알고리즘분석 설계과제 1

- VOD 서비스를 위한 관리 프로그램 설계 1차

인하 VOD(Video on Demand) 서비스 회사에서는 최대 100만 명의 회원들의 정보 및 최대 9만 개의 영화를 각각 red-black 트리를 이용하여 관리하는 프로그램을 작성하려 한다. 자세한 입출력 형태는 아래와 같다. 프로그래밍 언어는 C, C++, JAVA로 제한한다.

󰋮 초기 입력 및 자료 저장

1. 각 영화 정보가 담긴 **파일의 이름**을 먼저 입력받는다.

2. 각 회원 정보가 담긴 **파일의 이름**을 다음으로 입력받는다.

3. 각 영화의 정보는 기본적으로 다음 **세 개의 필드**로 구성되어 있다.

(1) 영화제목: 한글 20글자를 넘지 않는다. (유일함)

(2) 시청요금: 10,000원을 넘지 않는다.

(3) 시청기록: 시청회원번호로 구성된 sequence

4. 각 회원의 정보는 기본적으로 다음 **다섯 개의 필드**로 구성되어 있다.

(1) 회원번호: 유일한 8자리의 수 (10000000부터 시작)

(2) 이름: 한글 10글자를 넘지 않는다. (유일하지 않을 수 있음)

(3) 생년월일: YYYYMMDD 형태의 8자리의 정수로 구성

(4) 시청기록: 시청일자, 영화제목, 지불요금으로 구성된 sequence

- 시청일자: YYYYMMDD 형태의 8자리의 정수로 구성

- 영화제목: 한글 20글자를 넘지 않는다.

- 지불요금: 10,000원을 넘지 않는다.

(5) 거주지: 거주지는 두 양의 정수  (좌표)로 표현하고 입력 받는다(단, ).

5. **영화 정보**는 영화 입력파일의 각 줄에 “영화제목, 시청요금”의 순서로 저장되어 있다. 입력파일에는 시청기록 정보는 없다. 각 정보들은 **공백으로 구분**된다.

6. **회원 정보**는 회원 입력파일의 각 줄에 “이름, 생년월일, 거주지 , 거주지 , 최초시청일자, 최초시청영화제목, 지불요금”이 차례로 저장되어 있다. 회원번호는 입력파일에 있는 순서대로 10000000부터 생성한다. 각 정보들은 **공백으로 구분**된다.

7. 영화 정보를 저장하는 red-black 트리의 기준 키(key)는 영화제목이다. 영화제목의 사전순서에 따라 **red-black 트리**를 생성한다.

8. 회원 정보를 저장하는 red-black 트리의 기준 키(key)의 종류는 회원번호이다. 회원번호에 따라 **red-black 트리**를 생성한다. 이때, 시청기록의 지불요금은 시청한 영화제목을 영화 정보 red-black 트리에서 검색하여 저장하며, 영화 정보의 red-black 트리에 시청기록을 변경한다.

9. 위의 규칙에 위배되는 상황이 발생하거나 명시하지 않은 내용에 대해서는 적절한 처리를 한 뒤 document에 명시한다.

󰋮 질의 입력: 질의 입력은 두 가지이다. 각 질의는 대화형식으로 입력받는다.

1. 영화 정보 갱신: **기존 영화 정보 갱신(시청요금) 및 신규 영화 추가**

2. 영화 정보 검색: **각 영화별 총 수익 계산 및 시청 회원 검색**

3. 회원 정보 갱신: **기존 회원 정보 갱신(시청영화 추가) 및 신규 회원 추가**

4. 회원 정보 검색: **개별 회원 정보검색 또는 전체 회원 정보검색**

󰋮 출력: 출력은 질의 입력에 따라 구분된다.

1. 영화 정보 갱신의 경우 2가지 경우로 나뉜다.

① **시청요금 변경**: 기존 영화의 정보(시청요금)를 갱신한다. 영화제목에 따라 영화를 검색하여 새로운 시청요금으로 갱신한다. 영화가 검색되지 않는 경우 오류를 출력하고 다시 질의입력을 받는다.

② **영화 추가**: 신규영화의 제목과 시청요금을 입력한다.

2. 영화 정보 검색은 해당 영화를 시청한 시청 회원들이 지불한 총액, 즉 해당 영화를 서비스하여 얻은 총 수익금과 해당 영화를 시청한 회원들의 정보(회원번호, 이름, 생년월일, 주소지 좌표)를 출력한다.

3. 회원 정보 갱신의 경우 2가지 경우로 나뉜다.

① **추가시청**: 기존 회원의 정보(시청기록)를 갱신한다. 회원번호에 따라 회원을 검색하여 새로운 시청기록에 의해 기존 시청기록을 갱신한다. 회원이 검색되지 않는 경우 오류를 출력하고 다시 질의입력을 받는다.

② **신규가입**: 이름과 생년월일, 시청기록을 입력받아 할당되지 않은 회원번호 중 가장 작은 번호를 할당하여 저장하고 회원번호를 출력한다.

4. 회원 정보 검색은 검색할 회원번호를 입력받고 **red-black 트리**에서 탐색 후 탐색된 회원의 정보(이름, 생년월일, 주소지 좌표, 시청기록[시청일자, 시청영화제목, 지불요금])와 가입이후부터 지금까지의 총지불요금, 회원 red-black 트리에서 해당 회원이 저장된 노드의 깊이(depth: 자신을 제외하고 루트노드까지의 경로에 있는 모든 내부노드의 수)를 출력한다.

프로그램 입출력 예)

**input1.txt**

터미네이터1 1000

터미네이터2 1000

터미네이터3 2000

터미네이터4 2000

터미네이터5 3000

터미네이터6 3000

터미네이터7 4000

터미네이터8 4000

터미네이터9 5000

터미네이터10 6000

터미네이터11 7000

터미네이터12 7000

터미네이터13 8000

터미네이터14 9000

터미네이터15 10000

**input2.txt**

김인하 19850101 1123 90 20090415 터미네이터4 2000

박인하 19851231 34534 6660 20090416 터미네이터4 2000

신인하 19830505 54533 5678 20090415 터미네이터14 9000

오인하 19871231 23 2365467 20090418 터미네이터5 3000

이인하 19830105 111 2234 20090417 터미네이터1 1000

정인하 19841111 3245 123456 20090421 터미네이터7 4000

최인하 19860505 766 445 20090422 터미네이터4 2000

이인하 19830106 4346 6567 20090423 터미네이터10 6000

빨간색: 프로그램 출력 내용

파란색: 사용자 입력 내용

====================================================================

영화 정보 파일의 이름을 입력하세요: input1.txt

(red-black 트리 생성 및 유지)

회원 정보 파일의 이름을 입력하세요: input2.txt

(red-black 트리 생성 및 유지)

수행할 작업(1: 영화 정보 갱신, 2: 영화 정보 검색, 3: 회원 정보 갱신, 4: 회원 정보 검색, 5: 종료)을 입력하세요: 1

수행할 세부작업(1: 시청요금갱신, 2: 영화 추가, 3: 종료)을 선택하세요: 1

영화제목: 터미네이터4

시청요금: 1000

갱신되었습니다.

수행할 작업(1: 영화 정보 갱신, 2: 영화 정보 검색, 3: 회원 정보 갱신, 4: 회원 정보 검색, 5: 종료)을 입력하세요: 1

수행할 세부작업(1: 시청요금갱신, 2: 영화 추가, 3: 종료)을 선택하세요: 2

영화제목: 람보1

시청요금: 1000

입력되었습니다.

수행할 작업(1: 영화 정보 갱신, 2: 영화 정보 검색, 3: 회원 정보 갱신, 4: 회원 정보 검색, 5: 종료)을 입력하세요: 2

영화제목: 터미네이터4

총수익: 6000원

시청회원: 총 3명

10000000 김인하 1985년 1월 1일생 (1123, 90)

10000001 박인하 1985년 12월 31일생 (34534, 6660)

10000006 최인하 1986년 5월 5일생 (766, 445)

수행할 작업(1: 영화 정보 갱신, 2: 영화 정보 검색, 3: 회원 정보 갱신, 4: 회원 정보 검색, 5: 종료)을 입력하세요: 3

수행할 세부작업(1: 추가시청, 2: 신규가입, 3: 종료)을 선택하세요: 1

회원번호: 10000004

시청일자: 20100423

시청영화: 터미네이터4

갱신되었습니다.

수행할 작업(1: 영화 정보 갱신, 2: 영화 정보 검색, 3: 회원 정보 갱신, 4: 회원 정보 검색, 5: 종료)을 입력하세요: 3

수행할 세부작업(1: 추가시청, 2: 신규가입, 3: 종료)을 선택하세요: 2

이름: 심인하

생년월일: 19850707

주소지: 1111 1111

시청영화: 터미네이터4

시청일자(YYYYMMDD): 20100502

가입되었습니다. 회원번호는 10000008입니다.

수행할 작업(1: 영화 정보 갱신, 2: 영화 정보 검색, 3: 회원 정보 갱신, 4: 회원 정보 검색, 5: 종료)을 입력하세요: 4

(1) 회원번호: 10000004

(2) 이름: 이인하

(3) 생년월일: 1983년 1월 5일

(4) 주소지: (111, 2234)

(5) 시청기록

(a) 2009년 4월 17일, 터미네이터1, 1000원

(b) 2010년 4월 23일, 터미네이터4, 1000원

(6) 총지불요금: 2000원

(7) 노드깊이: 2

수행할 작업(1: 영화 정보 갱신, 2: 영화 정보 검색, 3: 회원 정보 갱신, 4: 회원 정보 검색, 5: 종료)을 입력하세요: 5

제출파일: 학번.zip 파일로 제출

(1) document file: 알고리즘 설명, 수행 결과 분석

(2) source code file: 컴파일 환경 및 code에 대한 comment 포함

(3) presentation file: 표지 포함 10페이지 이내의 powerpoint로 작성된 발표 자료

제출기한: 5월 16일 (제출기한 엄수)

제출방법: e-Class에 학번과 이름을 제목으로 제출