

## 1. 서론

1. 프로젝트 목적 및 배경: 7주차까지 배운 내용에 대한 실습을 위해 todo 리스트를 만드는 프로그램 작성
2. 목표: TODO 리스트 만들기

## 2. 요구사항

1. 사용자 요구사항: 사용자가 할 일을 입력, 삭제, 출력, 수정할 수 있는 프로그램
2. 기능 요구사항
  - A. 입력 기능
  - B. 삭제 기능
  - C. 출력 기능
  - D. 수정 기능

## 3. 설계 및 구현

### A. 입력

기능

```
case 1:
    printf("할 일을 입력하세요 (공백 없이 입력하세요): ");
    scanf_s("%s", tasks[taskCount], (int)sizeof(tasks[taskCount]));
    printf("할 일 \"%s\"가 저장되었습니다\n\n", tasks[taskCount]);
    taskCount++;
    break;
```

- 입력 (블록/함수에 입력되는 변수, 값들과 설명)
  - taskCount = 현재 작업 수
  - tasks = 할일 목록 저장 2차원 배열
- 결과 (블록/함수가 종료된 결과)
  - tasks에 할 일 추가
- 설명 (코드 내 작동 순서, 내용 등 추가 설명)
  - 사용자에게 삭제할 할 일의 인덱스를 입력받는다
  - 입력 받은 인덱스 -1에 있는 할 일을 배열에서 제거한다
  - 제거한 후, 뒤에 있던 할일들을 앞으로 당겨온다.

### B. 삭제 기능

```

case 2:
    // 할 일 삭제하는 코드 블록
    printf("삭제할 할 일의 번호를 입력해주세요. (1부터 시작):");
    scanf_s("%d", &delIndex);
    if (delIndex > taskCount || delIndex <= 0) {
        printf("삭제 범위가 벗어났습니다.\n");
    }
    else {
        printf("%d. %s : 할 일을 삭제합니다.\n", delIndex, tasks[delIndex - 1]);

        // 배열간 대입 (=배열에 문자 배열인 문자열의 대입) 이 불가능하기 때문에
        // 문자열 복사 함수로 삭제
        strcpy_s(tasks[delIndex - 1], sizeof(tasks[delIndex - 1]), "");

        // 특정 인덱스의 할 일 삭제 후 뒤에 있는 할 일 앞으로 옮기기
        for (int i = delIndex; i < taskCount + 1; i++) {
            strcpy_s(tasks[i - 1], sizeof(tasks[i]), tasks[i]);
        }
        taskCount -= 1;
    }
    break;

```

- 
- 입력 (블록/함수에 입력되는 변수, 값들과 설명)
  - delIndex = 삭제할 할 일의 번호 +1
  - taskCount = 현재 작업 수
  - tasks = 할일 목록 저장 2차원 배열
- 결과 (블록/함수가 종료된 결과)
  - delIndex-1의 인덱스를 갖는 값이 삭제됨
- 설명 (코드 내 작동 순서, 내용 등 추가 설명)
  - delIndex를 입력 받음
  - 삭제 범위를 벗어나면 벗어났다고 알림
  - 배열간 대입이 불가능하기 때문에 문자열 복사 함수로 삭제
  - 특정 인덱스의 할 일 삭제 후 뒤에 있는 할 일 앞으로 옮기기
  - taskCount에 1을 뺌

### C. 출력 기능

```

case 3:
    printf("할 일 목록\n");
    for (int i = 0; i < taskCount; i++) {
        printf("%d. %s\n", i + 1, tasks[i]);
    }
    printf("\n");
    break;

```

- 입력 (블록/함수에 입력되는 변수, 값들과 설명)
  - taskCount = 현재 작업 수
  - tasks = 할일 목록 저장 2차원 배열
- 결과 (블록/함수가 종료된 결과)
  - tasks에 저장된 할 일을 차례대로 출력함
- 설명 (코드 내 작동 순서, 내용 등 추가 설명)
  - 반복문을 통해 task에서 차례대로 한 줄씩 출력함

#### D. 수정 기능

```

case 5:
    //수정할 할 일의 번호를 입력 받음
    printf("수정할 할 일의 번호를 입력해주세요. (1부터 시작):");
    scanf_s("%d", &changeIndex);
    if (changeIndex > taskCount || changeIndex <= 0) { //입력받은 수정할 할 일의 번호가 정상적인 값인지 확인
        printf("수정 범위가 벗어났습니다.\n"); //입력받은 수정할 할 일의 번호가 정상적이지 않은 경우
    }
    else //입력받은 수정할 할 일의 번호가 정상적인 경우
    {
        printf("수정할 할 일을 입력하세요 (공백 없이 입력하세요): ");
        scanf_s("%s", tasks[changeIndex - 1], (int)sizeof(tasks[changeIndex - 1]));
        printf("할 일 \"%s\"가 저장되었습니다\n\n", tasks[changeIndex - 1]);
        changeIndex = -1;
    }
    break;

```

- 입력 (블록/함수에 입력되는 변수, 값들과 설명)
  - taskCount = 현재 작업 수
  - tasks = 할일 목록 저장 2차원 배열
  - changeIndex = 수정할 할 일의 번호 + 1
- 결과 (블록/함수가 종료된 결과)
  - 할 일이 수정됨
- 설명 (코드 내 작동 순서, 내용 등 추가 설명)
  - changeindex를 입력 받음

- 할 일을 입력 받고 이를 changeIndex-1인 인덱스에 덮어씌움
- changeIndex를 -1로 초기화함

#### 4. 테스트

- (테스트 : 입력에 따른 원하는 결과나 나오는지 확인하는 과정)

##### 1. 기능 별 테스트 결과: (요구사항 별 스크린샷)

###### A. 입력

```

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

1
할 일을 입력하세요 (공백 없이 입력하세요): 할 일
할 일 할 일가 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
-----

1
할 일을 입력하세요 (공백 없이 입력하세요): 밥먹기
할 일 밥먹기가 저장되었습니다

```

i.

###### B. 삭제

-----  
메뉴를 입력해주세요.

1. 할 일 추가

2. 할 일 삭제

3. 목록 보기

4. 종료

5. 할 일 수정

현재 할 일 수 = 2

-----

2

삭제할 할 일의 번호를 입력해주세요. (1부터 시작):2

2. 밥먹기 : 할 일을 삭제합니다.

i.

C. 출력

-----  
메뉴를 입력해주세요.

1. 할 일 추가

2. 할 일 삭제

3. 목록 보기

4. 종료

5. 할 일 수정

현재 할 일 수 = 2

-----

3

할 일 목록

1. 할 일

2. 밥먹기

i.

D. 수정

-----  
메뉴를 입력해주세요.

1. 할 일 추가

2. 할 일 삭제

3. 목록 보기

4. 종료

5. 할 일 수정

현재 할 일 수 = 2

-----

5

수정할 할 일의 번호를 입력해주세요. (1부터 시작):1

수정할 할 일을 입력하세요 (공백 없이 입력하세요): 씻기

할 일 씻기가 저장되었습니다

i.

E. 종료

```

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

4
종료를 선택하셨습니다. 프로그램을 종료합니다.

```

i.

#### F. 할 일이 다 참

```

-----
메뉴를 입력해주세요.
1. 할 일 추가
2. 할 일 삭제
3. 목록 보기
4. 종료
5. 할 일 수정
현재 할 일 수 = 9
-----

1
할 일을 입력하세요 (공백 없이 입력하세요): f
할 일 f가 저장되었습니다

할 일이 다 찼습니다. 프로그램을 종료합니다.

```

i.

#### 5. 결과 및 결론

1. 프로젝트 결과: todo 관리프로그램을 만들었다.
2. 느낀 점: 주석 다는 게 가장 힘든 것 같다.