String

1. 주제: 텍스트 파일에서 상관어(correlative words) 찾기

○ 상관어란?
- 서로 관련이 있는 단어의 쌍을 의미한다.
- 상관어를 찾는 대표적인 방법은 문장에서 빈번히 같이 사용되는 단어의 쌍을 찾는 것이다.

○ 상관어를 찾는 알고리즘
- 문장에 등장하는 모든 단어의 쌍에 대해 빈도수를 저장한 후, 가장 많이 등장하는 k 개의 단어 쌍을 찾는다.
- “문장 이란 ‘.’ ‘?’ ‘!’ 으로 구분되는 문자열을 의미한다.
- 단, 관사나 대명사, 접속사 등 중요도가 떨어지는 단어(stopword)들은 포함하지 않는다.
- 같은 단어 쌍에 대한 중복 계산을 피해야 한다. 이를 위하여 모든 단어는 소문자로 변경한다.

○ 입력:
- 파일 이름, 단어 쌍의 수? data.txt k
  - 입력 파일의 구성: 영어 텍스트 파일
  - 단어 쌍의 수: k ← int

- Stopword들이 저장된 stop.txt 파일은 이름이 고정되므로 파일 이름을 입력받지 않는다.

○ 출력
- 입력 파일에서 stopword를 제외하고 가장 많이 등장하는 k 개의 문자와 빈도수
- 입력 파일에서 stopword를 제외하고 가장 많이 등장하는 k 개의 문자 쌍과 빈도수
- 문자 쌍은 크기가 작은 단어가 먼저 나올 것
- 빈도수가 같은 경우, 단어의 오름차순으로 출력 (단어 쌍에 대해서도 동일)

2. 제출 내용: HW3.java 하나의 파일만 제출

① public class HW3 (파일 내에 나머지 클래스들은 public이 아님)
② default package 사용
③ 프로그램 내에 주석은 모두 삭제
④ Eclipse Workspace의 한글 encoding은 MS949로 설정
← **위의 조건들을 만족하지 않는 과제물은 심사하지 않음!!**

3. 동작의 예

실행의 예 1:

```
파일 이름, 단어 쌍의 수? blockchain.txt 3

Tok-k 문자열: blockchains(7) block(6) data(5)
Top-k 단어쌍: [block, data](3) [bitcoin, blockchain](2) [bitcoin, public](2)
```

실행의 예 2:

```
파일 이름, 단어 쌍의 수? novel.txt 5

Tok-k 문자열: whale(481) like(289) man(262) ship(239) captain(216)
Top-k 단어쌍: [sperm, whale](68) [whale, white](58) [ahab, captain](47)
                    [man, old](41) [great, whale](34)
```

4. 평가: 50점 만점

- 문자열: 10점, 단어 쌍: 40점
- 실행 결과가 틀리면 0점
- JDK 8로 컴파일할 예정
- 프로그램 구성이나 성능에 심각한 문제가 있으면 실행 결과와 관계없이 감점 처리함 (indentation, 변수나 함수 이름, Collection 객체들을 무분별하게 사용 등)
- 반대로 프로그램 구성이나 성능이 우수할 경우 점수 상향도 가능함