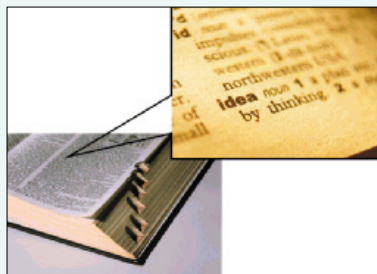


## 11

## 프로그래밍 프로젝트

- 01** 본문에서 설명된 선형 조사법의 소스는 단순히 탐색 키만을 해시 테이블에 저장하였다. 실제 응용에서는 이렇게 탐색 키만 저장해서는 아무 쓸모가 없다. 탐색 키와 관련된 자료들을 저장했다가 다시 꺼낼 수 있어야 한다. 본문의 프로그램을 참고하여 간단한 영어 사전을 만들어보자. 영어단어와 그 단어의 의미가 같이 해시 테이블에 저장된다. 프로그램이 시작될 때 파일에 저장된 영어 단어와 의미를 읽어서 해시 테이블에 저장한다. 사용자가 영어 단어의 탐색을 요구하면 해시 테이블에서 그 단어를 탐색하여 단어의 의미를 사용자에게 출력한다.



- 02** 선형 조사법과 이중 조사법을 비교하는 실험을 해보자. 먼저 500개의 사용자 이름이 들어 있는 리스트를 만든다. 그리고 크기가 1000인 해시 테이블을 선형 조사법과 이중 조사법으로 구현하여 500개의 이름이 해시 테이블에 추가될 때 충돌이 얼마나 일어나는지를 기록하라. 동일한 실험을 테이블 크기가 950, 900, 850, 800, 750, 700, 650, 600일 때 수행해본다.
- 03** 해싱을 이용하여 문자열을 저장할 수 있는 프로그램을 작성한다. 충돌해결책은 체인법으로 한다. 해시 테이블의 크기는 350이다. 그리고 입력 데이터 파일은 425개의 사용자 이름이다. 다음과 같은 해싱함수를 가정하라.

$$h(i) = i \% 350$$

입력 문자열은 아스키 코드를 모두 더하는 방식으로 정수로 변환된다. 테이블에 존재하는 연결 리스트들의 길이를 분석하여 체인법을 사용하는 해싱의 효율성을 측정하고자 한다. 즉 길이가 0, 1, 2, 3.. 인 연결리스트의 개수를 세어서 이들 결과를 분석한다. 결과를 분석하여 충돌이 심하게 일어나면 해시 함수를 변경하여 개선하여보라.

- 04** 간단한 재고 관리 프로그램을 해싱 기법을 이용하여 만들어보자. 다음과 같은 필드가 레코드안에 존재하여야 한다.

- 5자리의 부품 번호(키필드가 된다)
- 10자이내의 부품 설명
- 재주문 여부(yes아니면 no)
- 현재 재고량

이 재고 관리 프로그램을 사용하는 사용자는 다음과 같은 연산들을 수행할 수 있어야 한다.

- 재고 목록에 새로운 부품을 추가
- 재고 목록에서 부품 검색
- 부품의 키필드 변경
- 부품의 임의의 필드 변경