```
public class HufPar {
        private char character;
        private double freq;

        public HufPar (char c, double f) {
                character = c;
                freq = f;
        }
        public char obtemCaracter() {
                return character;
        }
        public double obtemFreq() {
                return freq;
        }
}
public class NodoBinario {

        private NodoBinario esquerdo;
        private NodoBinario direito;
        private HufPar value;

        public NodoBinario (HufPar h) { value = h; }

        void insereEsquerda (NodoBinario n) { esquerdo = n; }
        void insereDireita (NodoBinario n) { direito = n; }

        NodoBinario getEsquerdo() { return this.esquerdo; }
        NodoBinario getDireito() { return this.direito; }

        HufPar getHufPar() {
                return value;
        }

        void printNodo() {
                System.out.print( this.getHufPar().obtemCaracter() );
                System.out.print( " - ");
                System.out.print( this.getHufPar().obtemFreq());
        }
}
public class HufEncoder {

        private static final char GEN_CHAR = '#';

        private FilaDePrioridade workspace;

        public HufEncoder ( HufPar[] p ) { workspace = new FilaDePrioridade(p); }

        NodoBinario geraArvoreDeHuffman(){
                if (this.workspace.numNodos == 1) {
                        return workspace.extraiMenor();
                } else {
                        NodoBinario x,y;
                        double f;
                        while ( this.workspace.numNodos > 1) {

                           x = workspace.extraiMenor();
                           y = workspace.extraiMenor();
                           f = x.getHufPar().obtemFreq() + y.getHufPar().obtemFreq();
                           NodoBinario z = new NodoBinario ( new HufPar ( GEN_CHAR,f ) );
                           z.insereEsquerda(x);
                           z.insereDireita (y);
                           workspace.insere(z);
                        }
                        return geraArvoreDeHuffman();
                }
        }
}
```