



Microprogetto “Stack di nodi”

1. Stack (pila)

Completa la definizione della classe `NodeStack` (in *code*) per sostituire `Stack<Node>` nei metodi che rielaborano iterativamente gli schemi ricorsivi utilizzati nel programma `HuffmanCompression`, garantendo le stesse funzionalità della classe predefinita (`Stack<Node>`). Il protocollo deve quindi prevedere il costruttore e i metodi così specificati:

<code>public NodeStack()</code>	<i>costruttore: creazione di uno stack di nodi vuoto</i>
<code>public boolean empty()</code>	<i>verifica se lo stack è vuoto</i>
<code>public Node peek()</code>	<i>restituisce l'elemento in cima allo stack (senza rimuoverlo dallo stack)</i>
<code>public Node pop()</code>	<i>restituisce l'elemento in cima allo stack e lo rimuove dallo stack</i>
<code>public void push(Node n)</code>	<i>aggiunge un nuovo elemento n in cima allo stack</i>

Realizza la rappresentazione interna utilizzando strumenti base di Java, in particolare *array* di nodi (`Node`), pensando a dove si possono registrare i nodi inseriti nella coda e come si può identificare fra questi quello inserito per ultimo.

2. Verifica

Verifica il buon funzionamento della soluzione proposta utilizzando il programma contenuta nella cartella *code*; in particolare, la versione di `HuffmanCompression` fa riferimento alla classe `NodeQueue` che avrai realizzato in accordo con i requisiti forniti sopra.