

# 자료구조 실습02

## 리스트 구현

2014년 1학기

미디어소프트웨어학과  
민경하

# 할 것

1. 리스트 클래스 정의
2. 리스트 멤버 함수 구현
3. 리스트 사용

# 예제 파일(StdList.txt)

- 30명
  - 학번
  - 이름
  - 학과
  - 사는 곳
  - 평균 평점



학번	이름	학과	사는 곳	평균 평점
201011087	김상운	SOFT	서울	3.61
201011088	김세곤	SOFT	인천	4.15
201011091	김영은	SOFT	분당	4.35
201011094	김주형	SOFT	일산	4.08
201011096	김현실	SOFT	서울	3.81
201011097	김혜진	COMP	일산	3.54
201011101	백승호	COMP	서울	3.77
201011102	변소정	COMP	평택	3.93
201011111	오경희	COMP	서울	4.24
201011120	이상운	COMP	서울	4.11
201112087	김세운	BIO	서울	4.35
201112088	김영곤	BIO	일산	4.50
201112091	김주은	BIO	서울	4.05
201112094	김영형	BIO	인천	3.82
201112096	김혜실	BIO	분당	3.78
201114097	김승진	CHEM	광주	3.55
201114011	백소호	CHEM	남양주	3.13
201114102	변경정	CHEM	인천	3.36
201114111	오상희	CHEM	서울	2.99
201114120	이호운	CHEM	부천	3.11
201211087	김세경	ENGR	화정	3.02
201211088	김영운	ENGR	인천	4.18
201211091	김주곤	ENGR	판교	3.93
201211094	김영은	ENGR	파주	3.79
201211096	김혜형	ENGR	서울	4.25
201221097	김승실	GAME	서울	4.37
201221101	백소진	GAME	서울	4.02
201221102	변경호	GAME	평택	3.16
201221111	오상정	GAME	인천	3.84
201221120	이호희	GAME	이천	4.21

# 1. 리스트 클래스 정의

- 각 학생의 정보를 저장하는 “학생정보” 클래스 정의

```
class StdInfo {  
    int id;                // 학번  
    char name[8];          // 이름  
    char dept[8];          // 학과  
    char city[8];          // 사는 곳  
    float gpa;             // 평점  
};
```

# 1. 리스트 클래스 정의

- “학생 정보” 클래스에서 수행할 연산 정의 (멤버 함수)

```
class StdInfo {  
    int id;  
    char name[8];  
    char dept[8];  
    char city[8];  
    float gpa;  
public:  
    void build ( int _id, char *_name, char *_dept, char *_city, float _gpa );  
};
```

# 1. 리스트 클래스 정의

- 학생 리스트를 정의하는 “학생 리스트” 클래스 정의

```
class StdList {  
    int n;           // List의 길이  
    int cnt;         // List에 저장된 학생 수  
    StdInfo *list;   // 학생 정보에 대한 동적 배열  
};
```

# 1. 리스트 클래스 정의

- “학생 리스트” 클래스에서 수행할 연산 정의 (멤버 함수)

```
class StdList {  
    int n;           // List의 길이  
    int cnt;         // List에 저장된 학생 수  
    StdInfo *list;   // 학생 정보에 대한 동적 배열  
  
public:  
    void create ( int _n );  
    void insert ( StdInfo cStd );  
    void remove( char dname[8] );  
    void search ( char sname[8] );  
    void print ( );  
    int count ( );  
};
```

## 2. 리스트 멤버 함수 구현

- build ( )
  - 학생의 정보들을 전달받아서 StdInfo class의 멤버를 생성한다.

```
void build ( int _id, char* _name, char* _dept, char* _city, float _gpa )  
{  
  
}
```



## 2. 리스트 멤버 함수 구현

- create ( )
  - 학생 리스트에서 원소의 수가 n인 학생 정보를 생성한다.
  - cnt를 0으로 설정한다.

```
void create ( int _n )  
{  
  
}
```

## 2. 리스트 멤버 함수 구현

- insert ( )
  - cStd를 리스트의 적당한 위치에 삽입한다.
  - 리스트를 이름순으로 정렬된 상태를 유지한다.

```
void insert ( StdInfo cStd )  
{  
  
}
```

## 2. 리스트 멤버 함수 구현

- remove ( )
  - 리스트에서 dname과 같은 이름을 갖는 원소를 삭제한다.
  - 삭제할 이름이 없는 경우에는 삭제하지 않았음을 출력한다.

```
void remove( char dname[8] )  
{  
  
}
```

## 2. 리스트 멤버 함수 구현

- search ( )
  - 리스트에서 sname과 같은 이름을 갖는 원소를 찾아서 학번, 이름, 학과, 사는 곳, 평점을 출력한다.
  - 검색하는 이름이 없는 경우에는 찾는 원소가 없음을 출력한다.

```
void search( char sname[8] )  
{  
  
}
```

## 2. 리스트 멤버 함수 구현

- print ( )
  - 리스트에 있는 모든 원소를 처음부터 이름, 학번, 학과, 사는 곳, 평점 순으로 출력한다.

```
void print ( )  
{  
  
}
```

## 2. 리스트 멤버 함수 구현

- count ( )
  - 리스트에 있는 원소의 개수를 출력한다.

```
int count ( )  
{  
  
}
```

### 3. 리스트 사용

- 리스트를 사용하기 위해서 다음의 세 파일을 생성함.
  - main.cpp
  - list.h
  - list.cpp

### 3. 리스트 사용

- main.cpp
  - 학생 리스트 선언
  - 파일로부터 학생 정보를 한 줄씩 읽어서 학생 리스트에 추가
  - 학생 리스트에 대한 간단한 연산 수행



# 3. 리스트 사용

- main.cpp

```
#include "list.h"
// 선언 생략
void main ( )
{
    int i;
    int nStds;
    StdList cStdList;
    StdInfo tStd;
    int tid;
    char tname[8];
    char tdept[8];
    char tcity[8];
    float tgpa;

    ifstream fcin("StdList.txt");
    nStds = count the number of elements in the file // fcin >> nStds;
    cStdList.create ( nStds );

    for ( i = 0; i < nStds; i++ ) {
        fcin >> tid >> tname >> tdept >> tcity >> tgpa;
        tStd.build ( tid, tname, tdept, tcity, tgpa );
        cStdList.insert ( tStd );
    }
    fcin.close();

    //      계속
}
```

# 3. 리스트 사용

- main.cpp

```
#include "list.h"
// 선언 생략
void main ( )
{
    //      계속

    cStdList.print ( );

    cStdList.search ( "이상윤" );
    cStdList.search ( "현수아" );
    cStdList.count ( );
    cStdList.delete ( "백승호" );
    cStdList.delete ( "백승호" );
    cStdList.count ( );
*/
}
```

### 3. 리스트 사용

- list.h
  - 학생 정보 및 학생 리스트에 대한 class 정의  
학생 정보 및 학생 리스트에 대한 멤버 함수  
선언

```
//-----
#ifndef _LIST_H_
#define _LIST_H_
//-----

#include <iostream>
Using namespace std;
class StdInfo {
    int id;
    char name[8];
    char dept[8];
    char city[8];
    float gpa;
public:
    void build ( int _id, char *_name, char *_dept, char *_city, float _gpa );
};

class StdList {
    int n;//List의 길이
    int cnt;//List에 저장된 학생 수
    StdInfo *list;//학생 정보에 대한 동적 배열

public:
    void create ( int _n );
    void insert ( StdInfo cStd );
    void remove ( char dname[8] );
    void search ( char sname[8] );
    void print ( );
    int count ( );
};

//-----
#endif
//-----
```

### 3. 리스트 사용

- list.cpp
  - 학생 정보 및 학생 리스트에 대한 멤버 함수 정의

```
#include "list.h"

void StdInfo::build ( int _id, char *_name, char *_dept, char *_city, float _gpa )
{
}

void StdList::create ( int _n )
{
}

void StdList::insert ( StdInfo cStd )
{
}

void StdList::remove ( char *dname )
{
}

void StdList::search ( char *sname )
{
}

void StdList::print ( )
{
}

int StdList::count ( )
{
return 0;
}
```