

C언어 8주차 Project 보고서

행정학과

233973

김세희

[서론]

1. 프로젝트 목적 및 배경: 7주차까지 배운 내용에 대한 실습을 위해 진행함.
2. 목표: TODO 리스트 만들기

[요구사항]

1. 사용자 요구사항: 사용자가 할 일을 입력, 삭제, 수정, 출력할 수 있는 프로그램
2. 기능 요구사항
 - 1) 사용자에게 작업 요청 받기 (1.할 일 추가 2.할 일 삭제 3.목록 보기 4.종료 5. 할 일 수정)
 - 2) 요청 받은 작업에 따라 아래 기능 수행
 - 할 일 추가 입력: 사용자에게 할 일을 입력 받고 저장
 - 할 일 삭제를 입력: 인덱스를 입력 받고 해당 할 일을 삭제
 - 목록 보기 입력: 전체 할 일 목록 보여주기
 - 종료 입력: 프로그램 종료
 - 할 일 수정: 인덱스와 할 일(문자열)을 입력 받고, 해당 인덱스의 할 일 변경
 - 3) 할 일 10개로 다 찬 경우: 할 일이 다 찼다고 출력 후 프로그램 종료

[설계 및 구현]

1. 사용자에게 작업 요청 받기

```
int main() {
    printf("TODO 리스트 시작!\n");

    while (1) {
        int choice;

        printf("-----\n");
        printf("메뉴를 입력해주세요.\n");
        printf("1. 할 일 추가\n2. 할 일 삭제\n3. 목록 보기\n4. 종료\n5. 할 일 수정\n");
        printf("현재 할 일 수 = %d\n", taskCount);
        printf("-----\n");
        scanf_s("%d", &choice);
    }
}
```

1) 입력

- "scanf_s" 함수를 통해 사용자에게 숫자를 입력 받은 후 "choice" 에 저장.

2) 반환값

- Switch 함수를 통해 case 1~5의 결과가 출력

3) 결과

- 입력 값이 '1'인 경우 "addTask()" 함수 호출, '2'인 경우 "deleteTasks()" 함수 호출, '3'인 경우 할 일 목록을 조건문을 통해 출력, '4'인 경우 프로그램을 종료

```
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
-----
```

4) 설명

- While 문을 통해 무한으로 반복하는 루프를 만들어 사용자에게 입력값을 받을 수 있는 창을 띄운다.
- 사용자에게 어떤 작업을 할 것인지 입력값을 받는다.
- 입력 값에 따라 "switch" 문을 통해 해당하는 함수로 결과값을 가져온다.

2. 할 일 추가 입력: 사용자에게 할 일을 입력 받고 저장

```
void addTask() { //할 일 추가하는 함수
    if (taskCount >= MAX_TASKS) { //할 일 수가 다 찼을 경우 출력하는 조건문
        printf("할 일이 다 찼습니다.\n");
        return 0;
    }

    printf("할 일을 입력하세요 (공백 없이 입력하세요): ");
    scanf_s("%s", tasks[taskCount], (int)sizeof(tasks[taskCount])); //사용자에게 할 일
    printf("할 일 %s (이)가 저장되었습니다.\n", tasks[taskCount]);
    taskCount++; //사용자에게 할 일을 입력받은 후 수를 1개씩 올림
}
```

1) 입력

- "scanf_s" 를 통해 할 일을 사용자에게 입력받음.

2) 반환값

- "할 일이 다왔습니다". 라는 멘트 출력
- "tasks[tasksCount]" 리스트에 할 일 저장

3) 결과

- "taskCount"가 "MAX_TASKS"보다 큰 경우 "할 일이 다 왔습니다"를 출력
- 할 일을 입력후 "tasks[tasksCount]"에 저장
- "taskCount"를 +1 시킴

```
-----
1
할 일을 입력하세요 (공백 없이 입력하세요): 교수님사랑해
할 일 "교수님사랑해요" (이)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
-----
```

4) 설명

- 조건문을 통해 할 일이 10개 이상인 경우 프린트를 출력함.
- 사용자에게 할 일을 입력 받고 리스트에 저장 함.
- taskCount의 할 일을 하나씩 숫자를 올려줌

3. 할 일 삭제를 입력: 인덱스를 입력 받고 해당 할 일을 삭제

```
int deleteTask() { //할 일 삭제 함수의 시작점입니다.
{
    if (taskCount == 0) { //할 일이 0개인 경우 출력하는 조건문
        printf("할 일 목록을 수정할 것이 없습니다.");
        return 0;
    }

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

    int delIndex; //삭제 받을 항목 입력받을 변수 생성
    printf("삭제할 일의 번호를 입력해주세요 (1부터 시작): ");
    scanf_s("%d", &delIndex); //삭제 받을 번호 입력값을 delIndex 함수에 전달

    if (delIndex < 1 || delIndex > taskCount) { // delIndex 가 1보다 작거나 10보다 큰
        printf("삭제 범위가 벗어났습니다. \n");
        return;
    }

    printf("%d. %s: 할 일을 삭제합니다.\n", delIndex, tasks[delIndex - 1]);

    //삭제 받을 번호 한 일을 삭제
    for (int i = delIndex - 1; i < taskCount - 1; i++) {
        strcpy_s(tasks[i], tasks[i + 1], sizeof(CHAR_NUM));
    }

    taskCount--; //할 일 -1
}
```

1) 입력

- taskCount의 숫자를 입력받음
- 사용자에게 삭제할 일의 번호를 입력받은 후 "delIndex" 변수에 저장

2) 반환값

- 삭제 받을 번호를 삭제한 후에 리스트를 반환

3) 결과

- "taskCount" 값이 0인 경우 "할 일 목록을 수정할 것이 없습니다"를 출력
- 삭제 받은 번호의 할 일을 리스트에서 삭제 후 "taskCount" 함수에서 -1을 시킴

```

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

2
할 일 목록:
1. 공부하기
2. 밥먹기
삭제할 일의 번호를 입력해주세요 (1부터 시작): 1
1. 공부하기: 할 일을 삭제합니다.
할 일이 삭제되었습니다.
-----

```

4) 설명

- If 조건문을 통해 "taskCount"가 0인 경우 "할 일 목록을 수정할 것이 없습니다."를 출력
- 사용자에게 반복문을 통해 할 일 목록을 보여 줌
- "delIndex"라는 변수를 생성 후 사용자에게 입력값을 받아 저장
- If 조건문을 통해 "delIndex"가 1보다 작거나 10보다 큰 경우 에러를 방지하는 출력 시스템을 생성
- "delIndex" 변수를 통해 삭제 받은 번호의 리스트를 인덱스 해온후 "strcpy_s"함수를 통해 리스트를 복사하며 삭제함.
- "taskCount" 변수를 -1를 시켜줌.

4. 목록 보기 입력: 전체 할 일 목록 보여주기

```
//할 일을 출력하는 함수
void printTask(int taskCount) {
    for (int i = 0; i < taskCount; i++) {
        printf("%d. %s\n", i + 1, tasks[i]);
    }
    printf("\n");
}
```

1) 입력

- "taskCount"의 수를 입력받음

2) 반환 값

- 반환 값: 없음

3) 결과

- "tasks[]"를 전체 할 일을 출력

```
-----
3
할 일 목록
1. 밥먹기
2. 공부하기
-----
```

4) 설명

- 반복문을 통해 "taskCount"의 수 이하로 "task[]"에서 하나씩 가져와 출력하는 함수
- 위를 반복함으로써 할 일을 출력

5. 종료 입력: 프로그램 종료

```
break;
case 4:
    terminate = 1;
    break;
case 5:
    changeTask();
    break;
default:
    printf("잘못된 선택입니다. 다시 선택하세요.\n");
    break;
}
if (terminate == 1) {
    printf("종료를 선택하셨습니다. 프로그램을 종료합니다.\n");
    break;
}
```

1) 입력

- "choice" 변수를 통해 switch 함수에서 값을 받음

2) 반환값

- "terminate" 변수를 1로 반환

3) 결과

- 프로그램을 종료시킴

```
-----
4
종료를 선택하셨습니다. 프로그램을 종료합니다.
C:\Users\user\Desktop\1027C언어\x64\Debug\1027C언어.exe(프로세스 15876개)이(가) 종료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...
```

4) 설명

- 사용자에게 case 4를 입력 받은 경우 "terminate" 변수를 1로 반환시킴
- If 조건문을 통해 "terminate" 값이 1인 경우 프로그램을 종료시킴

6. 할 일 수정: 인덱스와 할 일(문자열)을 입력받고, 해당 인덱스의 할 일 변경

```
//할 일을 수정하는 함수
void changeTask() {

    int changeTask; //수정 할 일의 번호를 저장하는 변수 생성
    printf("수정할 할 일의 번호를 입력해주세요 (1부터 시작): ");
    scanf_s("%d", &changeTask); //수정 할 일의 번호를 입력받고 changeTask에 저장

    printf("수정할 내용을 입력하세요: ");
    scanf_s("%s", tasks[changeTask - 1], (int)sizeof(tasks[changeTask - 1])); //수정 할 일의 번호
    printf("%d. %s: 할 일이 수정되었습니다.\n", changeTask, tasks[changeTask - 1]);
}
```

1) 입력

- 사용자에게 "scanf_s" 함수를 통해 "changeTask" 변수 값을 입력받음
- 사용자에게 "scanf_s" 함수를 통해 수정할 일을 입력받음

2) 반환값

- 반환값: 없음

3) 결과

- 수정된 할 일 리스트를 출력

```
-----
5
수정할 할 일의 번호를 입력해주세요 (1부터 시작): 2
수정할 내용을 입력하세요: 수정할 일
2. 수정할 일: 할 일이 수정되었습니다.
```

4) 설명

- "changeTask" 변수를 생성 후 "scanf_s"함수를 통해 "changeTask" 값을 저장함.

- "scanf_s" 함수를 통해 수정 할 내용을 입력받고 미리 받아놓 "changeTask" 변수의 값을 통해 수정 할 인덱스에 수정 할 내용을 저장시킴.

7. 할 일 10개로 다 찬 경우: 할 일이 다 찼다고 출력 후 프로그램 종료

```
if (taskCount >= MAX_TASKS) { //할 일 수가 다 찼을 경우 출력하는 조건문
    printf("할 일이 다 찼습니다.\n");
    return 0;
}
```

1) 입력

- "taskCount"의 값

2) 반환값

- 반환값: 없음

3) 결과

- "taskcount"값이 MAXTASKS(10)보다 큰 경우 프로그램을 종료

```
-----
메뉴를 입력해주세요.
1. 할 일 추가
2. 할 일 삭제
3. 목록 보기
4. 종료
5. 할 일 수정
현재 할 일 수 = 10
-----
1
할 일이 다 찼습니다.
```

4) 설명

- 조건문을 통해 "taskCount"값이 10개 이상을 초과한 경우 print를 출력하며 프로그램을 종료시킴

8. 최종 테스트

```
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 목록
1. 드디어
-----
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
수정할 내용을 입력하세요; 과제끝
1. 과제끝: 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.

C:\Users\user\Desktop\1027C언어\x64\Debug\1027C언어.exe(프로세스 22544개)이(가) 종료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...
```