2022 2학기

프로그래밍과 문제해결

Assignment #4

담당교수: 윤은영

학과: 무은재학부

학번: 20220778

이름: 표승현

POVIS ID: hyeony312

“명예서약”

“나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.”

Problem: 인디언 홀덤

1. 프로그램 기능 개요

- Structure chart

Print list

Get select

Calculate

Music list (assn4)

Generate list

reader list

Add, delete, show, sort

- 입력: 1. 메뉴값을 입력받는다.

2. music list를 읽고 linked list를 생성한다.

- 처리: 1. 곡들을 이어 linked list를 생성한다.

2. 입력값에 따라 새로운 노드를 연결하거나 해제한다..

- 출력: 1. Music list를 출력한다.

2. 곡 정보들을 출력한다.

2. 알고리즘

-Psedocode

1 Head 선언

2 readfile을 이용해 txt를 읽고 링크드리스트 생성

3 while(1) {

4 add를 입력한 경우

5 addSong을 이용하여 노드 생성 및 연결

6 show를 입력한 경우

7 printlist를 이용해 사전순으로 정렬 및 출력

8 show\_favorites를 입력한 경우

9 상위 몇 개의 음악을 추출할지 입력 받기

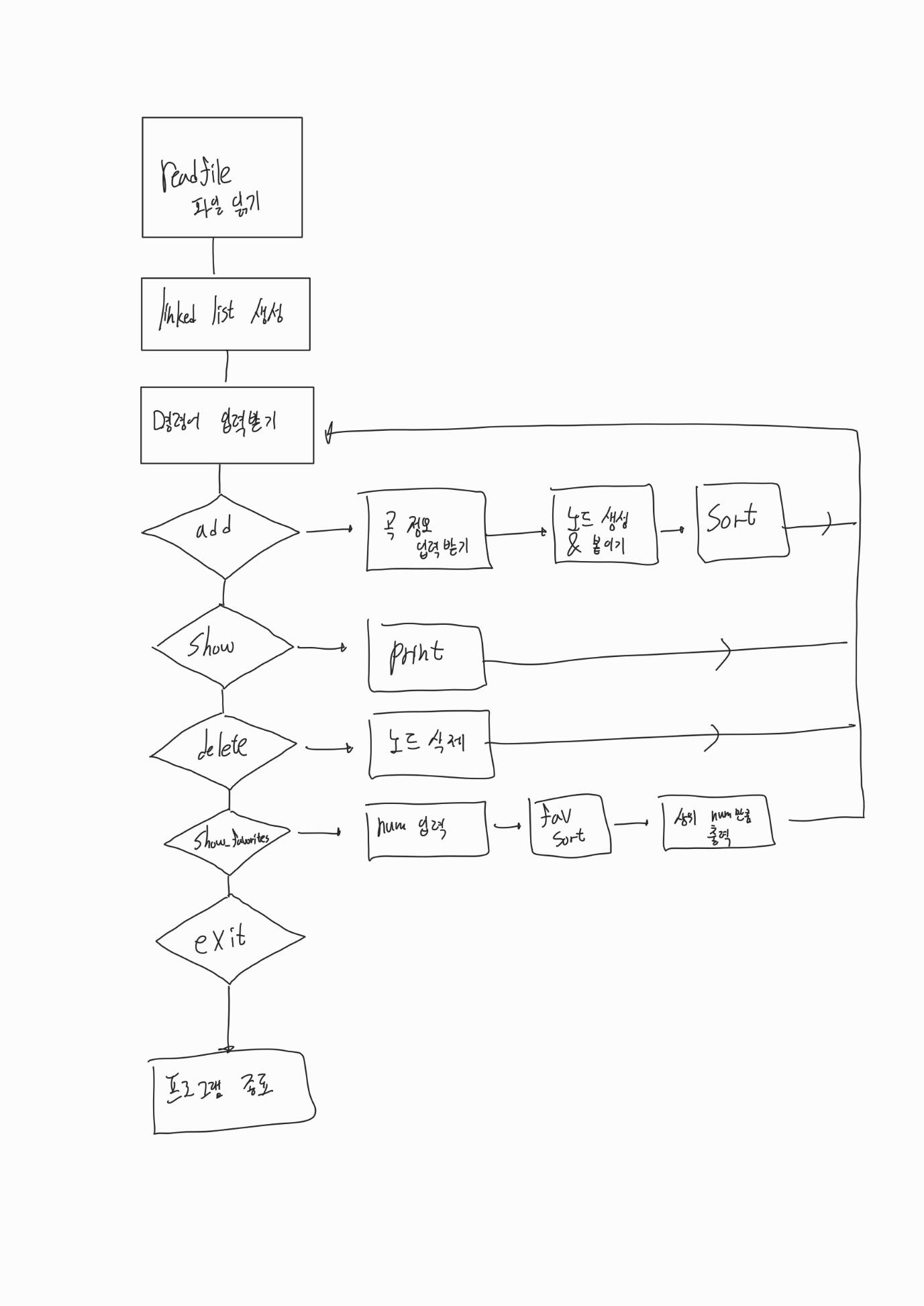
10 show\_fav를 이용해 입력 받은 개수만큼 선호도 순으로 노래 출력

11 exit을 입력한 경우

12 저장할 파일명을 입력 받기

13 입력 받은 파일명으로 링크드리스트를 텍스트 파일로 저장

14 프로그램 종료



3. 프로그램 구조 및 설명

void sortList(Head\* head); // 버블 소트를 이용하여 사전순으로 음악 정렬

void readfile(char filename[30], Head\* head); //텍스트 파일을 읽고 리스트 생성

void printList(Head\* head); // 리스트를 출력

int find(Head\* head, char title[14], int\* index); // 찾고자 하는 음악이 존재하는지 판단

int findindex(Head\* head, char title[14], int\* index); //찾고자 하는 음악의 인덱스를 반환

Node\* findLast(Head\* head); // 리스트의 마지막 노드를 찾는 함수

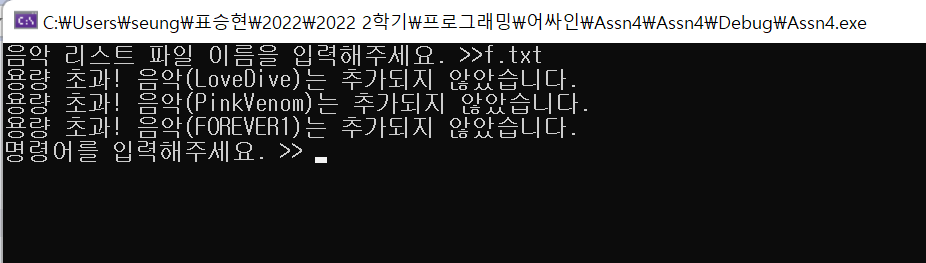
void deleteList(Head\* head); // 입력 받은 음악을 삭제

void addSong(Head\* head); // 음악을 추가

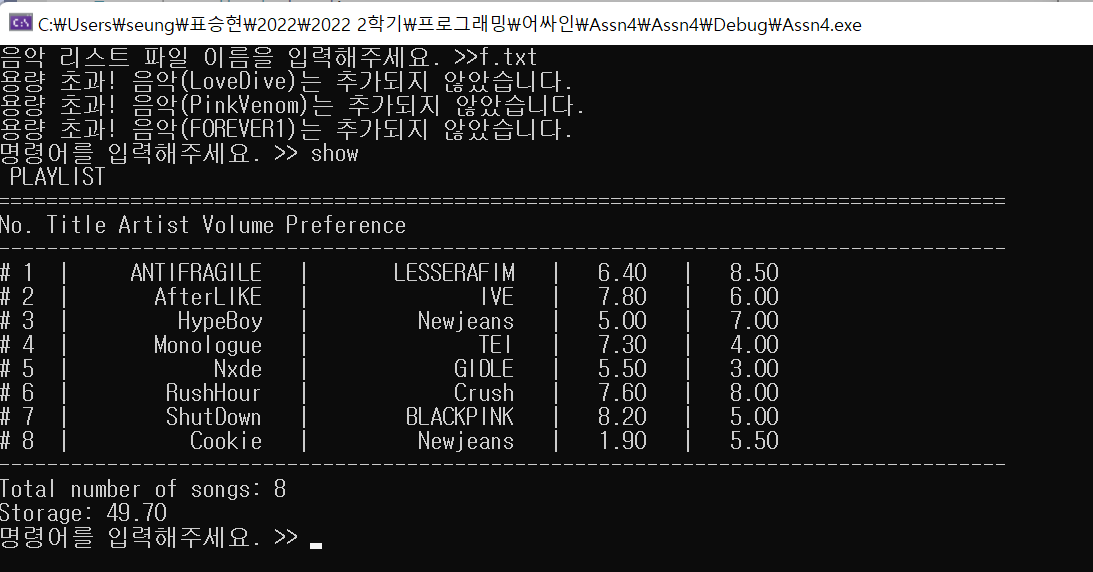
void show\_fav(Head\* head, int n); // 선호도 순으로 노래 출력

void myexit(Head\* head, char myfile[30]); // 파일을 저장하고 프로그램 종료

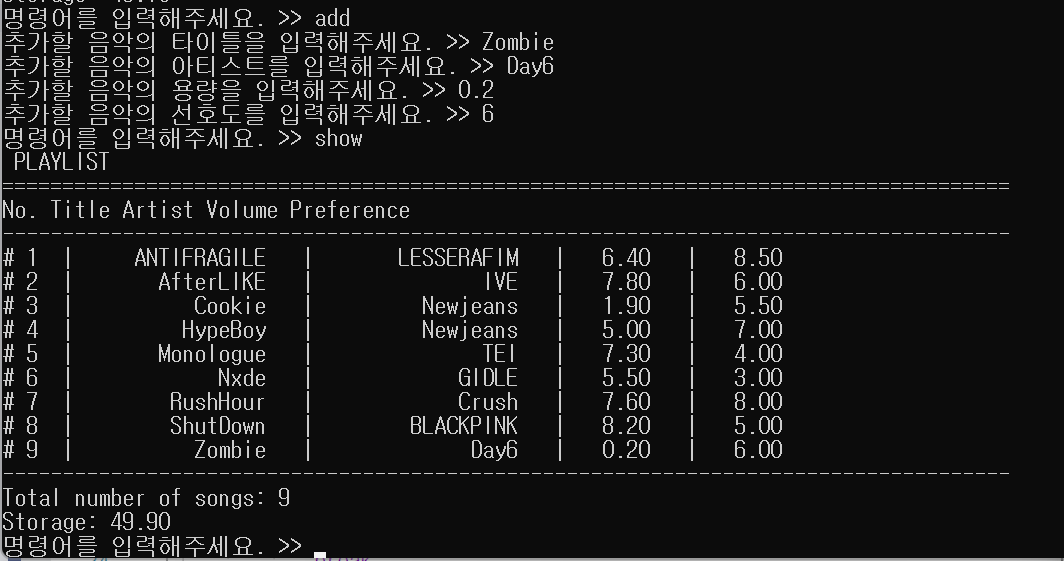
4. 프로그램 실행방법 및 예제



파일 이름을 입력받고 음악 리스트를 생성한다.



Show 명령어를 입력하면 생성된 음악 리스트를 사전순으로 출력한다.



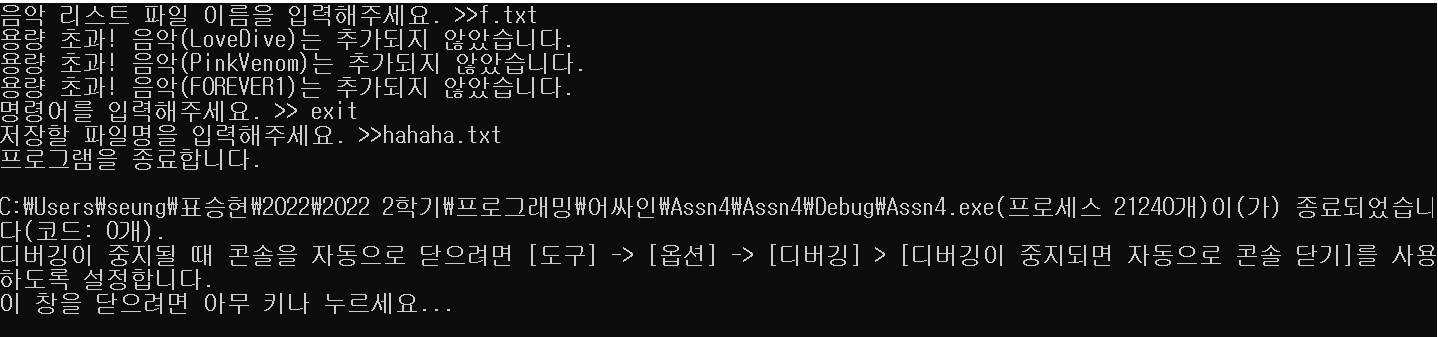
Add를 입력하고 음악 정보를 입력하면 리스트에 음악이 추가된다.



Delete를 입력하면 원하는 곡을 리스트에서 삭제할 수 있다.



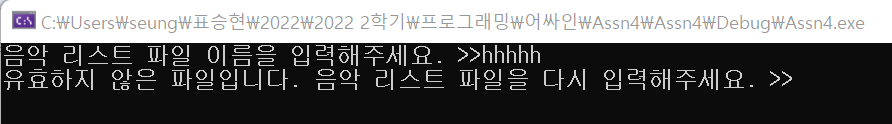
Show\_favorites를 입력하고 수를 입력하면 해당 수만큼의 노래를 선호도 순으로 출력한다.

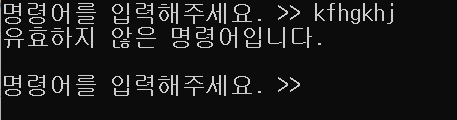


Exit를 입력하면 저장할 파일의 이름을 입력받고 음악 리스트의 곡들을 파일로 저장한다.

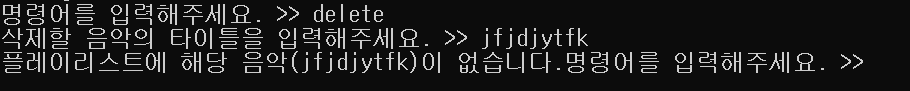


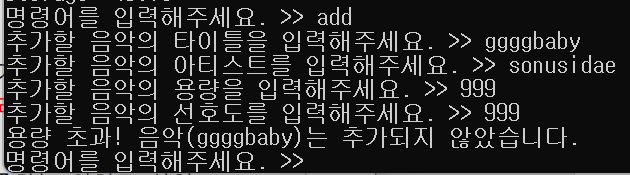
아래는 에러처리 예제들이다.

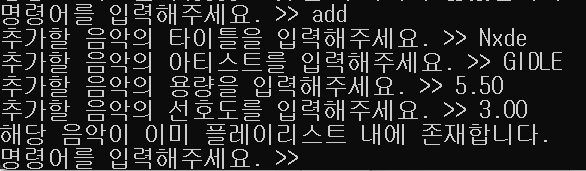












5. 토론

- 문자열을 다룰 때 문자열 관련 함수를 쓰지 않아 어려움을 겪었던 바가 있었는데, 문자열 함수를 활용하여 해결할 수 있었다.

6. 결론

링크드리스트의 개념과 활용법을 직접 적용해보며 몸으로 익힐 수 있었다.

7. 개선방향

사전순, 선호도순으로만 정리하는 것에서 더 나아가, 개개인이 만든 플레이리스트 순서를 저장할 수 있는 기능이 있으면 좋을 것 같다.

8. 헤더파일 작성

PDF에 제시된 두 헤더 파일 작성 예시에서 쓰인 문장들은 함수의 중복 선언을 피하기 위함이다. #ifdef는 해당 헤더가 정의되지 않으면 실행하라는 뜻으로 헤더가 선언되지 않았을 때에만 밑의 내용을 실행함으로 헤더를 선언한다.

#pragma once는 해당 헤더를 한 번만 실행하라는 뜻으로 똑같이 중복선언을 피하는 용도로 사용된다.