**자료구조**

**프로젝트 1 명세서**

|  |
| --- |
| 1. 프로젝트 주제 |

* 곰플레이어, 네이버 미디어 플레이어, MX 플레이어와 같은 음원 관리 프로그램의 자료관리 방법들을 분석하여 장점을 취하고, 단점을 극복할 수 있는 음원 관리 프로그램의 자료관리 시스템을 설계하고 구현한다.

|  |
| --- |
| 1. 프로젝트 목표 |

* 1. 음원들을 재생하기 위한 곡 목록 관리와 정렬이 주된 목적이다.
  2. 일반적인 음원 관리 프로그램에 요구될 수 있는 기능을 포함한다.
     + 기기에 노래 추가, 변경 및 삭제
     + 재생목록 추가, 변경 및 삭제
     + 전체 노래 목록 탐색(제목 기준)
     + 가수/앨범별 노래 목록 탐색
  3. MP3 파일에서 사용하는 ID3 태그를 참조해, 음반 출시연도와 장르별로 정렬하여 표시할 수 있는 기능을 추가 구현한다.
  4. 노래 목록은 List 자료구조로 설계하며, 각 리스트는 특정한 Primary Key를 기준으로 하여 정렬되는 Sorted List이다. 각 List의 크기는 50이며, 재생목록은 10개이다.
  5. 기기에 추가, 삭제되는 전체 노래를 관리할 List는 노래 제목의 오름차순으로 정렬되도록 구현한다. 이 때, ‘제목-가수’의 조합이 Primary Key가 된다(제목은 중복될 가능성이 있음).
  6. 가수/앨범/연도/장르별 노래 목록 탐색 시에는 해당 공통된 태그를 갖는 노래들의 Primary Key를 변경해 List를 다시 변경된Primary Key의 오름차순으로 정렬되도록 구현한다.
     + 가수/앨범/연도/장르별 List를 모두 따로 선언하여 관리하면 초기화할 때만 전체 List 정렬을 수행하므로, 탐색 모드 변경 시 발생되는 정렬 시간을 줄일 수 있다. 그러나 메모리는 1개의 List로 관리했을 때에 비해 5배의 사용량을 갖게 되므로, 노래의 개수가 많아진다면 메모리 사용량도 그에 비례해 증가하는 단점이 있다. 사용자의 사용 유형에 따라 다르지만 실제 탐색 모드 변경은 자주 일어나지 않는다고 가정하고, 본 프로젝트에서는 1개의 List로 관리하는 방식으로 구현한다.
     + 각 모드의 Primary Key는 ‘가수-제목’, ‘앨범-제목’, ‘연도-제목’, ‘장르-제목’ 이 된다.
  7. 사용자가 임의로 생성한 재생 목록 List는 사용자가 노래를 추가한 순서로 정렬되도록 구현한다. 이 때, ‘추가된 순서’가 Primary Key가 되며, 키를 변경함으로써 재생 순서를 바꾸는 등의 동작을 구현할 수 있다.
  8. List의 노래 검색 시 Primary Key를 기준으로 하는 Binary Search로 구현할 수도 있으나, 중복되는 노래 제목 및 제목에 포함된 일부만으로 검색되는 기능을 고려하여 List 전체를 탐색하는 Sequential Search로 구현한다.
  9. 프로그램 수행 간 변경되는 사항을 파일입출력으로 관리할 수 있다.

|  |
| --- |
| 1. 클래스 목록 |

|  |  |
| --- | --- |
| 클래스명 | 기능 |
| Application | 프로그램의 기능을 총괄한다. 프로그램이 실행되는 동안 사용자가 선택할 수 있는 메뉴를 출력한다. 멤버변수로 선언된 BookManager 객체의 메소드를 통해 음원을 관리하고, 파일입출력을 수행한다. |
| Music | 음원의 정보가 저장될 클래스이다. 제목, 가수, 앨범, 출시연도, 장르가 멤버변수로 선언되며, ArrayList에 정렬되는 유형에 따라 수정될 Primary Key 변수가 추가로 선언된다. 이 때, Primary Key를 기준으로 하는 비교 연산자 오버로딩을 구현하여 ArrayList의 정렬이 수행되도록 한다. |
| MusicManager | ArrayList로 선언된 음원 목록을 멤버변수로 갖고, 실질적인 음원 관리를 수행하는 클래스이다. 생성자에서 파일을 로드하여 멤버변수를 초기화하고, 음원의 추가 및 제거, 리스트를 정렬하는 기준(이름, 장르, 을 설정해 화면에 출력할 수 있다. 변경된 사항을 파일에 출력하여 저장할 수 있다. |
| ArrayList | 고정된 길이의 배열을 멤버변수로 갖는 List 클래스이다. 컴파일 전 MAXSIZE를 수정함으로써 길이를 조절할 수 있고, 본 프로젝트에서는 50으로 선언한다. Template Class로 작성하여 사용자가 원하는 자료형을 List로 관리할 수 있으며, 본 프로젝트에서는 Music을 원소로 갖는 List로 구현한다. Generic Class를 구현하기 위해 List 자료형에 종속적인 메소드를 사용하지 않으며, 자료형 내부의 연산자 오버로딩을 통해 ArrayList 내의 정렬 등의 연산이 수행되도록 한다. |

|  |
| --- |
| 1. 개발 환경 |

* Microsoft Visual Studio 2010 (C++)