

## 1. Introduction

- 프로젝트를 통해 FILE I/O, Socket programming, 및 Data structure 사용에 익숙해지도록 한다.

## 2. Overview

- **Search engine의 Indexing**은 요청한 키워드를 빠르게 찾아주기 위해 데이터를 모아서 분석하고 저장하는 기술이다. 본 프로젝트는 주어진 Holy Bible의 chapter를 이용하여 index를 만들고 (Index Builder), 해당 index를 관리하는 파일을 만들어 출력하고(Index Printer), client가 요구하는 keyword의 위치를 찾아서 전송(Keyword Searcher)하는 기능을 하는 server와 성서에서 사용자가 원하는 단어의 위치의 출력을 요청하는 client를 구현하는 것이다.

## 3. Specification

### • Server

- ✓ Server 실행 시 IP, Port 번호를 argument로 입력 받는다.
- ✓ Client로부터 키워드를 문자열로 전송 받아 키워드에 대한 검색을 수행하고, bible.txt 파일에서 관리하는 모든 Index 파일로부터 검색 결과를 전송한다.
  - 이를 위해 index builder, index printer, index searcher 3가지를 구현한다.
- ✓ 서버에서는 Index builder, Index Printer가 실행된 뒤 Index Searcher가 동작한다.
- ✓ **Index Builder**
  - 입력 파일에서 "장:절:"이 포함된 라인에 대해서만 index를 만든다. "장:절:"이 포함되지 않은 라인은 index 및 검색을 제외한다.
  - 성서 이름은 입력 파일에서 확장자를 제외한 이름으로 한다.  
예: genesis.txt 파일의 성서 이름은 genesis로 한다.
  - 서버가 실행되는 위치에서 bible이라는 directory를 찾아 그 안에 있는 모든 txt 파일에 대해 index를 생성한다.(./bible/ 안에 있는 모든 txt파일)
  - Hints
    - ✧ Index 생성 시간, 탐색 시간, 크기를 고려하여 data structure를 디자인 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 정해진 형식에 따라 출력한다.
  - 출력파일의 첫 부분에는 성서 이름: 장 수, 총 절 수, 총 인덱스 수, 총 워드 수를 출력한다.

- 이후 매 index 마다 다음의 형식으로 출력한다.

단어: 총 출현 횟수, 장:절:위치, 장:절:위치, ...

- 단어를 sorting하여 출력할 필요는 없다.

- Index file의 예시는 아래와 같다.

```

$ ./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

```

- ✓ Server에서는 index builder와 index printer로 만든 index 파일들을 모아 bible.txt 파일로 모아 관리한다.

- Index builder와 Index printer로 만든 index 파일들의 목록을 관리한다.

### ✓ Keyword Searcher

- bible.txt로부터 만들어진 index list를 체크하여 만들어진 index를 메모리에 load 한다.

- 사용자가 키워드를 입력하면 해당하는 키워드가 나타나는 장, 절, 위치를 출력한다.

- ✧ 위치는 각 절의 처음에서부터의 offset을 의미한다. 절의 첫 번째 글자는 offset을 0으로 한다.

- ✧ 키워드가 한 개인 경우 장, 절, 위치를 출력한다.

- ✧ 키워드가 여러 개인 경우 장, 절을 출력한다.

- 키워드는 여러 개(개수 제한 없음) 입력할 수 있으며 아래의 기능을 구현한다.

- ✧ 검색어는 최대 255자까지 입력할 수 있다.

- ✧ 사용자가 heaven을 입력하면 heaven이 들어간 장, 절, 위치를 출력한다.

- ✧ 사용자가 heaven earth를 입력하면 하나의 절 안에서 heaven과 earth의 순서에 관계없이 둘 다 있는 장, 절을 출력한다.

- ✧ 사용자가 "heaven earth"를 입력하면 하나의 절 안에서 heaven과 earth가 둘 다 있으며 heaven 다음에는 바로 earth가 나오는 장, 절을 출력한다.

- ✧ 사용자가 "heaven \* earth"를 입력하면 하나의 절 안에서 heaven과 earth 사이에 하나 이상의 다른 단어가 존재하는 장, 절을 출력한다.

'\*'을 사용할 때는 반드시 왼쪽과 오른쪽에 키워드가 나와야 한다. 또한 "heaven \* earth \* to"와 같이 키워드는 여러 개 사용 가능하다.

- 전송하는 구절에는 키워드를 알아볼 수 있도록 대괄호 "[ ]"를 붙인다.

- ✧ 한 개 또는 여러 개 단어 검색 (ex. God, created god)의 경우 키워드 각각에 대해 괄호 “[ ]”를 붙인다.
- ✧ 순서가 있는 여러 개의 단어 검색 (ex. “god created”, “created \* god”)의 경우 구절 안에서 처음 시작하는 키워드 앞에 대괄호 “[”를 마지막 끝나는 키워드 뒤에 대괄호 “]”를 붙인다.
- Client로 전송하는 결과는 성경 이름:장:절:해당 구절 형식에 맞춰 문자열로 전송하며 각 문자열 마지막에는 개행문자 한 개를 붙인다.
- 마지막 구절을 전송하는 경우 추가로 개행문자를 한 개 더 붙여 전송이 끝났음을 client에게 알려준다. (개행문자가 두 개인 경우 전송 완료를 의미한다.)

- **Client**

- ✓ Client 실행 시 접속할 server의 IP, Port 번호를 argument로 입력받는다.
- ✓ Server IP와 Port 번호를 이용하여 Server에 접속하며 접속이 완료된 경우 “> ” 프롬프트를 띄우고 standard input을 받을 수 있도록 대기한다.
- ✓ 사용자로부터 키워드를 입력받아 서버에게 입력받은 키워드 마지막에 개행문자 ‘\n’을 붙여 문자열로 전송한다.
  - Server에 전달하는 문자열 형식 예시  
god\n, created god\n, “god created”\n, “created\*god”\n
  - server로부터 검색 결과를 개행문자 두개가 올 때까지 전달 받으며 받은 문자열은 화면에 출력한다.
  - “!q”를 입력할 경우 Server와의 통신을 종료한다.
    - ✧ 사용자가 프롬프트에 “!q”를 입력할 때까지 검색과 결과 출력 기능을 계속할 수 있도록 구현한다.
- Server와 Client에서 socket은 connection-oriented 방식을 기준으로 구현한다.

#### 4. Restriction

- 과제는 본인이 직접 설치한 Linux 환경에서 구현한다. (gcc version 4.7.2 이상)
- 윈도우 환경에서 Visual studio 등을 이용해 프로그램을 작성한 후, 테스트 서버에서 컴파일되지 않거나 동작하지 않는 경우 과제 제출로 인정하지 않는다.
- DataBase, Lex & Yacc는 사용할 수 없다.
- Standard C library 이외의 library는 사용하지 않는다.  
파일의 입·출력은 open(), close(), read(), write(), lseek() 등의 system call 중 필요한 것을 선택하여 사용한다.
- 단어의 대·소문자는 구별할 필요가 없으며 영문 알파벳, -(hyphen), '(apostrophe)를 제외한 문자는 단어에서 제외하여 index에 추가한다.
- 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 제출 시 압축하여 "학번.zip" 형식으로 제출한다.(예: 2011123456.zip)
- 프로그램의 설계, 구현에 관한 내용을 담은 보고서를 별도로 제출한다. Flow chart 등을 활용하여 자신의 프로그램을 설명하도록 한다. 보고서는 워드, 한글 등의 형식도 가능하나 PDF format 으로 제출하는 것을 원칙으로 한다.
- 과제는 [swe3019@csl.skku.edu](mailto:swe3019@csl.skku.edu) 로 제출하며, 메일의 제목은 아래와 같은 형식을 따른다.  
[SWE3019] 학번, 이름 (예, [SWE3019] 2011123456 홍길동)

## 6. Logistics

- 과제 제출 결과는 <http://csl.skku.edu/SWE3019S13>에서 확인할 수 있다.
- 과제 제출 시간은 메일 도착시간을 기준으로 하며, 기한 이후 24시간 내에 제출할 경우 30%, 48시간 내에 제출할 경우 60% 감점한다. 이후 제출이 불가능하며, 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

Server 실행 화면	Client 실행 화면
<pre> \$./server 210.93.0.1 13782 started server: 210.93.0.1 13782  connected: 210.93.0.2 search: created god  search: "created * god" </pre>	<pre> \$./client 210.93.0.1 13782 &gt; created god genesis:1:1: In the beginning [God] [created] the heaven and the earth. genesis:1:21: And [God] [created] great whales, and every living creature that moveth, which the waters brought forth abundantly, after their kind, and every winged fowl after his kind: and [God] saw that it was good. genesis:1:27: So [God] [created] man in his own image, in the image of [God] [created] he him; male and female [created] he them. genesis:2:3: And [God] blessed the seventh day, and sanctified it: because that in it he had rested from all his work which [God] [created] and made. ... &gt; "god created" genesis:1:1: In the beginning [God created] the heaven and the earth. genesis:1:21: And [God created] great whales, and every living creature that moveth, which the waters brought forth abundantly, after their kind, and every winged fowl after his kind: and God saw that it was good. genesis:1:27: So [God created] man in his own image, in the image of [God created] he him; male and female created he them. genesis:2:3: And God blessed the seventh day, and sanctified it: because that in it he had rested from all his work which [God created] and made. ... &gt; !q </pre>

Disconnected: 210.93.0.2

connected: 210.93.0.3

search: "created \* god"

Disconnected: 210.93.0.3

\$/client 210.93.0.1 13782

> "created \* god"

genesis:1:21: And God [created great whales, and every living creature that moveth, which the waters brought forth abundantly, after their kind, and every winged fowl after his kind: and God] saw that it was good.

genesis:1:27: So God [created man in his own image, in the image of God] created he him; male and female created he them.

genesis:2:4: These are the generations of the heavens and of the earth when they were [created, in the day that the LORD God] made the earth and the heavens;

...

> !q