SuffixTree

index
/home/merybria/Scrivania/unisa magistrale/daa/midterm/codice/SuffixTree.py

Classes

struttureImpiegate.Tree.Tree(builtins.object)

SuffixTree

class **SuffixTree**(struttureImpiegate.Tree.Tree)

```
SuffixTree(stringSet)

class representing the suffix tree
```

Method resolution order:
SuffixTree
struttureImpiegate.Tree.Tree
builtins.object

Methods defined here:

__init__(self, stringSet)
```
constructor method for the suffix tree
:param stringSet: a set of strings from which the suffix tree should be built
```

__len__(self)
```
Return the total number of elements in the alberi.
```

child(self, position, s)
```
the function is used to access to the child position of the specified position whose substring
-starts with the string s
-is a prefix for the string s
:param position(SuffixTree.Position): the position where the function is called
:param s(str): the string
:returns the child node if it exists or None if it doesn't
```

getNodeDepth(self, position)
```
function that calculates the length of the string until that node
:param position: the position whose depth is desired
:return: the length of the substring
```

getNodeLabel(self, position)
```
function that gets the string corresponding to a certain position
:param position: position whose string is desired
:return: the corresponding string
```

getNodeMark(self, position)
```
function that accesses to the mark of the node
:param position: position whose mark is desired
:return: the list of the strings tho which the substring stored in that node belongs to
```

parent(self, p)
```
:returns the parent position of a generic position p
```

pathString(self, p)
```
function used to return the path string from the root to the specified position
:param p: position whose path string is being calculated
:return: the path string from the root to the position p
```

root(self)
```
:returns the root position in the tree
```

Data and other attributes defined here:

Position = <class 'SuffixTree.SuffixTree.Position'>
```
abstraction used to contain the reference to the tree and the node, whic contains information about the substring stored
```

Methods inherited from struttureImpiegate.Tree.Tree:

__iter__(self)
```
Generate an iteration of the alberi's elements.
```

children(self, p)
```
Generate an iteration of Positions representing p's children.
```

depth(self, p)
```
Return the number of levels separating Position p from the root.
```

height(self, p=None)

Return the height of the subtree rooted at Position p.

If p is None, return the height of the entire alberi.

is_empty(self)
Return True if the alberi is empty.

is_leaf(self, p)
Return True if Position p does not have any children.

is_root(self, p)
Return True if Position p represents the root of the alberi.

num_children(self, p)
Return the number of children that Position p has.

positions(self)
Generate an iteration of the alberi's positions.

postorder(self)
Generate a postorder iteration of positions in the alberi.

preorder(self)
Generate a preorder iteration of positions in the alberi.

Data descriptors inherited from struttureImpiegate.Tree.Tree:

__dict__
dictionary for instance variables (if defined)

__weakref__
list of weak references to the object (if defined)