

프로그래밍 과제 5

20222663 권기영

1. 전자사전 만들기 - storeWords(), searchWords()

<구현>

1) storeWords()

텍스트 파일에서 문장을 하나씩 불러와서 단어/뜻 으로 문자열 파싱 후 단어 배열에 넣는 함수이다.

‘:’을 기준으로 파싱한다.

단어 각각에 대한 구조체를 만들어 관리하는데, 멤버로는 단어, 뜻, 단어 길이, 모음 개수 총 4개가 있다.

약 58000개의 단어가 있다고 했으므로 60000 크기의 단어 구조체 배열을 저장공간으로 초기화하였고, fgets로 한 줄을 읽고 한 줄에 대한 단어, 뜻, 단어 길이, 모음 총 4개를 구해서 멤버의 값을 바꿔준다.

모든 단어를 다 저장했다면 ALL STORED를 출력한다.

2) searchWords()

사용자로부터 입력을 받은 뒤, 전자사전 전체를 for 문으로 돌면서 Dict[i].word와 비교해서 같은 단어가 있으면 뜻을 출력, 다 돌았는데도 같은 단어가 없으면 문장을 출력한다. 엔터를 입력받으면 종료한다.

<실행 예>

```
ALL STORED
1 : 단어 찾기, 2 : 끝말 잇기, 3 : 단어 분석, 4 : 종료
>> 1
단어?(엔터만 입력하면 종료) apple
apple : n.사과
단어?(엔터만 입력하면 종료) banana
banana : n.바나나
단어?(엔터만 입력하면 종료) kitty
kitty : n.새끼고양이
단어?(엔터만 입력하면 종료)
```

2. 끝말 잇기 - concludingRemarks()

<구현>

- 1) rand()함수를 통해 전체 사전 단어 중 하나를 고른다.
- 2) 총 3개의 목숨을 부여한다.
- 3) 단어의 길이가 3~10이 아니거나, 끝말잇기가 안되거나, 사전에 없는 단어거나, 이미 나온 단어라면 그에 맞는 오류 메시지를 출력하고 기회를 1회 소진한다.
- 4) 모든 조건에 충족한다면, 점수를 1점 올리고, 사용자가 입력한 단어를 다음 사용자가 입력할 단어와 비교하는 단어로 바꿔준다.
- 5) 기회 3번을 전부 소진하면 종료한다.

<실행 예>

```
ALL STORED
1 : 단어 찾기, 2 : 끝말 잇기, 3 : 단어 분석, 4 : 종료
>> 2
컴퓨터 : laryngeal
끝말잇기 단어 : lean
현재 점수 1점

끝말잇기 단어 : near
현재 점수 2점

끝말잇기 단어 : rear
현재 점수 3점

끝말잇기 단어 : rear
이미 나온 단어입니다.
현재 점수 3점

끝말잇기 단어 : re
단어의 길이가 3이상 10 이하가 아닙니다.
현재 점수 3점

끝말잇기 단어 : radvuf
단어사전에 없는 단어입니다
모든 기회를 소진했습니다.
끝말잇기 종료. 총 점수는 3점
```

이미 입력한 단어 rear를 입력하니 오류 메시지를 출력. 길이가 2인 단어 re를 입력하니 오류 메

세지를 출력. 없는 단어를 입력하니 오류 메시지를 출력. 총 3번의 오류가 발생했고, 끝말잇기를 종료시킴.

```
ALL STORED
1 : 단어 찾기, 2 : 끝말 잇기, 3 : 단어 분석, 4 : 종료
>> 2
컴퓨터 : lewdness
끝말잇기 단어 : strike
현재 점수 1점

끝말잇기 단어 : eraser
현재 점수 2점

끝말잇기 단어 : rice
현재 점수 3점

끝말잇기 단어 : angle
끝 글자와 첫 글자가 다릅니다.
현재 점수 3점

끝말잇기 단어 : eager
현재 점수 4점

끝말잇기 단어 : role
현재 점수 5점

끝말잇기 단어 : even
현재 점수 6점

끝말잇기 단어 : note
현재 점수 7점

끝말잇기 단어 : establish
현재 점수 8점

끝말잇기 단어 : high
현재 점수 9점

끝말잇기 종료. 총 점수는 9점
```

중간에 한번 끝말과 첫말이 다른 단어를 입력해서 오류 메시지가 한번 나옴. 그 외에 다른 입력이 모두 끝말잇기 조건에 부합하여 총 10턴이 진행된 뒤 끝났고, 한 번의 오류 외에 모두 성공했으므로 총 점수 9점이 출력된다.

3. 단어 분석 - wordAnalyze()

<구현>

1) compareByLength

구조체의 멤버 변수인 length를 기준으로 비교하는 함수이다. 길이를 내림차순으로 정렬해준다. void *로 전달받기 때문에 ((Dict *)a)->length 와 같이 형변환을 시켜줘야 함.

2) compareByGather

구조체의 멤버 변수인 gather을 기준으로 비교하는 함수이다. 모음의 개수를 내림차순으로 정렬해준다. 위와 같은 형변환이 진행되어야 한다.

3) wordAnalyze

length를 기준으로 qsort를 진행 후 dict[0]~dict[9]까지 출력. 그 뒤에 gather을 기준으로 qsort를 진행 후 dict[0]~dict[9]를 다시 한번 출력한다.

<실행 예>

```
1 : 단어 찾기, 2 : 끝말 잇기, 3 : 단어 분석, 4 : 종료
>> 3
글자 수 : 20, potassium dichromate : 중크롬산 칼륨
글자 수 : 20, equilateral triangle : 등변 삼각형
글자 수 : 20, breach of confidence : 비밀 누설
글자 수 : 20, pin something on one : 아무에게 무엇의 책임을 지우다
글자 수 : 20, reflecting telescope : 반사 망원경
글자 수 : 20, occupational disease : n.직업병
글자 수 : 20, temperature selector : 온도 선택기
글자 수 : 20, turbopropeller plane : 터보프로펠러기
글자 수 : 20, refracting telescope : 굴절 망원경
글자 수 : 20, reciprocating engine : 왕복 기관

모음 개수: 10, equatorial guinea : 적도 기니
모음 개수: 10, occupational disease : n.직업병
모음 개수: 9, pernicious anemia : 악성 빈혈
모음 개수: 9, quadratic equation : n.2차방정식
모음 개수: 9, equilateral triangle : 등변 삼각형
모음 개수: 9, semidemisemiquaver : n.64분 음표
모음 개수: 9, autointoxication : n.자가 중독
모음 개수: 9, autoinoculation : n.자가 접종
모음 개수: 9, ease someone out : 교묘히 해고하다
모음 개수: 9, qualitative analysis : n.(화학)정성 분석
```

길이 순으로 단어 10개를 출력한 뒤, 모음 개수 순으로 단어 10개를 출력한다.

4. 전체 프로그램 실행 - executeProgram()

총 4개의 입력 옵션(1 : 단어사전, 2: 끝말잇기 3: 단어분석 4: 종료)이 있고, 프로그램을 종료하기 전 까지 반복해서 입력을 받는다.

입력에 맞는 함수를 실행한다.