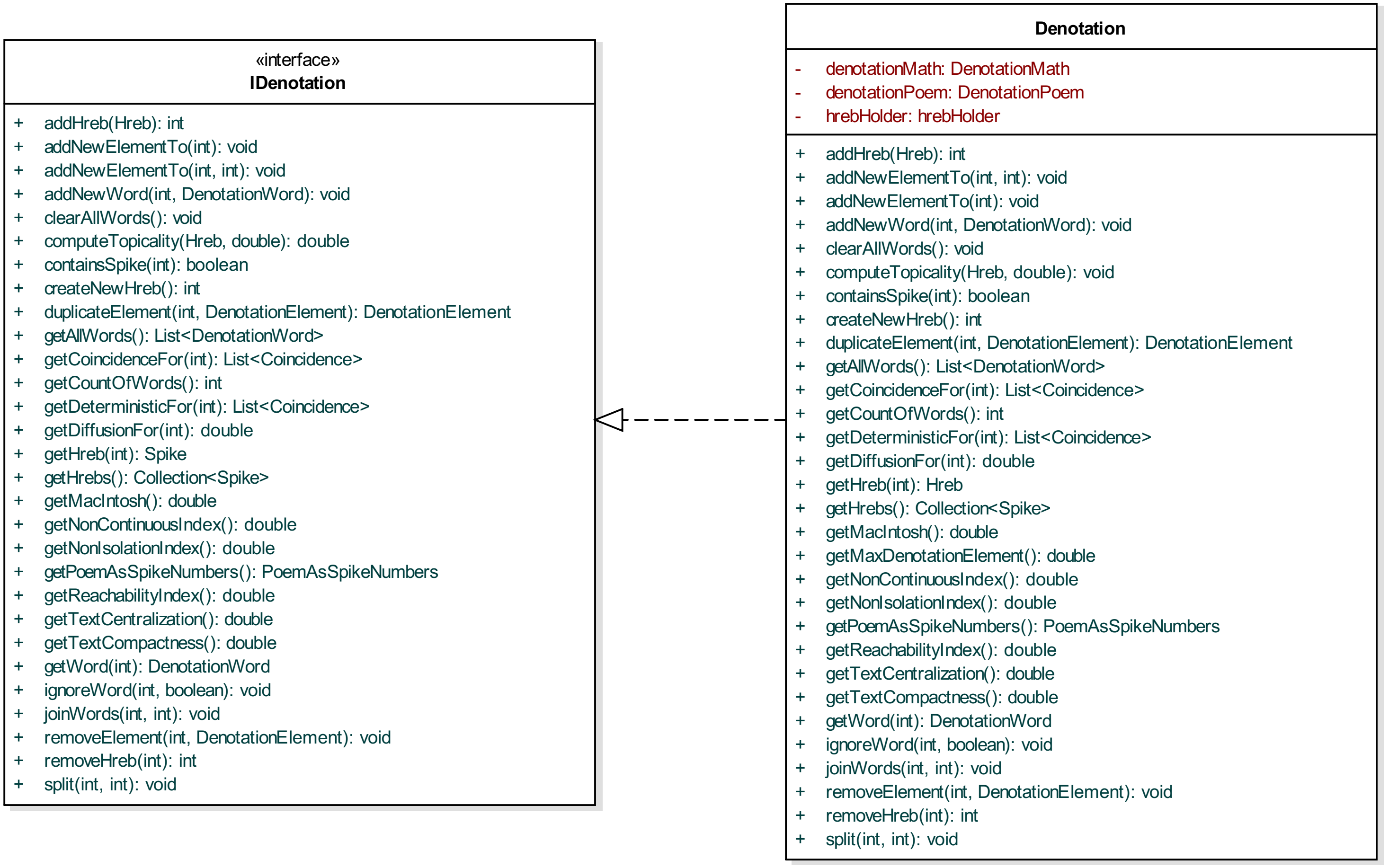
«interface»
IDenotation
+ addHreb(Hreb): int
+ addNewElementTo(int): void
+ addNewElementTo(int, int): void
+ addNewWord(int, DenotationWord): void
+ clearAllWords(): void
+ computeTopicality(Hreb, double): double
+ containsSpike(int): boolean
+ createNewHreb(): int
+ duplicateElement(int, DenotationElement): DenotationElement
+ getAllWords(): List<DenotationWord>
+ getCoincidenceFor(int): List<Coincidence>
+ getCountOfWords(): int
+ getDeterministicFor(int): List<Coincidence>
+ getDiffusionFor(int): double
+ getHreb(int): Spike
+ getHrebs(): Collection<Spike>
+ getMacIntosh(): double
+ getNonContinuousIndex(): double
+ getNonIsolationIndex(): double
+ getPoemAsSpikeNumbers(): PoemAsSpikeNumbers
+ getReachabilityIndex(): double
+ getTextCentralization(): double
+ getTextCompactness(): double
+ getWord(int): DenotationWord
+ ignoreWord(int, boolean): void
+ joinWords(int, int): void
+ removeElement(int, DenotationElement): void
+ removeHreb(int): int
+ split(int, int): void
Denotation
- denotationMath: DenotationMath
- denotationPoem: DenotationPoem
- hrebHolder: hrebHolder
+ addHreb(Hreb): int
+ addNewElementTo(int, int): void
+ addNewElementTo(int): void
+ addNewWord(int, DenotationWord): void
+ clearAllWords(): void
+ computeTopicality(Hreb, double): void
+ containsSpike(int): boolean
+ createNewHreb(): int
+ duplicateElement(int, DenotationElement): DenotationElement
+ getAllWords(): List<DenotationWord>
+ getCoincidenceFor(int): List<Coincidence>
+ getCountOfWords(): int
+ getDeterministicFor(int): List<Coincidence>
+ getDiffusionFor(int): double
+ getHreb(int): Hreb
+ getHrebs(): Collection<Spike>
+ getMacIntosh(): double
+ getMaxDenotationElement(): double
+ getNonContinuousIndex(): double
+ getNonIsolationIndex(): double
+ getPoemAsSpikeNumbers(): PoemAsSpikeNumbers
+ getReachabilityIndex(): double
+ getTextCentralization(): double
+ getTextCompactness(): double
+ getWord(int): DenotationWord
+ ignoreWord(int, boolean): void
+ joinWords(int, int): void
+ removeElement(int, DenotationElement): void
+ removeHreb(int): int
+ split(int, int): void