Specificaties

We gebruiken twee klasses voor deze opdracht: Punt en Sudoku.

|  |
| --- |
| Punt |
| **-X : int**  **-Y : int** |
| **+Punt(x1: int, y1: int)**  post: er is een Punt-object geconstrueerd met X-coördinaat x1 en Y-coördinaat y1.  **+GetX() : int**  post: retourneert X.  **+GetY() : int**  post: retourneert Y. |

|  |
| --- |
| Sudoku |
| **-tabel : int[][]**  **-blokken : boolean[][][]**  //blokken gebruiken we als een 3x3 tabel van bitsets; elk vakje van de tabel stelt een 3x3 blok van de sudoku voor, en met de bitset houden we bij welke getallen al in het respectievelijke blok voorkomen. |
| **+Sudoku()**  post: er is een leeg sudoku-object geconstrueerd met een tabel van  9 rijen en 9 kolommen, en een 3-dimensionale boolean array van 3 bij 3 bij 9.  **+checkKolom(p : Punt, w : int) : boolean**  post: controleert of het getal w voorkomt in de kolom van punt p. Retourneert true als dit wel het geval is, en false als dat niet zo is.  **+checkRij(p : Punt, w : int) : boolean**  post: controleert of het getal w voorkomt in de rij van punt p. Retourneert true als dit wel het geval is, en false als dat niet zo is.  **+nummerVierkant(p: Punt) : Punt**  post: we geven elk blok van 3 bij 3 van de tabel een coordinaat mee; deze methode geeft de coordinaat van het blok waar het punt p in ligt.  **+checkBlok(p : Punt, w : int): boolean**  post: controleert of het getal w voorkomt in het blok van punt p. Retourneert true als dit wel het geval is, en false als dat niet zo is.  (Deze methode maakt gebruik van de methode nummerVierkant())  +**maakSudoku(s: String) : String**  post: construeert de sudoku gegeven door String s, lost deze op, en retrourneert de sudoku weer in de vorm van een string.  (Deze methode maakt gebruik van de methode LosOp())  **+LosOp(p : Punt)**  post: vult tabel zo in dat de sudoku klopt.  (deze methode is recursief en maakt gebruik van backtracking) |