

1. 서론

이 프로젝트는 우리가 c언어를 처음 접할 때부터 배운 변수, 상수, 입출력, 반복문, 조건문, 함수, 배열 등등을 포괄적으로 사용하여 TODO리스트를 만드는 프로젝트이다. 기존의 4가지(할 일 추가, 삭제, 목록보기, 종료)기능에 '5. 할 일 수정'기능을 추가하고 기존 코드를 함수화 하는 것이 이 프로젝트의 최종목표이다.

2. 요구사항

사용자 요구사항: 할 일을 입력하고, 수정하거나, 삭제하기.

기능 요구사항: 사용자에게 할 일 입력 받고 삭제하고 수정하고 종료하는 작업을 요청 받고, 요청 받은 작업에 따라 기능 수행하기.

3. 설계 및 구현

```
#define MAX_TASKS 10 // 최대 할 일 개수 = 10개
#define CHAR_NUM 100 // 문자열의 최대 길이
```

최대 할 일 수와 각 할 일 문자열의 최대 길이 에 대한 상수를 정의한다.

```
char tasks[MAX_TASKS][CHAR_NUM] = {""};
int taskCount = 0;

printf("TODO 리스트 시작!\n");
```

2차원 문자배열을 선언하고 변수초기화 후 프로그램을 시작하는 문구인 리스트시작! 출력

```
while (1)
{
    int choice = -1;
    int terminate = 0;
    int delIndex = -1;
```

While(1)을 사용하여 무한루프 진입, 입력
메뉴를 저장하는 변수, 종료를 위한 플래그,
할 일 삭제를 위한 변수 초기화

```

switch (choice)
{
case 1: // 할 일 추가
    if (taskCount < MAX_TASKS)
    { // 할 일 목록이 일의 최대 개수 미만인 경우
        printf("할 일을 입력하세요 (공백 없이 입력하세요): ");
        scanf(" %99s", tasks[taskCount]); // 사용자로부터 99자 이내의 문자열을 입력
        printf("할 일 %s 가 저장되었습니다\n\n", tasks[taskCount]);
        taskCount++; // 할 일이 추가되면 할 일 수 증가시키기
    }
    else
    {
        printf("할 일 목록이 꽉 찼습니다.\n");
    }
    break;
case 2: // 할 일 삭제
    printf("삭제할 할 일의 번호를 입력해주세요. (1부터 시작): ");
    scanf("%d", &delIndex);
}

```

몇 가지 선택지가 주어지고, 그에 대한 선택을 할 때엔 switch문이 적합하므로 스위치 조건문 사용. 사용자로부터 입력받고 그에 맞는 문구를 출력하는 실행하는 내용들을 작성

```

    if (terminate == 1) // terminate 가 1이면 즉 case4일 때
    {
        printf("종료를 선택하셨습니다. 프로그램을 종료합니다.\n");
        break;
    }
}

```

무한루프를 종료시키기 위한 옵션4

4. 테스트

(1) 할 일 입력

```

현재 할 일 수 = 0
-----
1
할 일을 입력하세요 (공백 없이 입력하세요): c언어공부
할 일 c언어공부 가 저장되었습니다

-----
메뉴를 입력해주세요.
1. 할 일 추가
2. 할 일 삭제
3. 목록 보기
4. 종료
5. 할 일 수정
현재 할 일 수 = 1

```

(2) 할 일 목록보기

```

현재 할 일 수 = 1
-----
3
할 일 목록
1. c언어 공부

-----
메뉴를 입력해주세요 .
1. 할 일 추가
2. 할 일 삭제
3. 목록 보기
4. 종료
5. 할 일 수정
현재 할 일 수 = 1

```

(3)할 일 삭제

```

5. 할 일 수정
현재 할 일 수 = 1
-----
2
삭제할 할 일의 번호를 입력해주세요 . (1부터 시작) : 1
1. c언어 공부 : 할 일을 삭제합니다 .

-----
메뉴를 입력해주세요 .
1. 할 일 추가
2. 할 일 삭제
3. 목록 보기
4. 종료
5. 할 일 수정
현재 할 일 수 = 0

```

(4)할 일 수정

```

5
수정할 할 일의 번호를 입력해주세요 . (1부터 시작) : 1
1. 현재 할 일 : c언어 공부
새로운 할 일을 입력하세요 : 과제
1. 할 일이 수정되었습니다 .

-----
메뉴를 입력해주세요 .
1. 할 일 추가
2. 할 일 삭제
3. 목록 보기
4. 종료
5. 할 일 수정
현재 할 일 수 = 1

```

(5)종료

```

수정할 할 일의 번호를 입력해주세요 . (1부터 시작) : 1
1. 현재 할 일 : c언어 공부
새로운 할 일을 입력하세요 : 과제
1. 할 일이 수정되었습니다 .

-----
메뉴를 입력해주세요 .
1. 할 일 추가
2. 할 일 삭제
3. 목록 보기
4. 종료
5. 할 일 수정
현재 할 일 수 = 1
-----
4
] 종료를 선택하셨습니다 . 프로그램을 종료합니다 .

```

(6)전체 프로그램

```

할 일 목록
1. 자퇴

-----
메뉴를 입력해주세요 .
1. 할 일 추가
2. 할 일 삭제
3. 목록 보기
4. 종료
5. 할 일 수정
현재 할 일 수 = 1

-----
5
수정할 할 일의 번호를 입력해주세요 . (1부터
시작 ) : 1
1. 현재 할 일 : 자퇴
새로운 할 일을 입력하세요 : 휴학
1. 할 일이 수정되었습니다 .

-----
메뉴를 입력해주세요 .
1. 할 일 추가
2. 할 일 삭제
3. 목록 보기
4. 종료
5. 할 일 수정
현재 할 일 수 = 1

-----
2
삭제할 할 일의 번호를 입력해주세요 . (1부터
시작 ) : 1
1. 휴학 : 할 일을 삭제합니다 .

-----
메뉴를 입력해주세요 .
1. 할 일 추가
2. 할 일 삭제
3. 목록 보기
4. 종료
5. 할 일 수정
현재 할 일 수 = 0

-----
5
수정할 할 일의 번호를 입력해주세요 . (1부터
시작 ) : 1
수정 범위가 벗어났습니다 .

-----
메뉴를 입력해주세요 .
1. 할 일 추가
2. 할 일 삭제

```

5. 결과 및 결론

솔직히 말하자면 기존코드를 함수로 구현 하는게 어려웠지만 프로그램들이 하나하나 다 잘 돌아가는 걸 보고 시간을 들인 게 헛된 수고가 아니라는 생각이 들었다. 배운 내용을 한 프로젝트에 다 담다 보니, 복습도 되고 변수 상수를 배울 때부터 함수를 배울 때 까지의 과정이 머릿속에 주마등처럼 스쳐 지나갔다. 처음 c언어를 배울 때는 할 수 있는게 이해는 커녕

있는 그대로 받아들이기도 힘들었지만 계속 하다 보니까 이해가 되고 배운 내용들이 유기적으로 흘러가는 느낌이 들어 뿌듯함을 느꼈다.