Assignment 2 - Marking Guide

#	Description plltyl003	Marks
1	HuffmanNode using: - std::shared_ptr<HuffmanNode> for left/right child nodes in HuffmanNode.h [1/1] - std::shared_ptr<HuffmanNode> for root node in HuffmanTree.h [1/1]	2/2

2	Six special member functions for nodes [5 Marks each as follows]
	    Mark breakdown for one of the 6 special member functions:
	      - Function present [1]
	      - Function defined (i.e they **MUST NOT** use =default) [2]
	      - Works correctly (0 if they used =default) [2]
	      - -1 in Move constructor and assignment operator. Need to ensure that all variables are left in an empty state.
		29/30

3	Unit tests for node class - Unit tests present [0.5/1] - -0.5 do not test all of node class methods. - All unit tests work correctly and pass [2/2]	2.5/3
4	Counting letter frequencies - HuffmanTree uses unordered_map for frequencies [2/2] - Implemented correctly [3/3]	5/5
5	Build tree - Used priority queue [4/4] - Used the "<" operator correctly to return smallest item in std::priority_queue<HuffmanNode, compare> [5/5] - Create new parent node (from the two nodes with the smallest letter frequency) [1/1] - Left child is smaller nodes [2/2] - Right child is larger nodes [2/2] - This new node has no letter associated with it, only a frequency (which is the sum of the frequencies of its two children) [2/2] Insert new node into the priority queue [2/2]	18/18
6	Traverse tree to build code table - Not present or very poor attempt = 0 marks - Made a very good attempt, but doesn't work = 5 marks - Works perfectly = 10 marks	10/10

7	Encode
	    - Encode each character as bit string [3/3]
	    - Write output_file.hdr [2/2]
		5/5

8	Correct program behaviour
	    - Program works perfectly **= 5 marks**
	    - Program doesn't run, produces incorrect output or breaks **= 0 marks**
		5/5

9	Elegance and Efficiency
	    - 5 makrs = Perfect, elegant code, efficient etc.
	    - 0 marks = Everything works, but redundant code, inefficient, untidy etc.
		5/5

10	Unit tests for tree class
	    - Unit tests present [1/1]
	    - Unit tests compile and run [2/2]
	    - Unit tests, test all functions [3/4]
	    - -1 Do not test all tree class functions.
		6/7

11	[Extra credit] Bit packing and writing to file
	    - Works correctly [4/4]
	    - Unit tests [1/1]
		5/5

12	[Extra credit] Reading and bit unpacking
	    - Works correctly [3/4]
	    - -1 Bit unpacking not always successful
	    - Unit tests [1/1]
		4/5

	TOTAL	**96.5/100**