```
// Macro Guard
#ifndef HUFFMANNODE_H
#define HUFFMANNODE_H

#include <ostream>  // Provides std::ostream. To overload '<<' operator
#include <string>   // Provides std::string

// To avoid using 'std'
using std::string;
using std::ostream;

class HuffmanNode
{
    // fields
    private:
        char letter;        // The character the node corresponds to
        int frequency;      // Number of times has thw character occured (i.e. KEY value)
        string code;        // Encoded Huffman Code
        HuffmanNode* left;  // Pointer to its left child
        HuffmanNode* right; // Pointer to its right child

    // functions
    public:
        // Default Constructor
        HuffmanNode();

        // Constructors
        HuffmanNode(char, int);
        HuffmanNode(HuffmanNode*, HuffmanNode*);
        HuffmanNode(const HuffmanNode&);

        // Destructor
        ~HuffmanNode();

        // Accessor methods
        int getFrequency()              {return frequency;};
        char getLetter()                {return letter;};
        string getCode()                {return code;};
        HuffmanNode* getLeft()          {return left;};
        HuffmanNode* getRight()         {return right;};

        // Mutator methods
        void setHuffmanCode(string s)   {code = s;};

        //Overloaded methods
        friend ostream& operator<<(ostream&, const HuffmanNode&);
        HuffmanNode& operator=(const HuffmanNode&);
};

#endif
```