



# Object-Oriented Programming Report

## Assignment 3-2

Professor	Donggyu Sim
Department	Computer engineering
Student ID	2022202065
Name	박나림
Class (Design / Laboratory)	2 / C
Submission Date	2023. 5. 11

## Program 1

### □ 문제 설명

Binary\_Search 함수를 구현하는 문제로, 이 함수는 연결리스트의 첫번째를 가리키는 노드 포인터와 검색할 값을 인수로 가진다. 0~200 사이의 랜덤으로 생성된 정수를 가지는 16 개의 노드를 내림차순으로 정렬한 뒤 이진 탐색을 하는 형태이다. 탐색에 성공하면 해당 노드를 반환하며, 실패하면 안내 메시지를 띄운 뒤 가장 가까운 노드를 반환하도록 한다. 이러한 과정의 각 단계를 출력한다.

#### -구현 방법

먼저 랜덤 노드를 생성하기 위해 srand()함수를 이용하여 시드를 NULL 로 설정한 값으로 맞춰 랜덤한 값이 나오도록 만든다. 그리고 rand() 함수로 0~200 사이의 값을 for 문을 통해 16 번 반복하면서 해당 값을 새 노드 생성자에 전달하여 이니셜라이저로 초기화 시킨다. 이러한 노드를 연결하여 리스트로 만들고 1 단계로써 과정 결과를 출력한다.

그 다음 연결리스트를 내림차순으로 삽입정렬하기 위해 먼저 pHead 포인터에 대한 예외처리를 해주고, 정렬된 리스트를 저장할 새 노드 포인터와 현재 값으로 pHead 를 저장할 pCurr 을 만든다. while 문으로 pCurr 을 끝까지 반복하면서 다음 값을 가리키는 노드를 생성해준 뒤, 조건을 따진다. 일단 정렬 리스트가 비어 있을 경우 리스트에 현재 값을 저장한다. 현재 값이 리스트의 값보다 더 클 때는 마찬가지로 작업을 수행하며, 반대로 리스트의 값이 더 클 때는 리스트를 가리키는 노드 포인터를 새로 만들어서 노드 포인터를 끝까지 검사하면서, 해당 포인터의 다음 포인터가 nullptr 이거나 다음 포인터의 값이 현재 값보다 더 작을 때, 알맞은 위치에 값을 삽입할 수 있도록 구현한다. 정렬이 끝나면 pHead 를 리스트로 설정하고 이를 반환하며 2 단계로써 과정 결과를 출력한다.

마지막으로 값을 입력 받고 연결리스트를 이진 탐색하면서, 왼쪽과 오른쪽 노드 포인터를 생성하여 각각 시작 값과 마지막 값으로 초기화한 뒤 중간 값을 찾아서 if 문으로 중간 값과 검색 값을 비교하여 일치하는 값을 반환하도록 한다. 탐색에 실패한 경우에는 abs() 절댓값변환 함수를 이용하여 차이 값을 이용하여 가장 가까운 노드를 반환하도록 한다. 이러한 과정을 3 단계로써 결과를 출력하는 것으로 마친다.

## □ 결과 화면

```
Microsoft Visual Studio 디버그 콘솔
/* 랜덤 연결리스트 생성 */
31 149 38 137 181 160 49 132 73 115 116 50 58 39 142 63

/* 내림차순으로 연결리스트 정렬*/
181 160 149 142 137 132 116 115 73 63 58 50 49 39 38 31

/* 연결리스트 이진 탐색 */
검색할 값: 49
검색결과: 49

C:\Users\박나림\OneDrive\바탕 화면\과제\객 프\과제 3\3-2\Assign
이(가) 종료되었습니다(코드: 0x0).
```

```
Microsoft Visual Studio 디버그 콘솔
/* 랜덤 연결리스트 생성 */
102 126 177 100 46 167 19 152 4 185 165 195 176 197 89 187

/* 내림차순으로 연결리스트 정렬*/
197 195 187 185 177 176 167 165 152 126 102 100 89 46 19 4

/* 연결리스트 이진 탐색 */
검색할 값: 190
검색결과가 없으므로 가장 가까운 값을 반환합니다.
반환 값: 187

C:\Users\박나림\OneDrive\바탕 화면\과제\객 프\과제 3\3-2\Assign
이(가) 종료되었습니다(코드: 0x0).
```

랜덤 연결리스트의 생성과 정렬, 탐색을 순서대로 출력한 모습이다. 탐색 결과가 존재할 시 그대로 출력, 없으면 실패 메시지를 띄우고 가장 가까운 노드를 반환하여 출력한다.

## □ 고찰

처음에 내림차 순으로 정렬을 한 뒤, 이진 탐색으로 비교하는 과정에서는 오름차 순으로 생각하고 값을 저장하는 형태로 했다가 값이 제대로 안 나와서 시행착오를 겪었다. 다시 코드를 살펴보다가 순서가 바뀌었다는 것을 깨달아서 고쳤더니 제대로 값이 나올 수 있었다. 또한 각 함수마다 반환형을 void로 할지 Node\*로 할지에 대해서도 main 함수에서 여러 번 실행을 해보면서 알맞은 걸로 그때마다 고치는 형식으로 진행하였다. 과정을 진행하면서 어떨 때 반환이 필요할 지 더 잘 알 수 있었다.

## Program 2

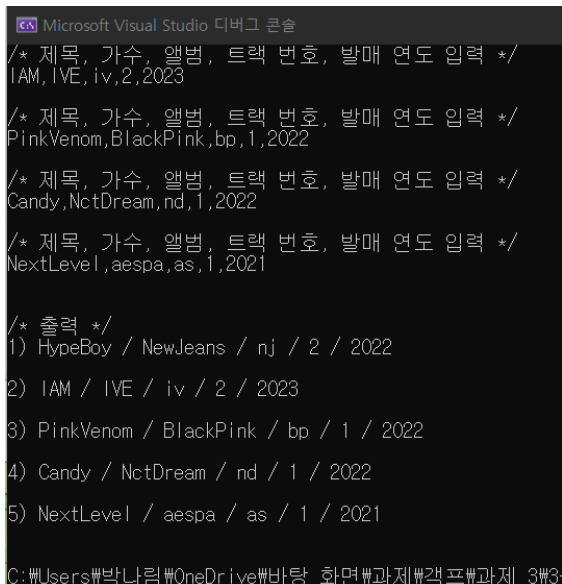
### □ 문제 설명

myMusic 클래스를 만들어서 private 멤버 변수로 음악 제목, 가수 이름, 앨범 제목, 트랙 번호, 발매 연도를 가지며, public 멤버 변수로 각각의 set, get 함수들을 가지도록 구현하는 문제이다.

#### -구현 방법

클래스에서 각 set 함수들은 문자열인 경우 깊은 복사를 위해 동적할당을 시켜준다. 이때 이미 존재하는 경우엔 먼저 삭제시키고 진행한다. get 함수들은 해당 멤버 변수들을 반환하여 main 함수에서도 사용할 수 있게 한다. 그래서 main 함수에서는 클래스 객체 배열을 사용자가 원하는 크기만큼 동적할당하고, 크기만큼 for 문으로 반복시키는 동안 제목, 가수, 앨범, 트랙 번호, 발매 연도를 strtok()을 이용하여 콤마 구분으로 입력 받아서 set 함수에 저장한다. 이때 정수는 atoi()로 변환한다. 해당 결과를 확인하기 위해 for 문으로 배열 끝까지 반복하면서 값을 가져오는 get 함수로 각 값을 출력하도록 한다.

### □ 결과 화면



```
Microsoft Visual Studio 디버그 콘솔
/* 제목, 가수, 앨범, 트랙 번호, 발매 연도 입력 */
IAM, IVE, iv, 2, 2023

/* 제목, 가수, 앨범, 트랙 번호, 발매 연도 입력 */
PinkVenom, BlackPink, bp, 1, 2022

/* 제목, 가수, 앨범, 트랙 번호, 발매 연도 입력 */
Candy, NctDream, nd, 1, 2022

/* 제목, 가수, 앨범, 트랙 번호, 발매 연도 입력 */
NextLevel, aespa, as, 1, 2021

/* 출력 */
1) HypeBoy / NewJeans / nj / 2 / 2022
2) IAM / IVE / iv / 2 / 2023
3) PinkVenom / BlackPink / bp / 1 / 2022
4) Candy / NctDream / nd / 1 / 2022
5) NextLevel / aespa / as / 1 / 2021

C:\Users\박나림\OneDrive\바탕 화면\과제\객포\과제 3\3-
```

각 정보들을 입력 받아서 set()에 전달하면, 해당 클래스 객체 배열에 저장이 된다.

## □ 고찰

이제 get 과 set 에 대한 역할을 이해해서 클래스는 다른 문제들보다 비교적 쉽게 구현했는데, main 함수에서 그 함수들을 불러올 때 헛갈린 점들이 있었다. 단순히 객체에서 함수를 호출할 때는 '->'포인터를 사용하여 불러왔었는데, 인덱스 값이 있는 배열 값으로써 함수를 호출할 때는 '.'으로 불러와야 한다는 것을 이번에 문제를 풀면서 확실히 알게 되었다. 둘이 역할이 다르다는 점을 이해해서 도움이 될 수 있었던 문제였다. 또한 콤마로 구분하는 부분에서 제일 예러가 많이 나왔었는데, 입력할 때 콤마를 쓰고 공백까지 같이 넣어버려서 런타임 에러나 액세스 위반 등이 발생했었던 것이었다. 그래서 콤마로만 입력했더니 정상적으로 출력돼서 해결할 수 있었다.

## Program 3

### □ 문제 설명

2 번 문제에서 클래스만 구현했던 것과 달리, 이번에는 그러한 음악 관리 클래스를 원형 연결리스트로 구현하는 문제이다. 이를 위해 MyMusicManagementList, ArtistNode, SongNode 3 개의 클래스를 만들어야 하는데, 첫번째 클래스는 "Musix\_info.txt"파일을 읽어서 문자열들을 저장한다. 두번째 클래스는 그 문자열 중 가수의 이름들에 대한 원형 리스트를 나타내며, 파일에서 새 가수의 이름이 나올 때마다 새 노드가 생성된다. 마지막 클래스는 각 가수의 노래 목록을 알파벳 순으로 저장하는 노드를 나타낸다. 이때 노래 값은 모두 소문자로 변형하여 저장하여, 출력도 변환된 값을 출력한다.

#### -구현 방법

먼저 첫번째 클래스에서 파일 정보들을 입력스트림 객체를 생성하여 불러온다. 이때 첫 줄은 목록 제목이므로 따로 빼내어 분리 저장한 뒤 그 다음 내용부터 시작하여 파일

끝 도달 전까지 반복하면서 strtok 을 사용해 '/'기호로 구분하여 저장한다. ArtistNode 형 이전, 현재 포인터를 생성하여 각각 nullptr, 헤드값포인터로 설정한뒤 끝까지 돌면서 가수 이름을 strcmp 로 비교한다. 이때 이름이 다르면 새로 노드를 생성하여 연결하고, 이름이 같은 경우 하위 노드로 노래 제목을 추가한다. 그 다음 두번째 클래스에서는 새로운 노래를 노드로 추가하는 과정을 구현한다. 알파벳순으로 저장하기 위해, 마찬가지로 strcmp 로 비교하여 알맞은 위치에 추가하도록 한다. 마지막 클래스는 각 멤버 변수들을 생성자에서 초기화하는 과정만 구현한다.

## □ 결과 화면

(미완성)

## □ 고찰

이번 문제에서는 파일로 불러온 내용들을 바로 노드로 추가하려다 보니 시행착오가 많았던 것 같다. 또한 노래 제목을 정렬하고 변환까지 하여 하위 노드로 추가하는 원리를 아직 잘 모르겠어서, 다 구현하지를 못하였다. 현재까지 한 부분에서는 포인터에 관한 예러가 발생하였는데, 각 클래스의 멤버 변수들을 중복하여 쓰면서 타입이 안 맞는 문제가 발생한 것 같다. 이 문제에 대해서는 연결리스트에 대해 더 공부를 해야될 것 같다.

## Program 4

### □ 문제 설명

항과 다항식 각각의 클래스를 만들어서 연결리스트로 구현한 뒤 두 다항식을 더하고 빼는 연산을 수행하는 프로그램을 만드는 문제이다. 각 노드는 순서대로 다항식의 항을 나타내며, 항의 계수와 지수, next 포인터를 가지고 있다. 덧셈 뺄셈 함수는 다항식 클래스 객체를 매개변수로 받아서 수행해야 한다. 이러한 과정의 결과는 내림차순으로 저장되어야 하며, 이때 사용자는 같은 차수는 입력하지 않는다고 가정한다.

#### -구현 방법

먼저 각 항들의 정보들을 저장하는 Term 클래스에서는 각각의 set 함수로는 변수를 설정하고, get 함수에서는 반환하도록 만든다. 다항식으로 연결리스트를 만드는 Polynomial 클래스에서는 각 멤버함수들을 구현한다. 먼저 첫번째로 삽입 함수는 항 노드를 전달 받아서 처음 들어온 경우엔 헤드 포인터를 해당 값으로 설정하고 바로 끝낸다. 그 이후부터는 해당 노드의 지수값을 불러와서 원래 가지고 있던 연결리스트의 지수값과 비교한다. 그래서 더 작을 때만 끝에 연결하고, 아니면 앞에 연결하는 식으로 구현한다. 덧셈과 뺄셈 함수에서는 결과를 저장할 리스트를 새로 만들고, 첫번째 두번째 다항식 모두 항이 존재할 때 while 문을 돌면서 지수값이 동일할 때만 차수를 계산한다. 그리고 새로운 임시 노드를 생성하여 set 함수로 각각 계수와 지수 값을 설정한 뒤 이 노드를 결과 리스트 객체로써 삽입 함수에 전달한다. 그 외 경우는 한쪽이 남을 때, 더 클 때 해당 다항식의 정보를 추가하며, 마지막으로 다시 객체로써 출력함수를 호출한다. 출력함수에서는 현재 포인터를 새로 생성하여 헤드 포인터 값으로 맞춘 뒤, 끝까지 while 문으로 돌면서 계수나 지수 조건에 따라 적절한 형태로 출력한다.

main 함수 내에서는 사용자로부터 다항식의 항 개수를 먼저 입력 받고, 그만큼 for 문을 돌면서 계수와 지수 정보를 입력 받아 set 함수로 설정하고 추가한다. 이 과정을 반복하여 만들어진 두 다항식의 덧셈과 뺄셈 연산을 수행한 뒤 총 출력하는 형식으로 구현한다.

## □ 결과 화면

```
Microsoft Visual Studio 디버그 콘솔
첫번째 다항식의 항 개수: 3
계수: 3
지수: 3
계수: 1
지수: 2
계수: 1
지수: 0
첫번째 다항식:  $3X^3 + 1X^2 + 1$ 

두번째 다항식의 항 개수: 4
계수: 2
지수: 3
계수: 1
지수: 2
계수: 1
지수: 1
계수: 1
지수: 0
두번째 다항식:  $2X^3 + 1X^2 + 1X^1 + 1$ 

/* 덧셈 */
 $5X^3 + 2X^2 + 1X^1 + 2$ 

/* 뺄셈 */
 $1X^3 + -1X^1$ 
```

사용자로부터 다항식의 정보들을 입력 받아 리스트로 만든 값을 출력한 뒤 덧셈과 뺄셈 연산을 하여 각각 다시 출력한 모습이다.

## □ 고찰

덧셈과 뺄셈 함수에서 매개변수를 하나만 받는 형태로 구성해야 하다 보니, 처음에는 바로 방법이 안 떠올라서 시행착오를 겪었다. 생각나는 방법은 매개변수를 2 개로 각각 두 다항식을 받는 것이었는데, 여러 방법을 시도해보다가 main 함수 내에서 첫번째 다항식 객체에 덧셈 함수를 호출하여 두번째 다항식을 전달하면 되는 것임을 깨달았다. 하지만 입력 조건을 맞추기가 어려워서 이 부분은 아직 완벽히 구현을 못했다. 문자열에서 변수를 각각 다른 조건으로 분리하여 저장하는 방법에 대해 더 공부가 필요한 것 같다.