMatrix
+ IdentityMatrix(size: size_t)
+ IdentityMatrix(const IdentityMatrix& other)
+ isIdentityMatrix(): bool
LinearAlgebraObject
+ print(): void
+ getRows(): size_t
+ getCols(): size_t
+ at(row: size_t, col: size_t): Complex&
+ at(row: size_t, col: size_t) const: Complex&
+ operator=(other: LinearAlgebraObject): LinearAlgebraObject&
+ clone(): LinearAlgebraObject*