**Report 3**

날짜 : 2020년 5월 13일(수)

제출마감: 2020년 5월 19일(화)

제출방법: eclass 에서 제출

* 제출 파일 : java 프로그램(.java 파일)과 도큐먼트(.docx, .hwp, .pdf 파일등)
  + java 프로그램 : 프로그램 설명 주석, 들여 쓰기, 띄어쓰기 등등이 포함된 java 언어 소스 파일, 하나의 문제는 하나의 .java 파일로 만들어야 함
  + 도큐먼트 : 커버 페이지, 문제에 대한 해결 방법 설명, 입력 및 출력 결과(화면 캡쳐) 등 포함.
  + 각 문제 20점 만점, 총 80점 만점으로 채점합니다.
* 파일명: “report3.zip” 으로 파일 하나로 묶어 제출

**Problems**

1. 주어진 “input.txt” 파일에 포함된 내용을 읽어서 단어를 검색하는 프로그램을 만들고자 한다. 단어의 시작 문자나 끝 문자를 검색하여 단어를 나열하고, 총 몇 개인지 출력한다. 아래 예시와 같이 동일한 출력 결과가 나오도록 프로그램을 쓰시오.

이 프로그램은 3개의 클래스/인터페이스로 구성된다.

1. CountWordEx : main() 메소드를 호출하여 프로그램 실행
2. CountWord : CountWordInterface를 구현하는 클래스
3. CountWordInterface : 아래 메소드로 정의됨

**public** **interface** CountWordInterface {

**public** **void** makeWordArray(); // 단어들의 배열 생성

**public** **void** printWordArray(); // 단어들의 배열 출력

**public** **int** countCWord(**char** c, **boolean** head);

// head == true 이면 c로 시작하는 단어

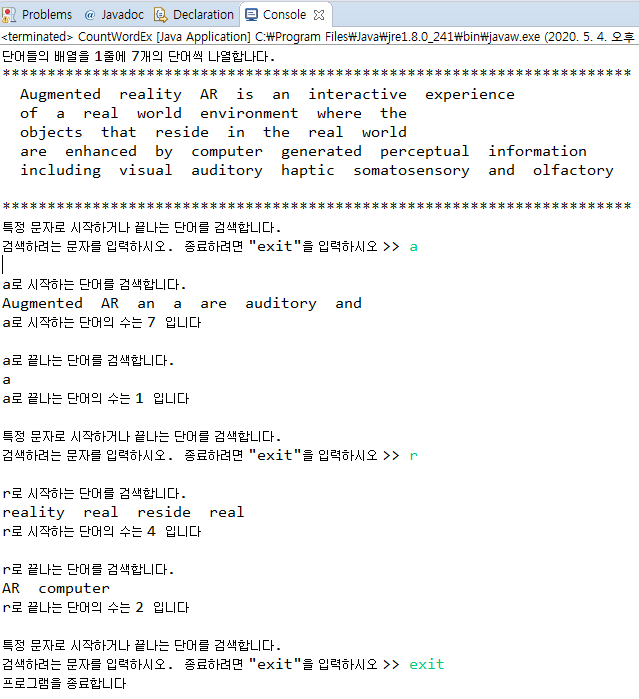
// == false 이면 c로 끝나는 단어

// 총 개수 구하여 리턴

}



실행 예시)



1. 사용자로부터 입력 파일명과 출력 파일명을 입력 받고, 입력 파일에 들어 있는 텍스트의 문자수, 단어수, 문장수를 count하여 출력하는 프로그램을 쓰시오. 출력결과는 출력 파일에 저장되도록 한다. BufferedReader/BufferedWriter 클래스를 사용해서 프로그램을 쓰시오.

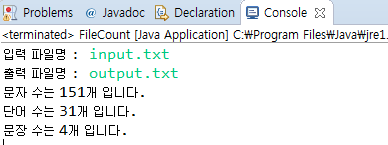
* 문자 : 공백이 아닌 모든 문자
* 단어 : 알파벳이나 숫자로 이루어진 연속적인 문자
* 문장 : 세가지 문자(. ? ! )로 끝나면 하나의 문장이다.
* 입력 파일 샘플 : input.txt



* 출력 파일 : output.txt (각줄에 문자수, 단어수, 문장수를 출력한다)

출력 화면에도 동일한 내용 출력

* 출력 결과 예시(주어진 input.txt 파일 사용하는 경우)



1. 입력 파일(예: word.txt)에서 단어나 문장을 검색하여 라인 번호와 함께 출력하는 프로그램을 쓰시오. 아래 인터페이스를 참고하여 메소드를 정의하거나, 다른 방법으로 인터페이스를 정의하여 사용해도 됨.

**public** **interface** WordSearchInterface {

//파일로부터 한줄씩 문자열을 읽어서 Vector<String>에 저장한다.

**void** readFile(String fileName) ;

// 저장한 Vector<String>으로부터 한줄씩 문자열을 가져와서

// indexOf(word) 메소드를 이용하여 포함 단어/문장의 포함 유무를 확인한후,

// 저장한 Vector<Integer>에 단어가 발견된 문장의 줄번호를 저장한다.

Vector<Integer> searchWord(String word);

// Vector<Integer>에 저장된 줄번호를 하나씩 읽고,

// Vector<String>으로부터 해당 줄번호의 문장을 출력한다.

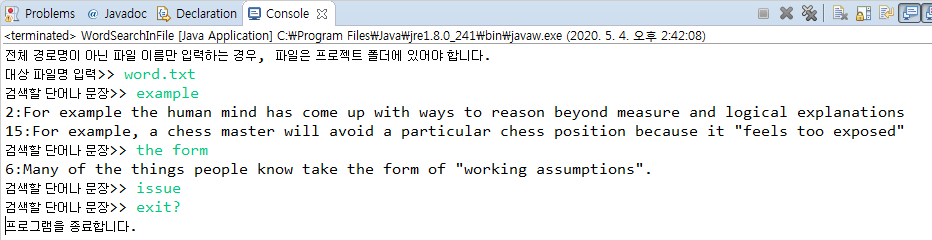
**void** printLines(Vector<Integer> noVector);

}

실행 결과의 예는 아래와 같고, “exit?” 를 입력하면 프로그램을 종료한다.



실행결과:



1. 아래 실행 결과를 참고하여 두 파일의 내용이 같은지를 검사하는 프로그램을 쓰시오. 이프로그램을 아래 두개의 클래스로 나누어 구현하고, 메소드는 자유롭게 정의하여 사용하면 된다.

**class** CompareFiles { } // 파일 내용 비교

**class** CompareFileApp { } // main() 메소드 포함

실행 결과)

