

과제2 소개

-메모 프로그램 만들기-

2019.04.15

전자공학프로그래밍

실습 진행 계획

- 2019-1학기 전자공학프로그래밍 실습 진행 계획

순서	내용	비고
실습1	Visual Studio 설치 및 사용법	2주차
실습2	함수와 배열을 이용한 실습	5주차 (과제1)
실습3	포인터와 함수를 활용한 실습	7주차 (과제2)
실습4	C 프로젝트	9주차 (프로젝트1)
실습5	JAVA 개발 환경 설명 및 클래스를 이용한 프로그래밍	10주차
실습6	상속, 인터페이스, 패키지를 활용한 JAVA 실습	12주차 (과제3)
실습7	JAVA 프로젝트	14주차 (프로젝트2)

과제 및 프로젝트 출제 계획

- 2019년도 1학기 과제 및 프로젝트 출제 계획
 - 과제 비율** : 전체 평가의 **30%** (과제 3 + 프로젝트 2)
 - 과제 별 평가 비율

	과제1	과제2	과제3
Language	C	C	JAVA
Scoring	5점	5점	5점

- 프로젝트 별 평가 비율

	프로젝트1	프로젝트2
Language	C	JAVA
Scoring	7점	8점

과제 2. 메모 프로그램

- **메모 프로그램**
프로그램 요구사항
- **메모 프로그램 구현**
프로그램 구조
구현 사항
실행 화면
- **과제 채점 기준** 소개

메모 프로그램

- 프로그램 **요구사항**

- 프로그램은 최대 **100**개의 메모를 저장할 수 있다.
- 하나의 메모는 하나의 텍스트와 여러 개의 해시태그를 저장할 수 있다.
- 하나의 메모에는 최소 **0**개, 최대 **100**개의 해시태그를 저장할 수 있다.
- 메모의 텍스트 길이는 최대 **200**글자, 해시태그의 길이는 최대 **30**글자이다.
- 프로그램은 다음 **5**가지 기능을 수행할 수 있다.
 - ▶ 메모 생성
 - 사용자에게 메모와 해시태그를 입력 받아 저장한다.
 - ▶ 모든 메모 출력
 - 저장하고 있는 모든 메모를 출력한다.
 - ▶ 메모 검색
 - 사용자에게 해시태그를 입력 받아 해당하는 해시태그를 가지는 모든 메모를 출력한다.
 - ▶ 메모 삭제
 - 사용자에게 번호를 입력 받아 해당 번호의 메모를 삭제한다.
 - ▶ 종료
 - 프로그램을 종료한다.
- 예외처리는 고려하지 않는다.

메모 프로그램 구현

- 프로그램 구조

```
#define MAX_COUNT_MEMO 100
#define MAX_LENGTH_TEXT 200
#define MAX_COUNT_HASHTAG 100
#define MAX_LENGTH_HASHTAG 30

typedef struct {
    char text[MAX_LENGTH_TEXT+1];
    char hashtag[MAX_COUNT_HASHTAG][MAX_LENGTH_HASHTAG+1];
    int nHashtag;
}MEMO;

void printMenu();
void insertMemo(MEMO *memo, int *nMemo);
void printAllMemo(MEMO *memo, int nMemo);
void searchByHashtag(MEMO *memo, int nMemo);
void deleteMemo(MEMO *memo, int *nMemo);
```

- 매크로

- ▶ 메모와 해시태그의 최대 길이 및 최대 개수 정의

메모 프로그램 구현

- 프로그램 구조

```
#define MAX_COUNT_MEMO 100
#define MAX_LENGTH_TEXT 200
#define MAX_COUNT_HASHTAG 100
#define MAX_LENGTH_HASHTAG 30

typedef struct {
    char text[MAX_LENGTH_TEXT+1];
    char hashtag[MAX_COUNT_HASHTAG][MAX_LENGTH_HASHTAG+1];
    int nHashtag;
}MEMO;

void printMenu();
void insertMemo(MEMO *memo, int *nMemo);
void printAllMemo(MEMO *memo, int nMemo);
void searchByHashtag(MEMO *memo, int nMemo);
void deleteMemo(MEMO *memo, int *nMemo);
```

- MEMO 구조체

- ▶ 텍스트, 해시태그, 저장된 해시태그의 개수

메모 프로그램 구현

Page. 8

- 프로그램 구조

```
#define MAX_COUNT_MEMO 100
#define MAX_LENGTH_TEXT 200
#define MAX_COUNT_HASHTAG 100
#define MAX_LENGTH_HASHTAG 30

typedef struct {
    char text[MAX_LENGTH_TEXT+1];
    char hashtag[MAX_COUNT_HASHTAG][MAX_LENGTH_HASHTAG+1];
    int nHashtag;
}MEMO;

void printMenu();
void insertMemo(MEMO *memo, int *nMemo);
void printAllMemo(MEMO *memo, int nMemo);
void searchByHashtag(MEMO *memo, int nMemo);
void deleteMemo(MEMO *memo, int *nMemo);
```

- 함수

- ▶ 선택지를 출력하는 함수
- ▶ 메모 생성, 모든 메모 출력, 메모 검색, 메모 삭제 기능을 수행하는 함수

메모 프로그램 구현

- 구현 사항
 - 문제 : 요구사항과 제약사항에 따라 메모 프로그램을 구현하라.
 - 제약사항
 - ▶ 메인 함수, 함수 정의, 매크로, 구조체를 수정하지 않는다.
 - ▶ 전역 변수와 **goto**구문은 사용하지 않는다.
 - ▶ 다음 페이지의 함수 기능 명세의 내용에 따라 다음 4가지 함수를 구현한다.
 - Void insertMemo(MEMO *memo, int *nMemo)
 - Void printAllMemo(MEMO *memo, int nMemo)
 - Void searchByHashtag(MEMO *memo, int nMemo)
 - Void deleteMemo(MEMO *memo, int *nMemo)
 - ▶ 예외처리는 고려하지 않는다.
 - Ex 1) 최대 저장 개수 이상의 메모를 저장할 경우
 - Ex 2) 최대 저장 개수 이상의 해시태그를 저장할 경우
 - Ex 3) 최대 길이 이상의 길이를 가진 텍스트, 해시태그를 저장할 경우
 - Ex 4) 메뉴 입력에 한글 혹은 영어를 입력할 경우

메모 프로그램 구현

- 구현 사항
 - 문제 : 요구사항과 제약사항에 따라 메모 프로그램을 구현하라.
 - 함수 기능 명세
 - ▶ 매개변수
 - memo : 메모를 저장한 배열의 시작 주소
 - nMemo : 현재 저장된 메모의 개수를 저장한 변수 혹은 그 변수의 포인터
 - ▶ Void insertMemo(MEMO *memo, int *nMemo)
 - 사용자에게서 텍스트와 한 줄로 된 해시태그를 입력 받는다.
 - 입력 받은 텍스트와 해시태그를 저장한다.
 - 해시태그는 #을 기준으로 분리하여 저장한다. #은 저장하지 않는다.
 - 저장한 메모를 출력한다.
 - ▶ Void printAllMemo(MEMO *memo, int nMemo)
 - 저장된 모든 메모를 출력한다.
 - 출력 양식은 실행 화면의 예시와 동일하게 한다.

메모 프로그램 구현

- 구현 사항
 - 문제 : 요구사항과 제약사항에 따라 메모 프로그램을 구현하라.
 - 함수 기능 명세
 - ▶ 매개변수
 - `memo` : 메모를 저장한 배열의 시작 주소
 - `nMemo` : 현재 저장된 메모의 개수를 저장한 변수 혹은 그 변수의 포인터\
 - ▶ `Void searchByHashtag(MEMO *memo, int nMemo)`
 - 사용자에게서 하나의 태그를 입력 받는다.(#은 포함하지 않는다.)
 - 입력 받은 태그와 일치하는 해시태그를 가진 모든 메모를 출력한다.
 - 출력 양식은 실행 화면의 예시와 동일하게 한다.
 - ▶ `Void deleteMemo(MEMO *memo, int *nMemo)`
 - 사용자에게서 메모 번호를 입력 받아 해당 메모를 지운다.(번호: 배열에 저장된 위치+1)
 - 배열에서 지워진 메모보다 뒤에 위치한 모든 메모를 한 칸 앞으로 이동시킨다.

메모 프로그램 구현

- 실행 화면 (프로그램의 실행 예시)

```
***** Menu *****
1. Insert
2. Print All Memo
3. Search by Tag
4. Delete
0. Exit Program
Choose the Item: 1

Write the Memo: 전프2 과제 예시 1
Write the Hashtag: #전프#HW2#예시1

Num Memo / Hashtag
1 전프2 과제 예시 1
  #전프 #HW2 #예시1

***** Menu *****
1. Insert
2. Print All Memo
3. Search by Tag
4. Delete
0. Exit Program
Choose the Item: 1

Write the Memo: 전프2 과제 예시 2
Write the Hashtag: #전프#과제2#HW2#예시2

Num Memo / Hashtag
2 전프2 과제 예시 2
  #전프 #과제2 #HW2 #예시2

***** Menu *****
1. Insert
2. Print All Memo
3. Search by Tag
4. Delete
0. Exit Program
Choose the Item: 1

Write the Memo: 전프2 과제 예시 3
Write the Hashtag: #전프#예시3#HW2#과제2

Num Memo / Hashtag
3 전프2 과제 예시 3
  #전프 #예시3 #HW2 #과제2
```

```
***** Menu *****
1. Insert
2. Print All Memo
3. Search by Tag
4. Delete
0. Exit Program
Choose the Item: 2

Num Memo / Hashtag
1 전프2 과제 예시 1
  #전프 #HW2 #예시1

2 전프2 과제 예시 2
  #전프 #과제2 #HW2 #예시2

3 전프2 과제 예시 3
  #전프 #예시3 #HW2 #과제2
```

```
***** Menu *****
1. Insert
2. Print All Memo
3. Search by Tag
4. Delete
0. Exit Program
Choose the Item: 3

Input Hashtag: 과제2
Num Memo / Hashtag
2 전프2 과제 예시 2
  #전프 #과제2 #HW2 #예시2

3 전프2 과제 예시 3
  #전프 #예시3 #HW2 #과제2
```

```
***** Menu *****
1. Insert
2. Print All Memo
3. Search by Tag
4. Delete
0. Exit Program
Choose the Item: 4

Input Num: 2

***** Menu *****
1. Insert
2. Print All Memo
3. Search by Tag
4. Delete
0. Exit Program
Choose the Item: 2

Num Memo / Hashtag
1 전프2 과제 예시 1
  #전프 #HW2 #예시1

2 전프2 과제 예시 3
  #전프 #예시3 #HW2 #과제2
```

과제 채점 기준

- Due Date : 5/1(수) 자정

감점	지각	5/1에서 5/2로 넘어가는 자정까지 제출	감점 X	이클래스에 업로드 해주세요. (오류시 메일로 보내주세요.)
		~ 3일 지각	(총점수) x 0.75	채점 후 25% 감점
		~ 7일 지각	(총점수) x 0.50	채점 후 50% 감점
		7일 이상 지각	0	과제 마감 시점을 기준으로 7 일 이후로는 받지 않습니다.
	코드 미제출	보고서에 코드가 있을 때	-30	구현 점수 -30
		보고서에 코드가 없을 때	구현점수 x	
	메인 함수 수정	메인 함수 조정	-5	구현 점수 -5
	cpp 파일 제출 프로젝트 제출		-5	구현 점수 -5

과제 채점 기준

Page. 14

- 평가 기준

- <구현(70%) + 보고서(30%)>

: 구현과 보고서의 평가는 기능별로 진행할 예정이며, 구현의 경우는 각 기능을 명시된 설명에 맞게 구현하고 설명하였는지, 구현하지 못하였다면 어디까지 구현하고 설명하였는지를 통해 평가할 예정입니다.

- ▶ Visual Studio 2017로 구현 권장

- 채점 환경이 Visual Studio 2017이기 때문입니다.
 - 다른 버전으로 작성할 경우 버전 차이로 인한 문제가 발생할 수 있습니다.

- ▶ 함수 기능 명세에 따라 기능 구현

- ▶ 결과 화면에 수행해 본 예시에 대한 분석 필수(보고서)

- ▶ 프로젝트 제출 x, .cpp 파일 x

- 프로젝트 제출 또는 .cpp 파일 제출의 경우 감점 예정

과제 채점 기준

Page. 15

- 평가 기준
 - 구현 부분(70%)
 - ▶ 구현한 부분에 대한 기능 평가 예정
 - 보고서 부분
 - ▶ 구현된 함수의 기능 명세 및 분석
 - 함수가 제대로 기능함을 알 수 있는 결과 화면 첨부
 - ▶ 소스코드 분석
 - 각 함수의 구현 방법에 대한 간단한 주석 작성
 - 자세할 필요는 없고 코드 분석에 도움이 될 정도로 간단하게만 작성 부탁드립니다.
 - ▶ 결과 화면에 대한 분석
 - 모든 보고서 작성은 본인이 직접 작성한 내용이어야 함
 - .c파일과 보고서 파일 제출 (프로젝트 제출x, .cpp파일 제출 x)

Q&A

AJOU UNIVERSITY
Embedded & Software Lab.

TA : **임창남**, 유동연

E-mail : **chn0714@naver.com**, dongs0125@ajou.ac.kr

Office : 원천관 **338**호 임베디드소프트웨어 연구실