CS405: Introduction to Empirical Modelling Assignment

Title

Empirically Modelling Self Balancing Binary Trees

Abstract

Understanding data structures is a key part of an education in Computer Science. They form the basis for the storage, retrieval and sorting of data. Generally these data structures are taught as transference of rigid ideas from the teacher to the pupil in a formal and abstract way. This takes away the sense of the student exploring the unknown, building up their own understanding of the data structure and in doing so loses the true breadth of the concept being conveyed. This has also led to a perceived inaccessibility of understanding data structures for those with less technical experience.

Educational tools based on empirical modelling principals focus on the development of a model, formed from the modellers experience of a construal. This model is comprised of definitions which can be continually updated allowing interaction with it, either directly or through pre-set actions. This allows for an overarching understanding of the construal and once this is built up, understanding of the traditional, formal representations becomes less difficult. Previous empirical modelling projects, such as the heapsort project, provide a representation of the data structure, which allows for a broader exploration of it than these conventional, formal methods.

This paper focuses on a different data structure than previous works, self balancing binary trees. It explores the construal behind this data structure in a similar fashion, modelling it with a set of definitions. It aims to educate the user on the form and function of the data structure whilst also allowing for manipulation of the definitions which underpin the model. Through inserting, deleting and manipulating the definitions of the model the user can gain an understanding of the structure without the usual constraints associated with it. It is also expected that the model will also lead into a better understanding of the construal for the modeller, which in turn will effect subsequent, currently unforeseen applications of the model.

Model

The model will be written using the tkEden tool, which provides the environment for the model to run in. The particular self balancing binary tree to be modelled is the red/black tree, a structure created in the 1970's. Nodes are coloured red or black and certain conditions are imposed, in relation to the surrounding nodes, to ensure height balancing.

Specifically, the standard rule of each binary tree that all nodes to the left of any node are smaller and the ones to the right larger, applies. Secondly, the number of black nodes from the root to any leaf is equal. Finally, that no two consecutive nodes are both red, which along with the previous condition ensures that the tree is balanced. These conditions will be modelled as dependencies which will update depending on the values of the items in the tree.

The model will support inserting new nodes into the tree. This will require the model to check how the new node affects the conditions expressed by dependencies in the model. Then the model will be updated by changing the definitions of the nodes with left or right rotations as appropriate, to ensure that the conditions are maintained. These left and right rotations involve the swapping of child and parent nodes, re-colouring of nodes following swaps then repeating if the conditions have still not been met. The removal of nodes will also be possible again triggering an update to the state of the model through its definitions to maintain the structure of the tree.

The data contained in the tree will also be expressed in an array in the model. This array will update with the tree, using the same definitions. This should also allow for the user to appreciate the relationship between the two representations and how the tree structure works if they are not familiar with it.

References

Jaratsri Rungrattanaubol, Meurig Beynon. Heapsort Extensions Project. <Online: http://empublic.dcs.warwick.ac.uk/projects/heapsortextendRun-bol2001/>

Frank Pfenning. Lecture Notes on Red/Black Trees. Principles of Imperative Computing, Lecture 17, Carnegie Mellon University. <Online: http://www.cs.cmu.edu/~fp/courses/15122-f10/lectures/17-rbtrees.pdf>

Meurig Beynon. Computing technology for learning - in need of a radical new conception. In Journal of Educational Technology & Society, 10 (1), 94-106, 2007.

Zhao Wang. The Research on Teaching Ideas of ”Data Structure and Algorithm” in Non-computer Major. Advances in Computer Science and Education, Volume 140, 2012, pp 249-254.

Weight

50%, 50% for paper and model respectively.