Marco Martinez
CISP 430
4/10/19
Assignment 3

CLASS DIAGRAM

Classes
> Main
> Parser
> ParseTable
> Evaluator
> GenericSlot
> GenericItemType
> ListEntry
> JList
> Bucket
> HashTable
> Queue
> Stack
> OperatorToken
> OperandToken

Associations
> OperatorToken(1) --- inherits --- (1)GenericSlot
> OperandToken(1) --- inherits --- (1)GenericSlot
> GenericSlot(1) --- inherits --- (1)GenericItemType
> ListEntry(1) --- includes --- (1)GenericItemType
> JList(1) --- contains --- (m)ListEntry
> Queue(1) --- inherits --- (1)JList
> Stack(1) --- inherits --- (1)JList
> Bucket(1) --- inherits --- (1)JList
> HashTable(1) --- contains --- (m)Bucket
> Parser(1) --- includes --- (1)ParseTable
> Parser(1) --- includes --- (1)Queue
> ParseTable(1) --- includes --- (1)Queue
> ParseTable(1) --- includes --- (1)Stack
> Evaluator(1) --- includes --- (1)Queue
> Evaluator(1) --- includes --- (1)OperandToken
> Evaluator(1) --- includes --- (1)HashTable
> Main(1) --- uses --- (1)Parser
> Main(1) --- uses --- (1)Evaluator
> Main(1) --- includes --- (1)Queue
> Main(1) --- includes --- (1)OperandToken

ListEntry Class Attributes
> INSTANCE VARIABLES
>> (-) GenericItemType data

(-) ListEntry next
(-) ListEntry prev

CLASS CONSTRUCTORS
(+) ListEntry()
(+) ListEntry(GenericDataItem)
(+) ListEntry(ListEntry)

CHANGE STATE SERVICES
(+) void setData(GenericItemType)
(+) void setNext(ListEntry)
(+) void setPrev(ListEntry)

READ STATE SERVICES
(+) GenericItemType getData()
(+) ListEntry getNext()
(+) ListEntry getPrev()

JList Class Attributes
INSTANCE VARIABLES
(-) ListEntry head
(-) ListEntry tail
(-) ListEntry currentIteration
(-) int totalCount
(-) int currentCount

CLASS CONSTRUCTORS
(+) JList()
(+) JList(GenericItemType)
(+) JList(ListEntry)
(+) JList(JList)
(+) JList(Stack)
(+) JList(Queue)
(+) JList(PriorityQueue)

CHANGE STATE SERVICES
(+) void init()
(+) void add_fromHead(GenericItemType)
(+) void add_fromHead(ListEntry)
(+) void add_fromMid(GenericItemType)
(+) void add_fromMid(ListEntry)
(+) void add_fromTail(GenericItemType)
(+) void add_fromTail(ListEntry)
(+) void bubbleSort_ascending()
(+) void bubbleSort_descending()
(+) GenericItemType linearSearch(GenericItemType)
(+) GenericItemType linearSearch(ListEntry)
(-) ListEntry lSearch(GenericItemType)
(+) void remove(GenericItemType)

(+) void remove(ListEntry)
(-) void delete(GenericItemType)
(+) void reverseList()

READ STATE SERVICES
      (+) boolean isFull()
      (+) boolean isEmpty()
      (+) int getCount()
      (+) ListEntry getStart()
      (+) ListEntry getEnd()
      (+) void Iterator_initialize()
      (+) boolean Iterator_hasNext()
      (+) GenericItemType Iterator_iterate()

Bucket Class Attributes

CLASS CONSTRUCTORS
      (+) Bucket()
      (+) Bucket(GenericItemType data)
      (+) Bucket(ListEntry le)
      (+) Bucket(JList l)
      (+) Bucket(Bucket b)

READ STATE SERVICES
      (+) int searchLocation(GenericSlot key)
      (+) int searchLocation(GenericItemType key)
      (+) int searchLocation(ListEntry key)
      (-) int keyLocation(GenericItemType key)

HashTable Class Attributes

CONSTANT DEFINITION
      (+) int MAXBUCKETS

INSTANCE VARIABLES
      (-) Bucket[] ht
      (-) int index

CLASS CONSTRUCTORS
      (+) HashTable()
      (+) HashTable(GenericSlot gs)
      (+) HashTable(HashTable ht)

CHANGE STATE SERVICES
      (+) void initialize()
      (+) void insertIntoHT(GenericSlot data)
      (+) GenericItemType searchHT(GenericSlot data)
      (+) void deleteFromHT(GenericSlot data)

READ STATE SERVICES
      (+) void Iterator_initialize()

(+) boolean Iterator_hasNext()
(+) Bucket Iterator_getNext()
(+) int findLocation(GenericSlot key)
(+) int getIndex()
(+) Bucket[] getHashTable()
(+) Bucket getHashTable(int index)
(+) int getMax()

Stack Class Attributes
CLASS CONSTRUCTORS
(+) Stack()
(+) Stack(GenericItemType)
(+) Stack(ListEntry)
(+) Stack(JList)
(+) Stack(Stack)
(+) Stack(Queue)
(+) Stack(PriorityQueue)

CHANGE STATE SERVICES
(+) void push(GenericItemType)
(+) GenericItemType pop()

READ STATE SERVICES
(+) GenericItemType showTop()

Queue Class Attributes
CLASS CONSTRUCTORS
(+) Queue()
(+) Queue(GenericItemType)
(+) Queue(ListEntry)
(+) Queue(JList)
(+) Queue(Stack)
(+) Queue(Queue)
(+) Queue(PriorityQueue)

CHANGE STATE SERVICES
(+) void enQueue(GenericItemType)
(+) GenericItemType deQueue()

Parser Class Attributes
INSTANCE VARIABLE DECLARATION
(-) String infix
(-) Queue postfix
(-) ParseTable intermediate
(-) String[][] nextstateTable

CLASS CONSTRUCTORS
(+) Parser()
(+) Parser(String newInfix)

(+) Parser(String newInfix, String[][] table)
(+) Parser(Parser copy)

CHANGE STATE SERVICES
    (+) void assignNextstateTable(String[][] table)
    (+) void translateInfixToPostfix()
    (+) void doAction(String action, String temp, int currentCol)

READ STATE SERVICES
    (+) int determineRow()
    (+) String[][] getNextstateTable()
    (+) String getInfix()
    (+) Queue getPostfix()
    (+) int getInfixLength()
    (+) ParseTable getIntermediate()

ParseTable Class Attributes

INSTANCE VARIABLE DECLARATION
    (-) Queue Q1
    (-) Stack S2

CLASS CONSTRUCTORS
    (+) ParseTable()
    (+) ParseTable(Queue newQueue)
    (+) ParseTable(ParseTable copy)

CHANGE STATE SERVICES
    (+) void doQ1(GenericSlot token)
    (+) void doS2(GenericSlot token)
    (+) void doER()
    (+) void doU1()
    (+) void doUC()
    (+) void doU2()

READ STATE SERVICES
    (+) GenericItemType deQueueFromQ1()
    (+) GenericItemType showTopOfS2()
    (+) int getCountOfS2()
    (+) int getCountOfQ1()
    (+) String getContents()
    (+) Queue getQ1()
    (+) Stack getS2()

Evaluator Class Attributes

CONSTANT DEFINITIONS
    (-) int BUCKET_MAX

INSTANCE VARIABLE DECLARATIONS
    (-) Queue postfix

(-) OperandToken result
(-) HashTable symboltable

CLASS CONSTRUCTORS
(+) Evaluator()
(+) Evaluator(Queue newPostfix)
(+) Evaluator(Queue newPostfix, HashTable newSymboltable)
(+) Evaluator(Evaluator copy)

CHANGE STATE SERVICES
(+) void addSymbolToTable(OperandToken symbol)
(+) OperandToken determineValue(OperandToken token)
(+) void evaluatePostfix()
(+) double determineOperation(double a, double b, String operator)

READ STATE SERVICES
(+) Queue getPostfix()
(+) OperandToken getResult()
(+) HashTable getSymboltable()

OperatorToken Class Attributes
CONSTANT DEFINITONS
(-) int PRIO_MAX

INSTANCE VARIABLE DECLARATIONS
(-) String token
(-) int prio

CLASS CONSTRUCTORS
(+) OperatorToken()
(+) OperatorToken(String newToken)
(+) OperatorToken(OperatorToken copy)

CHANGE STATE SERVICES
(+) void setToken(String newToken)
(+) void setPrio(int newPrio)

READ STATE SERVICES
(+) boolean isLess(GenericItemType git)
(+) boolean isEqual(GenericItemType git)
(+) boolean isGreater(GenericItemType git)
(+) int determineIndex()
(+) String getToken()
(+) int getPrio()
(+) String toString()

OperandToken Class Attributes
INSTANCE VARIABLE DECLARATIONS
(-) String token

(-) double value

CLASS CONSTRUCTORS
      (+) OperandToken()
      (+) OperandToken(String newToken)
      (+) OperandToken(String newToken, double newValue)
      (+) OperandToken(String newToken, float newValue)
      (+) OperandToken(String newToken, int newValue)
      (+) OperandToken(OperandToken copy)

CHANGE STATE SERVICES
      (+) void setToken(String newToken)
      (+) void setValue(int newValue)
      (+) void setValue(float newValue)
      (+) void setValue(double newValue)

READ STATE SERVICES
      (+) boolean isLess(GenericItemType git)
      (+) boolean isEqual(GenericItemType git)
      (+) boolean isGreater(GenericItemType git)
      (+) int determineIndex()
      (+) String getToken()
      (+) double getValue()
      (+) String toString()

PSEUDOCODE (Main)

declare Scanner scan as new Scanner(System.in)
declare Parser myParser
declare Evaluator myEval
declare String infix
declare String yesNo as "y"
declare Queue postfix
declare OperandToken result;

while((yesNo.toLowerCase()).equals("y")) {
    set infix as getInfix(scan);
    set myParser as new Parser(infix, generateNextstateTable());
    call myParser.translateInfixToPostfix();
    set postfix as myParser.getPostfix();
    set myEval as new Evaluator(postfix, generateSymboltable());

    call determineAdditionalVariables(myEval, scan);
    call myEval.evaluatePostfix();
    set result as myEval.getResult();

```
        call displayResults(result);
        set yesNo as getYesNo(scan);
}
```