AUTOMATA THEORETIC MODEL CHECKING OF WEB SERVICE SYSTEMS

Class Date
Class used to specify the day, month and year which specify a complete date.
Attributes: Day, Month, Year

Class ServiceInput
Class used to specify the properties of the web service.
Attributes: Type, PropertyName, Value

Class Service
Class used to specify the web service in its simplest form.
Attributes: PropertyList, ServiceName, PropertyCount

Class State
Class used to specify the states of the service automaton.
Attributes: ServiceName, StateName, StateNumber, isStart, isFinish

Class TransitionCondition
Class used to specify the constraints of the service automaton.
Attributes: ConditionName, ConditionResult, Value, DecidingProperty

Class Transition
Class used to specify the transitions of the service automaton.
Attributes: ServiceName, InputState, OutputState, Condition

Class ServiceAutomata
Class used to represent the service as an automaton.
Attributes – StatesList, TransitionsList, Service, StateCount, TransitionCount

Function ConstructAutomata
Function used for constructing the service automaton.
Inputs: File
Outputs: Service Automaton

Class LTLOperator
Class used to represent the linear temporal logic operators.
Attributes: OperatorSymbol, Arity, Variables

Class LTLExpression
Class used to specify the linear temporal logic formulae.
Attributes – VariablesList, OperatorsList, Formula, VariableCount, OperatorCount

Function BuildExpression
Function used to build the LTL formulae using tree construction.
Inputs: LTLOperators, LTLVariables
Outputs: LTLFormula

Class GraphNode
Class used to represent the state of the formula automaton.
Attributes: Name, Father, Incoming, New, Old, Next, Accepting

Function SetEquality
Function to check whether two formula sets are equal.
Inputs: Formula Sets
Outputs: Boolean

Function ExpressionUnion
Function to find the Union of two formula sets.
Inputs: Formula Sets
Output: Union Formula Set

Function SetComplement
Function to find the complement of a set with respect to another.
Inputs: Formula Sets
Outputs: Set Complement

Function Neg
Function to find the negation of a LTL formula.
Inputs: Formula
Output: Negation Formula

Function expand
Recursive function to expand the formula graph.
Inputs: Node, NodeSet
Output: NodeSet

Function CreateGraph
Function to create the state graph of the formula.
Input: Formula
Output: State Graph

Class FormulaSet
Class representing a set of formulas.
Attributes: Formulas, FormulaCount

Function CreateTrueSet
Function to create set of true formulas.
Inputs: PresentSet, OldSet, Formula
Outputs: TrueFormulasSet

Class GraphTransition
Class to represent the transitions of the formula automaton.
Attributes: IncomingNode, OutgoingNode, TrueFormulas

Class FormulaAutomata
Class to represent the formula as an automaton.
Attributes: StatesSet, TransitionsSet

Class BuchiStates
Class representing states of the product automaton.
Attributes: StateName, Accepting

Class BuchiTransitionSets
Class representing true formula sets of the product automaton.
Attributes: FormulaSet, FormulaCount

Class BuchiTransitions
Class representing the transitions of the product automaton.
Attributes: TrueSets, SetCount, InputState, OutputState

Class BuchiAutomaton
Class representing the product of service and formula automaton.
Attributes: StatesSet, TransitionsSet, StateCount, TransitionCount

Function ServiceToBuchi
Function converting service automaton to Buchi.
Inputs: ServiceAutomaton
Outputs: ServiceBuchiAutomaton

Function FormulaToBuchi
Function converting formula automaton to Buchi.
Inputs: FormulaAutomaton,
Outputs: FormulaBuchiAutomaton

Function ProductAutomaton
Function to generate product automaton from service and formula.
Inputs: ServiceBuchiAutomaton, FormulaBuchiAutomaton
Outputs: ProductBuchiAutomaton

Class AdjacencyList
Class containing all the adjacent nodes to a particular node in the graph.
Attributes: AdjacentStates, StateName

Function CycleCheck
Function to perform Depth First Search to check for cycles in directed graph.
Inputs: ProductAutomaton
Outputs: AcceptingCycles