SSE2030: INTRODUCTION TO COMPUTER SYSTEMS (Fall 2011)

Programming Assignment #1
Due: September 23, 11:59 PM

1. Introduction

- 과제를 통해 File I/O 및 Data structure사용에 익숙해지도록 한다.

2. Overview

- Search engine에서 indexing은 사용자가 입력한 키워드를 이용하여 매우 크고 많은 문서에서 해당하는 키워드가 나타나는 위치를 빠르게 찾을 수 있도록 도와주는 기술이다. 본 과제에서는 주어진 Holy Bible의 창세기(Genesis) chapter를 이용하여 index를 만들어 파일에 저장하는 역할을 수행하는 "Index Builder"와, 생성된 index를 이용한 "Index Printer"를 작성하는 것이다. 다음 과제에서는 index builder를 업그레이드 하며, 생성된 파일을 읽어 들여 search를 하는 프로그램을 작성할 예정이다.

3. Specification

◆ Index Builder
- 주어진 성서 창세기를 이용하여 단어별로 index를 만든다.
  - ✓ 입력 파일은 http://atschool.eduweb.co.uk/SBS777/bible/text/genesis.txt에서 다운로드 받을 수 있다.
  - ✓ 입력 파일에서 장:절: 이 포함된 라인에 대해서만 index를 만들고, 장:절: 이 포함되지 않은 라인은 index 및 검색에 대해서 제외한다.
  - ✓ 성서 이름은 입력 파일에서 확장자를 제외한 이름으로 한다.
    예를 들어, genesis.txt 파일의 성서 이름은 genesis로 한다.
- Index 생성 시간, 탐색 시간, 크기를 고려하여 data structure 를 디자인 한다.
  - ✓ **Bonus** - Hash 알고리즘을 사용하여 index를 관리하도록 한다.
- Index 마다 단어, 나타난 장/절, 횟수, 절 내에서의 위치 등을 관리한다.
- Index는 파일로 저장한다.

◆ Index Printer

- Index builder에서 생성된 index 파일을 읽어 들여 인덱스 된 내용을 정해진 형식에 따라 출력한다
  - ✓ 출력 파일의 첫 부분에는 성서 이름: 장 수, 총 절 수, 총 인덱스 수, 총 워드 수를 출력해야 한다.
  - ✓ 이 후에는, 매 index 마다 다음의 포맷으로 출력한다.
    단어: 총 출현 횟수, 장:절:위치, 장:절:위치, …
  - ✓ 단어를 sorting하여 출력할 필요는 없다.

4. Restriction

- 과제는 본인이 직접 설치한 리눅스 환경에서 수행하고, 과제 보고서에 컴파일하고 실행한 화면을 캡쳐하여 추가한다.
- 임의의 큰 Text 에 대해서도 처리가 가능하게 한다.
- DataBase, Lex & Yacc 는 사용하지 않는다.
- Standard C library 이외의 library 는 사용하지 않는다.
  파일의 입출력은 open(), close(), read(), write(), lseek() 등의 system call 중 필요한 것을 선택하여 사용해야 한다.
- 단어의 대, 소문자는 구별할 필요가 없으며 영문 알파벳, – (하이픈), ' (어퍼스트로피) 를 제외한 문자는 단어에서 제외하여 인덱스에 추가한다.
  예를 들어, index로 관리되는 단어들은 다음과 같다.
  god, and, adam, brother's, priests', kirjath-arba, sons'
- genesis.txt 뿐만 아니라 http://atschool.eduweb.co.uk/SBS777/bible/text/에 있는 임의의 파일에 대해서도 동작해야 한다.
- Index를 저장하는 파일의 형식은 자유롭게 결정하되, 파일의 형식에 대해서 보고서에 반드시 기술해야 한다.

5. Hand in instructions

- 작성한 프로그램 코드 상단에 이름과 학번을 주석으로 표기한다.
- 과제 및 보고서는 제출시 "학번.zip" 파일로 압축하여 제출한다. (예: 2011711283.zip)
- 프로그램의 설계, 구현에 관한 내용을 담은 보고서를 별도로 제출한다. 보고서에는 프로그램의 구조를 그림으로 삽입하도록 한다. 보고서는 워드, 한글 등의 형식도 상관 없지만 가능하면 PDF 포맷으로 제출한다.
- 과제는 sse2030@csl.skku.edu로 보내고, 메일의 제목은 아래와 같은 형식을 따른다.
  [SSE2030] PA1, 학번, 이름

6. Logistics

- 과제 제출 결과는 http://csl.skku.edu/SSE2030F11 에서 확인할 수 있다.
- 과제 제출 시간은 메일 도착시간을 기준으로 하며, 기한 이후 24시간 내에 제출할 경우 30%, 48시간 내에 제출할 경우 60% 감점한다. 그 이후에는 0점 처리한다.
- 과제에 대한 의논은 함께 할 수 있으나, 프로그램 소스코드 작성은 스스로 해야 한다.
- 다른 사람의 과제를 copy 한 경우, 두 사람 모두 0점 처리한다. 인터넷 등에서 찾은 소스 코드를 그대로 copy 한 경우에도 0점 처리한다. 두 번 이상 이와 같은 이유로 0점 처리된 경우 F 학점을 받을 수 있다.

7. Example

```
$ ./indexBP genesis.txt                          // index builder & printer 실행
** Index Builder : Start          **
   Elapsed Time   : 62558(usec)                  // index builder 수행시간이 usec 단위로 나타남
** Index Builder : End            **

** Index Printer : Start          **            // indexBP를 실행하면 index file인 genesis_index와 output file인
** Index Printer : End            **            // genesis_ouput이 생성됨
$ ./cat genesis_output
genesis: 50 1533 2510 38260                      // 성서: genesis, 장수: 50, 절수: 1533, Index수: 2510, 단어수: 38260
leaves: 1, 3:7:94                                // 단어: 총 개수, 장:절:위치
ephraim's: 3, 48:14:58, 48:17:153, 50:23:15      // 단어: 총 개수, 장:절:위치, 장:절:위치, 장:절:위치
hadar: 2, 25:15:0, 36:39:43                      // 위치는 절 내에서 단어가 나타나는 위치로, Byte 단위로 한다.
depart: 2, 13:9:149, 49:10:22
parted: 1, 2:10:73
…
$
```