step(currentBsState)

```
$currentContext \leftarrow value[currentBsState]$
**for each** state in states[currentContext] **do**
    $tran \leftarrow trOut[state]$
    **while** $tran \neq NIL$ **do**
        **if** $isEventPresent(currentContext, tran)$ **and** $isBufferFree(currentContext, tran)$ **then**
            $newContext \leftarrow createNewContext(currentContex, tran)$
            $newTransition \leftarrow createNewTransition(currentContext, tran)$
            $item \leftarrow contextSearch(newContext, ctHashMap)$
            **if** $item \neq NIL$ **then**
                $destinationBsState \leftarrow subValue[item]$
                $dest[newTransition] \leftarrow destinationBsState$
            **else**
                $createNewState(newContext, tran)$
                $step(destinationBsState)$
            **end if**
        **end if**
    **end while**
**end for**
```

takeEventFromBuffer(context,action)

```
$l \leftarrow link[action]$
$pos \leftarrow index[l]$
**return** $buffer[context][pos]$
```

isEventPresent(context,transition)

```
$actionRequest \leftarrow actIn[transition]$
$eventBuffer \leftarrow takeEventFromBuffer(context, actionRequest)$
$eventRequest \leftarrow event[actionRequest]$
**return** $(actionRequest = NIL$ **or** $eventBuffer = eventRequest)$
```

isBufferFree(context,transition)

```
$actionProduced \leftarrow actOut[transition]$
**while** $actionProduced \neq NIL$ **do**
    $eventBuffer \leftarrow takeEventFromBuffer(context, actionProduced)$
    **if** $eventBuffer \neq NIL$ **then**
        **return** $FALSE$
    **end if**
    $actionProduced \leftarrow next[actionProduced]$
**end while**
**return** $TRUE$
```

createNewContext(context,transition)

```
1:  $newContext \leftarrow initializeContext()$
2:  $state \leftarrow dest[transition]$
3:  $actionRequest \leftarrow actIn[transition]$
4:  $eventRequest \leftarrow event[actionRequest]$
5:  if $eventRequest \neq NIL$ then
6:     $eventBuffer \leftarrow NIL$
7:  end if
8:  $actionProduced \leftarrow actOut[tran]$
9:  while $actionProduced \neq NIL$ do
10:    $l2 \leftarrow link[actionProduced]$
11:    $pos2 \leftarrow index[l2]$
12:    $buffer[newContext][pos2] \leftarrow actionProduced$
13:    $actionProduced \leftarrow next[actionProduced]$
14: end while
15: return $newContext$
```

createNewTransition(context,transition,currentBsState)

```
1:  $newTransition \leftarrow initializeTransition()$
2:  $obs[newTransition] \leftarrow obs[transition]$
3:  $rel[newTransition] \leftarrow rel[transition]$
4:  $src[newTransition] \leftarrow currentBsState$
5:  $addTransition(newTransition)$
6:  return newTransition
```

createNewState(context,transition)

```
1:  $destinationBsState \leftarrow initializeState()$
2:  $value[destinationBsState] \leftarrow context$
3:  if $isFinal(context)$ then
4:     $final[destinationBsState] \leftarrow TRUE$
5:  else
6:     $finale[destinationBsState] \leftarrow FALSE$
7:  end if
8:  $addState(destinationBsState)$
9:  $dest[transition] \leftarrow destinationBsState$
10: $addContextToHashMap(context)$
```

dfs(state)

```
1:  $color[state] \leftarrow GRAY$
2:  $transitionsIncoming \leftarrow trIn[state]$
3:  while $transitionsIncoming \neq NIL$ do
4:     $stateSource \leftarrow scr[transitionsIncoming]$
5:     if $color[stateSource] = WHITE$ then
6:        dfs[stateSource]
7:     end if
8:     $transitionsIncoming \leftarrow next[transitionsIncoming]$
9:  end while
10: $color[state] \leftarrow BLACK$
```

prune(autom)

```
$totalState \leftarrow states[autom]$
while $totalState \neq NIL$ do
   if $final[totalState] = TRUE$ then
      $dfsVisit(totalState)$
   end if
   $totalState \leftarrow next[totalState]$
end while
while $totalStat \neq NIL$ do
   if $color[totalState] = WHITE$ then
      $removeTheState(autom, totalState)$
   end if
   $totalState \leftarrow next[totalState]$
end while
```

isTransitionObservable(context,transition)

```
$label \leftarrow NIL$
$currentObservation \leftarrow currentObs[context]$
{Controllo sulla presenza della lista di osservazioni}
if $currentObservation \neq NIL$ then
   $label \leftarrow currentObervation$
   $transitionLabel \leftarrow obs[tansition]$
   $idObsarvation \leftarrow id[transitionLabel]$
end if
if $transitionLabel \neq NIL$ and ($label \neq NIL$ or $idLabel \neq idObsarvation$) then
   return FALSE
else
   return TRUE
end if
```