Was soll das Programm können?

- gegebene Wörter in gegebenem Feld finden

- Ausgabe der Lösung

-> auch in benutzerfreundlicher Weise

- eigenes Feld erstellen

-> Größe und Wort: erstelle Feld damit

-> nur Wort: erstelle minimales Feld

- Felder extern speichern

Wie soll es strukturiert sein?

Class „Benutzeroberfläche“:

- Benutzerinteraktion entgegennehmen und verarbeiten

- Ergebnisse ausgeben

- Operationen delegieren

Class „Woertersuche“:

- Wörtersuche lösen

- Feld und Wörterliste speichern

Class „Wort“:

- Wort speichern

- Lösung für Wort speichern

- verwaltet Inhalt

Class „Feld“:

- speichert Feld

- verwaltet Inhalt

Class „Speicherwerk“:

- Woertersuche extern speichern

- Woertersuche aus dem Speicher laden

Enumeration „Richtung“:

- Richtungsbegriffe werden initialisiert

Wie soll es etwa umgesetzt werden?

Class „Benutzeroberfläche“:

-gitter: Woertersuche

---

+main(String… args)

// Wertet Benutzereingaben aus und delegiert sie

-printMenu()

// gibt Menüinformationen in der Konsole wieder

Class „Woertersuche“:

-feld: Feld

-woerter: Wort[]

---

+printWordlist()

+addToWordlist(wort: Wort)

+removeFromWordlist(index: int): boolean

+clearWordlist()

+createField(height: int, length: int): boolean

+createMinimalField()

+setField(*file*): boolean

// erstellt ein Feld nach Eingabe durch Textdokument

+getLength(): int

+getHeight(): int

+solve(): boolean

// findet Wörter in Feld; lässt Lösungen von den einzelnen Worten selbst speichern

-printSolution()

Class „Wort“:

-WORT: String

// jedes Wort sollten ausschließlich in Großbuchstaben gespeichert werden und sollte auch nicht veränderbar sein, dafür erstellt man dann neue Worte

-richtung: Richtung

-x: int

-y: int

---

+Wort(wort: String)

// Wort ohne Lösung

+Wort(wort: String, richtung: String, x: int, y: int)

// Wort mit Lösung

+getWord(): String

+getPositionInField()

// wird in Konsole ausgegeben

+setSolution(richtung: String, x: int, y: int)

**//eventuell mit nested Class**

Class „Feld“:

-FELD: char[][]

// Feldgröße ist nicht veränderbar, aber der Inhalt

---

+Feld(length: int, height: int)

+setChar(x: int, y: int, c: char)

+getChar(x: int, y: int): Char

Class „Speicherwerk“:

+save(woertersuche: Woertersuche)

+load(path: String): *file*

+saveAndLoad(woertersuche: Woertersuche, path: String)

Enumeration „Richtung“:

-richtung: String