|  |
| --- |
| **WNode** |
| **-letter : char**  **-xcoord : int**  **-ycoord : int**  **-p4 : WNode**  **-p6 : WNode** |
| **+WNode(r : char, x : int, y : int, q4 WNode, q6 WNode)**  post: er is een WNode-object gecreëerd met letter=r, xcoord=x, ycoord=y, p4=q4, p6=q6  **+Rxcoord() : int**  post: retourneert xcoord;  **+Rycoord() : int**  post: retourneert ycoord;  **+LetterWnode() : char**  post: retourneert letter;  **+SetPosition4(res : WNode)**  post: p4=res.  **+SetPosition6(res : WNode)**  post: p4=res.  **+has4() : boolean**  post: retourneert true als de WNode een andere WNode aan de linkerkant bevat  **+has6() : boolean**  post: retourneert true als de WNode een andere WNode aan de rechterkant bevat  **+p4() : WNode**  post: retourneert p4.  **+p6() : WNode**  post: retourneert p6.  **+toString() : String**  post: retourneert een String representatie van de WNode. |

|  |
| --- |
| **Woordveld** |
| **-res : char[][]**  **-string : String[]** |
| **+readFile(row : int, col : int) throws IOException**  post: bestand wordt geladen, LeesFile wordt aangeroepen.  **+LeesFile(x : FileReader, row : int, col : int)**  post: Scanner doorloopt het bestand. VulVeld wordt aangeroepen met parameters row, col.  **+VulVeld(row : int, col : int)**  post: Vul wordt aangeroepen met parameters row,col,0.  **+Vul(row : int, col : int, p : int)**  post: als p gelijk is aan row, Print(row,col). Als p niet gelijk aan row, vul de rij. Vul wordt opnieuw aangeroepen met row,col,p+1.  **+Print(row : int, col : int)**  post: de woordzoeker wordt geprint  **+getArray() : char[][]**  post: retourneert res. |

|  |
| --- |
| **Woordboom (uitbreiding van Woordveld)** |
| **+ hulp : char[][]**  **+ root : WNode** |
| **+Woordboom()**  post: nieuw Woordboom-object wordt aangemaakt met size 0 en root een nieuw WNode-object.  **+root()**  post: Vulboom wordt aangeroepen met root,0,1 en root,1,0.  **+VulPosition4(ress : WNode, x : int, y : int)**  pre: x en y zijn groter of gelijk aan 0, x is kleiner dan res[0].lenght en y is kleiner dan res.length.  post: de boom wordt gevuld met alle WNode’s in de rij  **+VulPosition6(ress : WNode, x : int, y : int)**  pre: x en y zijn groter of gelijk aan 0, x is kleiner dan res[0].lenght en y is kleiner dan res.length.  post: de boom wordt gevuld. Wanneer VulPosition4(WNode ress, x int, y int) klaar is met het vullen van de rij, zal VulPosition6 aanroepen de volgende rij te vullen.  **+toString() : String**  post: toString(WNode t, int indents) wordt aangeroepen met WNode root en int 0.  **+toString(t : WNode, indents : int)**  post: er wordt een stringrepresentatie gegeven van de searchree. |
| **Zoeken (uitbreiding van Woordboom)** |
| **+pl1 : WNode**  **+pl2 : WNode**  **+Al : Set<WNode>**  **+xcoord : int[]**  **+ycoord : int[]**  **+listp1 : ArrayList<WNode>**  **+totaal : ArrayList<WNode>** |
| **+Start(L : String) throws IOException**  post: Woordboom() wordt aangeroepen. Een lege HashSet<WNode> wordt gecreëerd.  **+Zoekp4(R : WNode, x : int, L : String)**  pre: R.Rxcoord() is kleiner of gelijk aan res[0].length  post: Alle WNode’s met x-coördinaat R.Rxcoord worden doorlopen. Als de letter in de WNode gelijk is aan de gezochte letter, wordt deze toegevoegd aan de Set<WNode>.  **+Zoekp6(R : WNode, x : int, L : String)**  pre: R.Rycoord() is kleiner of gelijk aan res[0].length  post: Alle WNode’s worden doorlopen vanaf WNode R. Als de letter in de WNode gelijk is aan de gezochte letter, wordt deze toegevoegd aan de Set<WNode>.  **+Location(L : String)**  post: Locaties eerste letter van het te zoeken woord worden opgeslagen.  **+Zoek1(L : String, x : int, p2 : ArrayList<WNode>)**  post: Wanneer de gevonden letters in de zoekrichting corresponderen met het te zoeken woord worden deze opgeslagen in p2.  **+Zoek2(L : String, x : int, p2 : ArrayList<WNode>)**  post: Wanneer de gevonden letters in de zoekrichting corresponderen met het te zoeken woord worden deze opgeslagen in p2.  **+Zoek3(L : String, x : int, p2 : ArrayList<WNode>)**  post: Wanneer de gevonden letters in de zoekrichting corresponderen met het te zoeken woord worden deze opgeslagen in p2.  **+Zoek4(L : String, x : int, p2 : ArrayList<WNode>)**  post: Wanneer de gevonden letters in de zoekrichting corresponderen met het te zoeken woord worden deze opgeslagen in p2.  **+Zoek5(L : String, x : int, p2 : ArrayList<WNode>)**  post: Wanneer de gevonden letters in de zoekrichting corresponderen met het te zoeken woord worden deze opgeslagen in p2.  **+Zoek6(L : String, x : int, p2 : ArrayList<WNode>)**  post: Wanneer de gevonden letters in de zoekrichting corresponderen met het te zoeken woord worden deze opgeslagen in p2.  **+Zoek7(L : String, x : int, p2 : ArrayList<WNode>)**  post: Wanneer de gevonden letters in de zoekrichting corresponderen met het te zoeken woord worden deze opgeslagen in p2.  **+Zoek8(L : String, x : int, p2 : ArrayList<WNode>)**  post: Wanneer de gevonden letters in de zoekrichting corresponderen met het te zoeken woord worden deze opgeslagen in p2.  **+VindCombinatie(L String, x : int, p2 : ArrayList<WNode>)**  post: Zoekmethoden Zoek1, Zoek2,...,Zoek8 worden aangeroepen tenzij de lengte van L 1 is, dan wordt gelijk Coordinaat(listp1) aangeroepen.  **+Coordinaat(s : ArrayList<WNode>)**  pre: s heeft geen negatieve lengte.  post: De coördinaten worden weergegeven in een stringrepresentatie.  **+Check(p2 : ArrayList<WNode>, L : String)**  post: Iedere letter wordt nog een keer gecontroleerd. Als de lengte van het gevonden woord gelijk is aan het te zoeken woord, dan wordt Coordinaat() aangeroepen. |

|  |
| --- |
| **graph (uitbreding van Woordveld)**  **implementeert ActionListener** |
| **-jp : JPanel**  **-wf : JPanel**  **-size : JPanel**  **-veldd : JPanel**  **-cf : JButton**  **-tf : JTextField**  **-sw : JButton**  **-col : JTextField**  **-row : JTextField**  **-jf : JFrame**  **-veld : JTextField[][]** |
| **+graph()**  post: Er wordt een scherm geopend met breedte 765 en hoogte 650. Er worden tekstvelden aangemaakt, zodat de gebruiker de afmetingen van de woordzoeker in kan vullen en het te zoeken woord. De Button “Create field” en “Search Word” worden aangemaakt.  **+actionPerformed(ex : ActionEvent)**  post: Indien er op de “Create field”-button is geklikt wordt de woordzoeker aangemaakt. Indien er op de “Search Word”-button is gedrukt wordt het woord gezocht en gearceerd. |