ATLAS.ti Report

XOP\_Encoding\_Proj

Codes

Report created by dyg on Aug 9, 2017

# ● ->

Groups:

icon Modifiers

Comment:

Subscoping operator. This means that scopes are nested. For example, if you have a code at position x that reads “Constituent, Description” and then a paragraph/line/s later, at position y, you have “->, Complexity” then you are no denoting teh complexity of the consituent being discussed. This is opposed to having just “Complexity” at position y which would denote the complexity of whatever the previous structuring move was.

# ● <-

Groups:

icon Modifiers

Comment:

Subscoping lifting operator. This operator ends subscoping context, returning the scope to whatever the prior scope was. For example if you had “Operation” then a few lines down you have “->” “Complexity” then the scope would be talking about Operation Complexity. If after a few more lines you had “<-“ “Observation” Then you are now making a general observation about the Operation, not the Operation complexity. This is in contrast to just “Observation” which, in this case, would then be making an Observation of the Operation Complexity

# ● Abstraction

Groups:

icon Actions

Comment:

This code denotes that the document is abstracting or generalizing that which the context is set to. Ex: “Now we can generalize this problem to the shortest-path problem"

# ● ADT

Groups:

icon Concepts

# ● Advantages

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes the pros, the upside or the advantages to that which the context is set to.

# ● Algorithm

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes that the text is discussing an algorithm in general. This could be the same as the thesis topic, it could be a related algorithm (and have a “Related” modifier) etc.

# ● Application

Groups:

icon Elements of Explanatory Structure

Comment:

This scoping tag denotes that the text is now discussing the Use cases, or applications of the current context. Defaults to the Thesis Topic.

# ● Aside

Groups:

icon Modifiers

Comment:

This is a special modifier code. It denotes that the input tags describe text that is not directly related to the scope the input tags refer to. Cases where this modifier is used without any accompanying tag are considered equivalent to the tag set “Aside” “Comment”. That is to say, if Aside occurs with no input, then the Comment tag is considered as default input

# ● Assumption

Groups:

icon Actions

Comment:

This code denotes that the text is giving the reader or telling the reader an assumption about the context

# ● balanced

Groups:

icon In Vivo Terms

# ● Base Case

Groups:

icon Actions

Comment:

This code denotes the text is giving an explicit base case in an inductive procedure in relation to the context

# ○ Big Picture

# ● Binary Search Trees

Groups:

icon In Vivo Terms

# ● Black Box

Groups:

icon Document Level Codes

# ● breadth first search

Groups:

icon In Vivo Terms

# ● Cartoon

Groups:

icon Content Expression

Comment:

The content is represented in a drawn or animated graphic

# ● Cases

Groups:

icon Actions

Comment:

This code denotes that the text is breaking down the context into chunks of information, or cases. Often used after a “Base Case” code.

# ● caveat

Groups:

icon Modifiers

Comment:

This modifier takes any number of tags of any type. It denotes that the purpose of the text at hand is to further clarify a point, provide extra detail, or point out a consequence of some aspect of that which is being discussed. Ex: In general, this process will require many rotations for an insert because...

# ● Class

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes the explicit discussion of a group, set or class of some thing.

# ● Code

Groups:

icon Content Expression

Comment:

The content is represented as a block of code from some programming language

# ● combine

Groups:

icon In Vivo Terms

# ● Comment

Groups:

icon Actions

Comment:

This code is a dummy code whose use is just to provide a binding for an “Aside”, “Caveat”, or “Meta” modifier

# ● comparison sorts

Groups:

icon In Vivo Terms

# ● completed vertices

Groups:

icon In Vivo Terms

# ● Complexity

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes that the text is discussing the computational complexity of whatever the context is set to.

# ● Conclusion

Groups:

icon Actions

Comment:

This code denotes that the text is making a conclusion about the context

# ● Condition

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes that the text is discussing a condition, that whatever the context was set to has, that must be satisfied.

# ● conquer

Groups:

icon In Vivo Terms

# ● Constituent

Groups:

icon Elements of Explanatory Structure

Comment:

This code can be seen as a more general form of “Operation”. This code denotes that the text is discussing some constituent part of whatever the context was set to. This is typically used to denote data structures required for an algorithm.

# ● constraints

Groups:

icon In Vivo Terms

# ● Contrast

Groups:

icon Actions

Comment:

This code denotes that the text is contrasting the context with something else. This code may no longer be viable because it expresses in a worse way what the combination of “Explanandoid” “Relation” modifiers express.

# ● Data Structure

Groups:

icon In Vivo Terms

# ● definition

Groups:

icon Actions

Comment:

This code denotes that the text is defining some term about the context.

# ● depth first search

Groups:

icon In Vivo Terms

# ● Derivation

Groups:

icon Actions

Comment:

This code denotes that the text is making a derivation about something in relation to the context

# ● Description

Groups:

icon Actions

Comment:

This code is the most general Action code. It denotes that the text is describing the context in some manner.

# ● Design

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes that the text is talking about the design, or design considerations of that which the context is set to.

# ● directed graph

Groups:

icon In Vivo Terms

# ● Disadvantages

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes that the text is discussing the downsides, the cons or shortfalls of whatever the context was set to.

# ● distance matrix

Groups:

icon In Vivo Terms

# ● divide

Groups:

icon In Vivo Terms

# ● divide and conquer

Groups:

icon Concepts  icon In Vivo Terms

# ● dynamic programming

Groups:

icon In Vivo Terms

# ● dynamic programming functional equation

Groups:

icon In Vivo Terms

# ● Erroneous

Groups:

icon Modifiers

Comment:

This code denotes that the purpose of the text, upon which the input tags represent, is to be incorrect, however that may be defined.

# ● Example

Groups:

icon Actions

Comment:

This code denotes that the text is giving, or providing an example in relation to the context

# ● Explanandoid

Groups:

icon Modifiers

Comment:

This modifier tags takes a scoping tag as input. If denotes that the scoping tag, typically “problem” is used to illustrate something that is close to the thesis topic, but not quite the same. This tag is often used with the “Related” modifier to denote the discussion of a related problem that is similar to the Thesis Topic but is not the thesis topic. Ex: Considering, discussing and explaining the shortest-path problem in relation to the motivating problem for dijkstra’s algorithm.

# ● FIFO

Groups:

icon Concepts

# ● Fill the mold

Groups:

icon Document Level Codes

# ● focus

Groups:

icon Modifiers

Comment:

This modifier denotes the purpose of the text that it is attached to is to further refine some aspect of that which is being discussed. To “focus in” to some aspect of that which is being discussed. This code is most likely no longer useful given that we now have explicit subscoping operators.

# ● frontier

Groups:

icon In Vivo Terms

# ● Goal

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes that the text is discussing the goal, the end game, that which is the desired outcome, of whatever the context was set to.

# ● Greedy

Groups:

icon Concepts  icon In Vivo Terms

# ● History

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes that the text is discussing the history of whatever the context was set to.

# ● Implementation

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes that the text is discussing implementation details of that which the context is set to. These are things such as “typically algorithm x is not implemented with a fibonnacci heap because of y and z”. This code does not refer to things that are required for the implementation of an algorithm, such as a priority queue for Dijkstra’s. Such things would either be Constituent or Operation tags.

# ● Implication

Groups:

icon Actions

Comment:

This code denotes that the text is giving an implication about the context. This could be anything that fits the logical connective if..then..else..

# ● In vivo term introduction

Groups:

icon Actions

Comment:

This code denotes that the text is defining a term that practitioners of algorithms would be familiar with. In Vivo terms are the “language of the trade” so to speak.

# ○ Industry

# ● Insertion

Groups:

icon Concepts  icon In Vivo Terms

# ● internal paths

Groups:

icon In Vivo Terms

# ● Invariant

Groups:

icon Concepts  icon In Vivo Terms

Comment:

Another in vivo term that may not be transmissible

# ● Legend

Groups:

icon Actions

Comment:

This code denotes that the text is giving a legend to understand something. For example if the text content is expressed in a graphic, there will commonly be a paragraph that describes, in detail, all things of interest in the graph. Here is a textual example: “The blue nodes are the recursive calls of merge sort, the yellow are the calls to the merge suboperation"

# ● length

Groups:

icon In Vivo Terms

# ● LIFO

Groups:

icon Concepts

# ● Mathematic

Groups:

icon Content Expression

Comment:

The content is represented using Mathematic formulae, variables, or equations

# ● Meta

Groups:

icon Modifiers

Comment:

This modifier takes any number of tags of any type. It denotes text that is not directly related to explaining the scope at hand. Ex: This algorithm is hard for many people to understand because of its simplicity.

# ● Motivation

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes that the text is discussing the motivation for whatever the current context is set to.

# ● Observation

Groups:

icon Actions

Comment:

This code denotes the text making a general observation about the context. Example: “At node 13 the tree has height 0” or “We can see that the last loop will never finish and will continue processing in infinitum"

# ● Operation

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotest that the text is discussing an Operation that is a requisite and central part of whatever the current context is set to.

# ● Outline

Groups:

icon Actions

Comment:

This code denotes that the text is giving a bulleted list of the content the document will go through

# ● path

Groups:

icon In Vivo Terms

# ● Pedagogical

Groups:

icon Modifiers

Comment:

A modifier, whose purpose is to denote that the specific purpose of a statement is pedagogical in nature.

# ● principle of optimatlity

Groups:

icon In Vivo Terms

# ● priority queue

Groups:

icon In Vivo Terms

# ● Problem

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes that the text is discussing a problem that is solved by whatever the context is referring to. This does not say what kind of problem it is, such things are handled through modifiers.

# ● Proof

Groups:

icon Actions

Comment:

This code denotes the text giving a mathematic or logical proof about the context

# ● Property

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes that the text is now discussing some property of whatever is the current context.

# ● Proposal

Groups:

icon Actions

Comment:

This code is similar to “Assumption” however this code denotes the text suggesting a path forward. For example: We can solve this problem with BFS algorithm

# ● rebalance

Groups:

icon Concepts  icon In Vivo Terms

Comment:

rebalance is an in vivo term for Trees in general. This means it will not be able to be applied to other algorithms, perhaps this code should be subsumed into SubOp code.

# ● recurrence

Groups:

icon In Vivo Terms

# ● Recursive

Groups:

icon Concepts  icon In Vivo Terms

# ● Related

Groups:

icon Modifiers

Comment:

This modifier takes any number of arguments of any type. It denotes that the tags it is attached to are substantially related to the thesis topic in some manner. Ex: The more general shortest-path problem, is related, to the motivating problem for Dijkstra’s algorithm.

# ● Review

Groups:

icon Modifiers

Comment:

This modifier takes any number of tags of any type. It denotes that the purpose of those tags is to provide a pedagogical review of the material to the reader.

# ● running time

Groups:

icon In Vivo Terms

# ● Search

Groups:

icon In Vivo Terms

# ● self-teaching

Groups:

icon Document Level Codes

# ● shortest-path

Groups:

icon In Vivo Terms

# ● simple paths

Groups:

icon In Vivo Terms

# ● Solicitation

Groups:

icon Actions

Comment:

This code denotes that the text is explicitly asking something of the reader. These statements are typically formed as questions.  Ex: “If you find the recursion confusing, make sure you go back and understand it before moving forward"

# ● Solution

Groups:

icon Elements of Explanatory Structure

Comment:

This code denotes that the text is now discussing a solution to whatever is in the current context. This is always used to “close” a previous “Problem” context.

# ● stable sorting

Groups:

icon In Vivo Terms

# ● State

Groups:

icon Elements of Explanatory Structure

Comment:

This scoping operator denotes that the text is now discussing something related to the state or the state of whatever is in the current context. Defaults to Thesis Topic

# ● stopping rule

Groups:

icon In Vivo Terms

# ● Structuring

Groups:

icon Elements of Explanatory Structure

# ● Summary

Groups:

icon Actions

Comment:

This code denotes a concluding block of text that summarizes the previous contexts

# ● Symmetry

Groups:

icon Concepts

# ● Table

Groups:

icon Content Expression

Comment:

The content is explicitly displayed in a Table

# ● Thesis Topic

# ● undirected graph

Groups:

icon In Vivo Terms

# ● unvisited

Groups:

icon In Vivo Terms

# ● weighted

Groups:

icon In Vivo Terms