

C프로그래밍및실습

복합쇼핑몰

진척 보고서 #2

제출일자: 2023-12-10

제출자명: 강지효

제출자학번: 232676

1. 프로젝트 목표 (16 pt)

1) 배경 및 필요성 (14 pt)

복합쇼핑몰이란 영화 상영관, 쇼핑센터, 식당 등을 한 건물 내에 갖춘 복합 건물을 말한다. 산업이 발전하면서 사람들의 취미가 늘어났다. 다양한 취미를 하나의 건물에서 해결해주기 위한 복합쇼핑몰이 차차 늘어나며 현재는 많은 복합쇼핑몰이 있다. 더욱 복잡해진 복합쇼핑몰에는 안내자가 필요하기 마련이다. 그래서 복합쇼핑몰에 들어오자마자 모든 것을 키오스크로 간단히 해결해주는 프로그램이 필요하다.

2) 프로젝트 목표

복합쇼핑몰의 기능이 무엇이 있는지를 분석하고 고객이 원하는 취미 활동을 입력받아 위치, 가격, 예매(영화)등의 다양한 기능을 입장 할 때의 키오스크를 통해 미리 해결하는 것이 목표이다.

3) 차별점

기존에는 각각의 가게에 직접 가서 모든 것을 해결해야 했다면, 건물에 들어가자마자 키오스크를 통해 문제를 해결할 수 있으니 시간을 단축할 뿐만 아니라 위치를 알려주기 때문에 복잡한 건물 안에서 길을 잃거나 그로 인해 약속 시간에 늦을 일이 없도록 도와준다.

2. 기능 계획

1) 기능 1 (예: 유사 취향 고객 식별)

- 고객이 이곳에 온 목적을 알아내어 장소의 위치를 알려주는 기능

(1) 세부 기능 1

- 영화관의 경우 - 영화 장르를 고를 수 있고 그에 맞는 영화와 상영 시간을 제시하여

사용자에게 선택하게 하여 그 영화를 예매해주는 프로그램

- 쇼핑의 경우 - 식료품, 의류, 명품의 카테고리로 나눠 각 카테고리의 브랜드를 알려주고 그 브랜드를 선택 할 경우 위치를 알려주는 프로그램
- 식당의 경우 - 한식, 일식, 중식으로 카테고리를 나누고 이에 맞는 메뉴와 가격을 알려주고 고객에게 선택하도록 하는 프로그램.
- 운동의 경우 - 헬스, 축구, 테니스로 카테고리를 나누고 각각의 위치를 알려줌.

2) 기능 2

- 일행이 있다면 그 일행의 위치를 알려주는 기능.

1) 세부 기능 1

- 일행이 간 곳의 위치를 알려주고, 들어가 있는 사람에게 만나기 쉽게 핸드폰 문자로 일행이 들어온 것을 알림.

(사람이 들어갈 때 일행이 있는지를 물어보고, 있다면 일행의 핸드폰 번호 뒷자리 4자리를 입력하게 한다. 그리고 나중에 그 일행이 들어올 때 자신의 핸드폰 번호 뒷자리를 입력하게 될 경우 먼저 들어간 사람이 간 곳을 알려주고 먼저 들어간 사람에게는 알림이 오게 된다.)

2) 기능 3

- 주차장 이용 시 주차 비용과 주차 가능한 층 수 정보 제공.

1) 세부 기능 1

고객이 차를 가지고 왔는지를 물어보고 주차 공간 정보 제공 및 자리 예약.

-> 이렇게 하면 주차장 내에서 주차 자리가 없어 주차 자리를 찾는 시간을 줄일 수 있을 것이라고 생각.

3. 진척사항

1) 기능 구현

(1) 영화관 예매 기능

- 입력

1. 보고 싶은 영화 장르 번호 입력.
2. 보고 싶은 영화 번호 입력.
3. 원하는 영화 시간 번호 입력.
4. 시간이 맞는지 확인 하는 절차에 "예", "아니오" 입력

- 출력

1. 영화 장르 출력
2. 영화 장르에 맞는 제목 출력
3. 해당 영화의 시간표 출력
4. 시간이 맞는지 확인
5. 예를 입력 시 시간과 영화를 알려주며 함수 끝
6. 아니오를 입력 시 시간을 다시 골라 달라며 다시 시작
7. 이외의 답변을 입력 시 다시 예, 아니오를 입력하도록 유도.

- 영화관 기능을 수행하는 movie_theater 함수.

```
void movie_theater() { // 영화를 선택했을 시의 맥락이 되는 함수.
    int genre;
    char time_elemental[9][9] = { "10시20분", "11시00분", "11시40분", "12시40분", "13시30분", "14시20분", "15시20분", "16시10분", "17시00분" };
    char time_frozen_kingdom[9][9] = { "10시30분", "11시10분", "11시50분", "12시50분", "13시40분", "14시30분", "15시30분", "16시20분", "17시10분" };
    char time_minions[9][9] = { "10시40분", "11시20분", "12시00분", "12시40분", "13시40분", "14시20분", "15시20분", "16시10분", "17시00분" };
    char time_swingers_endgame[9][9] = { "10시20분", "11시00분", "11시40분", "12시40분", "13시30분", "14시20분", "15시20분", "16시10분", "17시00분" };
    char time_guardians_of_galaxy[9][9] = { "10시30분", "11시10분", "11시50분", "12시50분", "13시40분", "14시30분", "15시30분", "16시20분", "17시10분" };
    char time_batman[9][9] = { "10시40분", "11시20분", "11시50분", "12시50분", "13시40분", "14시30분", "15시30분", "16시20분", "17시10분" };
    char time_captain_america_winter_soldier[9][9] = { "10시20분", "11시00분", "11시40분", "12시40분", "13시30분", "14시20분", "15시20분", "16시10분", "17시00분" };
    char time_the_roundup[9][9] = { "10시30분", "11시10분", "11시50분", "12시50분", "13시40분", "14시30분", "15시30분", "16시20분", "17시10분" };
    char time_kingsman[9][9] = { "10시40분", "11시20분", "11시50분", "12시50분", "13시40분", "14시30분", "15시30분", "16시20분", "17시10분" };

    while (1) {
        printf("보고싶은 장르를 골라주세요\n 1. 메이메이션\n 2. 히어로\n 3. 액션\n 4. 종료 : ");
        scanf_s("%d", &genre);
        if (genre == 4) {
            break;
        }
        while (1) {
            if (genre == 1) {
                int movie_number;
                printf("무엇을 보고 싶으신가요\n 1. 엘리멘탈\n 2. 겨울왕국\n 3. 미니언즈 : ");
                scanf_s("%d", &movie_number);
                printf("\n");
                if (movie_number == 1) {
                    select_movie(time_elemental);
                    break;
                }
                else if (movie_number == 2) {
                    select_movie(time_frozen_kingdom);
                    break;
                }
                else if (movie_number == 3) {
                    select_movie(time_minions);
                    break;
                }
                else {
                    printf("다시 선택해주세요.");
                }
            }
        }
    }
}
```

```

    }
    else if (genre == 2) {
        int movie_number;
        printf("무엇을 보고 싶으신가요?\n 1. 어벤져스 엔드게임\n 2. 가디언즈 오브 갤럭시 3\n 3. 배트맨 : ");
        scanf_s("%d", &movie_number);
        printf("\n");
        if (movie_number == 1) {
            select_movie(time_avengers_endgame);
            break;
        }
        else if (movie_number == 2) {
            select_movie(time_guardians_of_galaxy);
            break;
        }
        else if (movie_number == 3) {
            select_movie(time_batman);
            break;
        }
        else {
            printf("다시 선택해주세요.");
        }
    }
    else if (genre == 3) {
        int movie_number;
        printf("무엇을 보고 싶으신가요?\n 1. 캡틴 아메리카 윈터솔저\n 2. 범죄도시2\n 3. 킹스맨 : ");
        scanf_s("%d", &movie_number);
        printf("\n");
        if (movie_number == 1) {
            select_movie(time_captain_america_winter_soldier);
            break;
        }
        else if (movie_number == 2) {
            select_movie(time_the_roundup);
            break;
        }
        else if (movie_number == 3) {
            select_movie(time_kingsman);
            break;
        }
        else {
            printf("다시 선택해주세요.");
        }
    }
}
//1번 함수 만들기 끝

```

- 적용된 배운 내용

반복문을 이용하여 기능이 계속 이어지도록 만들었음. 번호 4를 입력했을 시 반복문을 나가도록 설정

조건문을 이용하여 movie_number에 저장된 번호에 따라 다른 내용이 출력되도록 설정

2차원 배열을 이용하여 영화 상영 시간을 문자열 형태로 저장.

- movie_theater 함수에서 영화를 선택하는 기능인 select_movie 함수

```

void select_movie(char** time_movie) { //함수화 시켜서 하나로 다 사용하기. movie_theater의 세부 함수
    char(*p)[9] = time_movie; //배열 포인터에 저장 완료
    int want_time;
    int random;
    char select_time[10];
    char answer[10];
    char c;
    for (int i = 0; i < 9; i++) {
        //인덱스에 맞는 숫자부터 출력됨. 하지만 그 이후에 계속 나오고 한자가 나옴.
        //문자열에 띄어쓰기가 포함되어 있어서 그랬음. 왜 그러는지 질문
        printf("%d. %s\n", i + 1, p[i]);
    }
    printf("위의 시간표에서 시간을 골라주세요(숫자 작성) : ");
    scanf_s("%d", &want_time);
    strcpy_s(select_time, sizeof(p[want_time - 1]), p[want_time - 1]); //select_time에 p[want_time]시간 대입.
    select_movie_time(time_movie, want_time, select_time);
}

```

- 적용된 배운 내용

배열 포인터에 함수에서 매개변수로 받은 2차원 배열 time_movie를 저장.

이를 이용하여 시간표를 출력함

strcpy_s 함수를 이용하여 select_time 변수에 원하는 시간 문자열 형태로 저장.

- select_movie 함수에서 영화 시간을 고르는 기능을 하는 select_movie_time 함수

```
//1번 함수 만들기 시작
void select_movie_time(char** time_movie, int want_time, char* select_time) { //select_movie의 세부 함수
    char(*p)[9] = time_movie;
    char answer[10];
    char c;
    do {
        if (want_time <= 9 && want_time >= 1) {
            printf("%s이 선택한 시간이 맞습니까? (예, 아니오로 대답해주세요) : ", select_time);
            scanf_s("%s", answer, sizeof(answer)); //answer에 대답 저장 완료 하지만 엔터까지 answer 변수에 저장되는 듯. 이걸 해결해야 할 듯.
            while ((c = getchar()) != '\n' && c != EOF); //계행 문자 삭제
            if (strcmp(answer, "예") == 0) { //계행 문자가 포함되면서 오류 발생
                printf("%s으로 예매 완료 되었습니다. 시작 10분전에 입장해주세요.\n", select_time);
                break;
            }
            else if (strcmp(answer, "아니오") == 0) {
                printf("그렇다면 원하는 시간을 다시 골라주세요.\n");
                printf("위의 시간표에서 시간을 골라주세요(숫자 작성) : ");
                scanf_s("%d", &want_time);
                strcpy_s(select_time, sizeof(p[want_time - 1]), p[want_time - 1]); //select_time에 p[want_time]시간 대입.
                continue;
            }
            else {
                printf("답변을 다시 선택해주세요.\n");
            }
        }
    } while (answer != "예");
}
```

- 적용된 배운 내용

함수에서 매개변수로 2차원 배열을 받아 배열 포인터에 저장.

while~ 반복문을 사용하여 계행 문자를 삭제

do-while 반복문을 이용하여 시간이 맞는지를 일단 한번 무조건 실행하도록 유도 조건은 answer 문자열 변수가 "예"가 된다면 종료하도록 함.

strcmp 함수를 이용하여 answer와 "예", "아니오"가 같은 문자열 인지 검사하도록 하여 조건문에 적용.

(2) 쇼핑 기능

- 입력

1. 쇼핑할 품목에 대하여 번호 입력
2. 의류, 명품 입력 시 원하는 브랜드 입력
3. 브랜드 번호 입력 시 브랜드의 위치 정보 제공

- 출력

1. 쇼핑할 품목 출력
2. 브랜드 종류 출력
3. 브랜드 위치 출력
4. 브랜드 이름, 위치를 한 마디로 안내

- 쇼핑 기능을 수행하는 shopping 함수

```
//2번 함수 만들기 시작
void shopping() {
    int purpose_shopping;
    char market_place[9][9] = { "나이키", "스파이더", "8seconds", "MLB", "FILA", "유니클로", "스투시", "무신사", "아디다스" };
    char clothes_brand[9][9] = { "스파이더", "나이키", "아디다스", "MLB", "8seconds", "유니클로", "FILA", "무신사", "스투시" };
    char Luxury_brand[9][20] = { "CHANEL", "DIOR", "BALENCIAGA", "GUCCI", "Yves Saint Laurent", "PRADA", "Louis Vuitton", "Hermes", "Burberry" };
    char(*p)[9] = clothes_brand;
    char(*q)[20] = Luxury_brand;
    char brand_name[10] = "";
    int brand_number;
    int j;
    printf("쇼핑할 품목이 무엇인가요?\n 1. 식자재 2. 의류 3. 명품 4. 종료: ");
    scanf_s("%d", &purpose_shopping);
    while (1) {
        if (purpose_shopping == 1) {
            printf("지하 1층에 이마트가 있습니다.\n");
        }
        else if (purpose_shopping == 2) {
            printf("어떤 브랜드를 원하십니까?\n 1. 스파이더 2. 나이키 3. 아디다스 4. MLB 5. 8seconds 6. 유니클로 7. FILA 8. 무신사 9. 스투시 : ");
            scanf_s("%d", &brand_number);
            strcpy_s(brand_name, sizeof(brand_name), p[brand_number - 1]);
            for (int i = 0; i <= 8; i++) {
                printf("%d. %s", i + 1, p[i]);
                if (i != 2 || i != 5) {
                    printf("\t");
                }
                if (i == 3 || i == 6) {
                    printf("\t");
                }
                if (i == 2 || i == 5) {
                    printf("\n");
                }
            }
            printf("\n");
            printf("%s는 위의 표의 %d번에 있습니다.\n", brand_name, brand_number);
        }
    }
}
```

```

else if (purpose_shopping == 3) {
    printf("어떤 브랜드를 원하십니까?\n 1. CHANEL 2. DIOR 3. BALENCIAGA 4. GUCCI 5. Yves Saint Laurent 6. PRADA 7. Louis Vuitton 8. Hermes 9. Burberry : ");
    scanf_s("%d", &brand_number);
    strcpy_s(brand_name, sizeof(brand_name), q[brand_number - 1]);
    for (int i = 0; i <= 8; i++) {
        printf("%d. %s", i + 1, q[i]); //3,5,7번 입력시 오류
        if (i == 0) { //왜 i==1으로만 설정했을까? 되면서 i==0을 위해 설정하면 왜 i==1일때 달이 안될까..?
            printf("\t");
        }
        if (i == 1) {
            printf("\t");
        }
        if (i == 2 || i == 5) {
            printf("\n");
        }
    }
    printf("\n");
    printf("%s는 위의 표의 %d번에 있습니다.\n", brand_name, brand_number);
}
else if (purpose_shopping == 4) {
    break;
}
printf("쇼핑할 품목이 무엇인가요?\n 1. 식자재 2. 의류 3. 명품 4. 종료: ");
scanf_s("%d", &purpose_shopping);
}
}
//2번 함수 만들기 끝.

```

- 적용된 배운 내용

2차원 배열을 초기화 한 후 배열 포인터를 사용하여 2차원 배열 저장.

반복문 while을 무한루프를 통해 계속 키오스크를 사용 할 수 있게 제작 4번을 입력 시 반복문을 빠져나가게 설정.

조건문을 사용하여 purpose_shopping 변수의 값에 따라 실행 되는 문장을 다르게 설정.

문자열 함수 strcpy_s를 이용하여 brand_name 변수에 브랜드 이름 저장.

(3) 식당 기능

- 입력

1. 먹고 싶은 음식의 종류 번호 입력
2. 먹고 싶은 음식 번호 입력
3. 메뉴 확인 번호 입력

- 출력

1. 음식의 종류 출력
2. 각 카테고리에 맞는 메뉴 출력
3. 메뉴 맞게 골랐는지 확인
4. 맞을 시 가격과 메뉴 한 마디로 출력

5. 아닐 시 메뉴를 다시 고르도록 유도
6. 잘못된 답을 입력 시 다시 입력하도록 유도

- 식당 기능을 수행하는 cafeteria 함수

```
void cafeteria() {
    //구조체에 값 저장완료
    Kfood_price kfood = { {"제육볶음\0"}, {"김치찌개"}, {"동글비빔밥"} }, { 7000, 6000, 6000 };
    Jfood_price jfood = { {"등심카츠", "마제소바", "모듬초밥"} }, { 9000, 9000, 15000 };
    Cfood_price cfood = { {"짜장면", "짬뽕", "탕수육(소)"} }, { 5000, 7000, 10000 };
    int kind_of_food;
    printf("어떤 종류의 음식을 드시고 싶으신가요?\n 1. 한식 2. 일식 3. 중식 : ");
    scanf_s("%d", &kind_of_food);
    int food_number;
    int yes_or_no;
    if (kind_of_food == 1) {
        for (int i = 0; i < 3; i++) {
            printf("%d. %s - %d원 ", i + 1, kfood.food_name[i], kfood.food_price[i]);
        }
        printf("\n");
        printf("위의 메뉴에서 하나 골라주세요 : ");
        scanf_s("%d", &food_number);
        printf("고르신 메뉴가 %s(이)가 맞습니까?\t 1. 예 2. 아니오 : ", kfood.food_name[food_number - 1]);
        scanf_s("%d", &yes_or_no);
        select_korean_menu(kfood, yes_or_no, food_number);
    }
    if (kind_of_food == 2) {
        for (int i = 0; i < 3; i++) {
            printf("%d. %s - %d원 ", i + 1, jfood.food_name[i], jfood.food_price[i]);
        }
        printf("\n");
        printf("위의 메뉴에서 하나 골라주세요 : ");
        scanf_s("%d", &food_number);
        printf("고르신 메뉴가 %s(이)가 맞습니까?\t 1. 예 2. 아니오 : ", jfood.food_name[food_number - 1]);
        scanf_s("%d", &yes_or_no);
        select_japanese_menu(jfood, yes_or_no, food_number);
    }
    if (kind_of_food == 3) {
        for (int i = 0; i < 3; i++) {
            printf("%d. %s - %d원 ", i + 1, cfood.food_name[i], cfood.food_price[i]);
        }
        printf("\n");
        printf("위의 메뉴에서 하나 골라주세요 : ");
        scanf_s("%d", &food_number);
        printf("고르신 메뉴가 %s(이)가 맞습니까?\t 1. 예 2. 아니오 : ", cfood.food_name[food_number - 1]);
        scanf_s("%d", &yes_or_no);
        select_chinese_menu(cfood, yes_or_no, food_number);
    }
}
//3번 함수 만들기 끝.
```

```
//3번 함수 만들기 시작
typedef struct Korean_Food_price {
    char food_name[3][10];
    int food_price[3];
}Kfood_price;

typedef struct Japanese_Food_price {
    char food_name[3][10];
    int food_price[3];
}Jfood_price;

typedef struct Chinese_Food_price {
    char food_name[3][10];
    int food_price[3];
}Cfood_price;
```

- 적용된 배운 기능

구조체를 이용하여 각 국의 음식 이름, 가격을 저장.

반복문을 사용하여 kind_of_food 변수에 저장된 숫자에 따라 다른 기능을 실행 하도록 설정

각 국의 음식 메뉴 고르는 함수를 다 따로 만들어서 사용

- 각 국의 음식 메뉴를 고르는 기능을 수행하는 select_(국가)_menu

```
void select_korean_menu(Kfood_price food, int yes_or_no, int food_number) {
    do {
        if (yes_or_no == 1) {
            printf("%s %d원입니다.\n", food.food_name[food_number - 1], food.food_price[food_number - 1]);
            break;
        }
        else if (yes_or_no == 2) {
            printf("그렇다면 메뉴를 다시 골라주세요.\n");
            for (int i = 0; i < 3; i++) {
                printf("%d. %s - %d원 ", i + 1, food.food_name[i], food.food_price[i]);
            }
            printf("\n");
            printf("위의 메뉴에서 하나 골라주세요 : ");
            scanf_s("%d", &food_number);
            printf("고르신 메뉴가 %s(이)가 맞습니까?\t 1. 예 2. 아니오 : ", food.food_name[food_number - 1]);
            scanf_s("%d", &yes_or_no);
            continue;
        }
        else {
            printf("잘못된 입력입니다. 다시 입력해주세요.\n");
            printf("고르신 메뉴가 %s(이)가 맞습니까?\t 1. 예 2. 아니오 : ", food.food_name[food_number - 1]);
            scanf_s("%d", &yes_or_no);
        }
    } while (1);
}
```

```
void select_japanese_menu(Jfood_price food, int yes_or_no, int food_number) {
    do {
        if (yes_or_no == 1) {
            printf("%s %d원입니다.\n", food.food_name[food_number - 1], food.food_price[food_number - 1]);
            break;
        }
        else if (yes_or_no == 2) {
            printf("그렇다면 메뉴를 다시 골라주세요.\n");
            for (int i = 0; i < 3; i++) {
                printf("%d. %s - %d원 ", i + 1, food.food_name[i], food.food_price[i]);
            }
            printf("\n");
            printf("위의 메뉴에서 하나 골라주세요 : ");
            scanf_s("%d", &food_number);
            printf("고르신 메뉴가 %s(이)가 맞습니까?\t 1. 예 2. 아니오 : ", food.food_name[food_number - 1]);
            scanf_s("%d", &yes_or_no);
            continue;
        }
        else {
            printf("잘못된 입력입니다. 다시 입력해주세요.\n");
            printf("고르신 메뉴가 %s(이)가 맞습니까?\t 1. 예 2. 아니오 : ", food.food_name[food_number - 1]);
            scanf_s("%d", &yes_or_no);
        }
    } while (1);
}
```

```

void select_chinese_menu(cfood_price food, int yes_or_no, int food_number) {
    do {
        if (yes_or_no == 1) {
            printf("%s %d원입니다.\n", food.food_name[food_number - 1], food.food_price[food_number - 1]);
            break;
        }
        else if (yes_or_no == 2) {
            printf("그렇다면 메뉴를 다시 골라주세요.\n");
            for (int i = 0; i < 3; i++) {
                printf("%d. %s - %d원 ", i + 1, food.food_name[i], food.food_price[i]);
            }
            printf("\n");
            printf("위의 메뉴에서 하나 골라주세요 : ");
            scanf_s("%d", &food_number);
            printf("고르신 메뉴가 %s(이)가 맞습니까?\t1. 예 2. 아니오 : ", food.food_name[food_number - 1]);
            scanf_s("%d", &yes_or_no);
            continue;
        }
        else {
            printf("잘못된 입력입니다. 다시 입력해주세요.\n");
            printf("고르신 메뉴가 %s(이)가 맞습니까?\t1. 예 2. 아니오 : ", food.food_name[food_number - 1]);
            scanf_s("%d", &yes_or_no);
        }
    } while (1);
}

```

- 적용된 배운 내용

구조체를 매개변수로 받아 함수 내에서 사용

do-while문을 이용하여 반복문 진행 조건 없이 무한루프를 사용하여 반복 yes_or_no 변수 값이 1이 됐을 때 반복문 탈출

- 운동 기능을 수행하는 exercise 함수

```

void exercise() {
    Exerkind exercise_kind;
    initialize(&exercise_kind);
    int place_number;
    char c;
    char* place_name = (char*)malloc(sizeof(char) * strlen(exercise_kind.exercise_place[2]));
    //각 장소의 주소 저장 완료
    for (int i = 0; i < 3; i++) {
        exercise_kind.place[i] = &exercise_kind.exercise_place[i];
    }
    printf("하고자 하는 운동이 무엇인가요? : ");
    scanf_s("%s", place_name, sizeof(place_name));
    while ((c = getchar()) != '\n' && c != EOF);
    if (strcmp(place_name, "헬스") == 0) {
        printf("%s은 %x위치에 있습니다.", exercise_kind.exercise_place[0], exercise_kind.place[0]);
    }
    if (strcmp(place_name, "축구") == 0) {
        printf("%s은 %x위치에 있습니다.", exercise_kind.exercise_place[1], exercise_kind.place[1]);
    }
    if (strcmp(place_name, "테니스") == 0) {
        printf("%s은 %x위치에 있습니다.", exercise_kind.exercise_place[2], exercise_kind.place[2]);
    }
    free(place_name);
}

```

```
//4번 함수 만들기 시작
typedef struct exercise_kind {
    char type[3][10];
    char exercise_place[3][10];
    char* place[3];
}Exerkind;
```

- 적용된 배운 기능

운동의 종류, 운동 하는 장소, 그 장소의 주소를 저장하는 변수를 하나의 구조체 exercise_kind로 만들. typedef를 이용하여 Exerkind라고 새로운 별칭 지정

Initialize 함수를 이용하여 구조체 초기화

place_name 문자열 변수를 malloc을 이용하여 동적으로 메모리 할당.

while 반복문을 사용하여 계행 문자 삭제

strcmp 함수를 사용하여 place_name 변수와 운동의 종류 이름과 맞는지 비교하여 각각에 맞는 기능 실행.

free()를 통해 동적으로 할당한 메모리 삭제

- 구조체 멤버를 초기화 해주는 initialize 함수

```
//구조체 초기화 함수
void initialize(Exerkind* exercise_kind) {
    const char* exerciseType[3] = { "헬스", "축구", "테니스" };
    const char* exercisePlace[3] = { "헬스장", "축구", "테니스" };
    for (int i = 0; i < 3; i++) {
        strcpy_s(exercise_kind->type[i], sizeof(exercise_kind->type[i]), exerciseType[i]);
        strcpy_s(exercise_kind->exercise_place[i], sizeof(exercise_kind->exercise_place[i]), exercisePlace[i]);
        exercise_kind->type[i][9] = "\0";
        exercise_kind->exercise_place[i][9] = "\0";
    }
}
```

- 적용된 배운 내용

매개변수로 구조체를 포인터로 받아 사용함.

exerciseType 2차원 배열에 운동 종류로 문자열 초기화

exercisePlace 2차원 배열에 운동 하는 장소로 문자열 초기화

for문 안에서 strcpy_s 함수를 이용하여 exercise_kind 안의 멤버 초기화.

또한 맨 마지막에 널 문자 추가.

2) 테스트 결과

(1) 영화 예매 기능

- 보고 싶은 영화의 장르를 입력하면 그에 맞는 영화 제목 출력 후 영화 선택 시 그 영화의 시간표 출력 후 원하는 시간 번호 입력 시 제대로 골랐는지 확인 후 '예'가 나올 경우 예매 완료

- 테스트 결과 스크린샷

```
보고싶은 장르를 골라주세요
1. 애니메이션
2. 히어로
3. 액션
4. 종료 : 1
무엇을 보고 싶으신가요?
1. 엘리멘탈
2. 겨울왕국3
3. 미니언즈 : 1

1. 10시20분
2. 11시00분
3. 11시40분
4. 12시40분
5. 13시30분
6. 14시20분
7. 15시20분
8. 16시10분
9. 17시00분
위의 시간표에서 시간을 골라주세요(숫자 작성) : 1
10시20분이 선택한 시간이 맞습니까? (예, 아니오로 대답해주세요) : 1
답변을 다시 선택해주세요.
10시20분이 선택한 시간이 맞습니까? (예, 아니오로 대답해주세요) : dksldh
답변을 다시 선택해주세요.
10시20분이 선택한 시간이 맞습니까? (예, 아니오로 대답해주세요) : 아니오
그렇다면 원하는 시간을 다시 골라주세요.
위의 시간표에서 시간을 골라주세요(숫자 작성) : 2
11시00분이 선택한 시간이 맞습니까? (예, 아니오로 대답해주세요) : 예
11시00분으로 예매 완료 되었습니다. 시작 10분전에 입장해주세요.
```

(2) 쇼핑 기능

- 쇼핑하고자 하는 품목을 선택하면 그에 맞는 기능들이 시작 됨. 의류, 명품의 경우는 브랜드를 선택하고 이후 그 브랜드의 위치를 알려줌. 4번을 누르면 종료

- 테스트 결과 스크린샷

1. 식자재 2. 의류 3. 명품 4. 종료 : 1
지하 1층에 이마트가 있습니다.

쇼핑할 품목이 무엇인가요?
1. 식자재 2. 의류 3. 명품 4. 종료 : 2
어떤 브랜드를 원하십니까?
1. 스파이더 2. 나이키 3. 아디다스 4. MLB 5. 8seconds 6. 유니클로 7. FILA 8. 무신사 9. 스투시 : 3
1. 스파이더 2. 나이키 3. 아디다스
4. MLB 5. 8seconds 6. 유니클로
7. FILA 8. 무신사 9. 스투시
아디다스는 위의 표의 3번에 있습니다.

쇼핑할 품목이 무엇인가요?
1. 식자재 2. 의류 3. 명품 4. 종료 : 3
어떤 브랜드를 원하십니까?
1. CHANEL 2. DIOR 3. BALENCIAGA 4. GUCCI 5. Yves Saint Laurent 6. PRADA 7. Louis Vuitton 8. Hermes 9. Burberry : 4
1. CHANEL 2. DIOR 3. BALENCIAGA
4. GUCCI 5. Yves Saint Laurent 6. PRADA
7. Louis Vuitton 8. Hermes 9. Burberry
GUCCI는 위의 표의 4번에 있습니다.

쇼핑할 품목이 무엇인가요?
1. 식자재 2. 의류 3. 명품 4. 종료 : 4

(3) 식당 기능

- 원하는 종류의 음식을 고르면 그에 맞는 메뉴가 출력됨. 메뉴를 고르면 고른 메뉴가 맞는지 확인하고 1번을 입력하면 메뉴와 가격을 출력, 2번을 입력하면 다시 메뉴를 고르게 유도, 그 이외의 답을 입력하면 메뉴가 맞는지 다시 입력하게 함.

-테스트 결과 스크린샷

어떤 종류의 음식을 드시고 싶으신가요?
1. 한식 2. 일식 3. 중식 : 1
1. 제육볶음 - 7000원 2. 김치찌개 - 6000원 3. 돌솥비빔밥 - 6000원
위의 메뉴에서 하나 골라주세요 : 1
고르신 메뉴가 제육볶음(이)가 맞습니까? 1. 예 2. 아니오 : 3
잘못된 입력입니다. 다시 입력해주세요.
고르신 메뉴가 제육볶음(이)가 맞습니까? 1. 예 2. 아니오 : 2
그렇다면 메뉴를 다시 골라주세요.
1. 제육볶음 - 7000원 2. 김치찌개 - 6000원 3. 돌솥비빔밥 - 6000원
위의 메뉴에서 하나 골라주세요 : 2
고르신 메뉴가 김치찌개(이)가 맞습니까? 1. 예 2. 아니오 : 1
김치찌개 6000원입니다.

(4) 운동 기능

- 원하는 운동의 종류를 입력 시 운동을 할 수 있는 장소의 주소를 알려줌.

하고자 하는 운동이 무엇인가요? : 헬스
헬스장은 348ff88e 위치에 있습니다.

4. 계획 대비 변경 사항

1) 기능의 2, 3의 이름 변경

- 이전 : 기능 2 : 주차 기능 3 : 일행 확인
- 이후 : 기능 2 : 일행 확인 3 : 주차 기능
- 사유 : 일행 확인하는 기능이 더욱 중요해 보여서 순서를 바꿨다.

2) 기능 구현의 시작 날짜 변경

- 이전 : 주차 기능의 시작 날짜 : 11/24, 일행 확인 기능의 시작 11/3
- 이후 : 주차 기능의 시작 날짜 : 12/10, 일행 확인 기능의 시작 12/10
- 사유 : 기능 1의 양이 너무 방대하고 사용되는 함수들도 많아서 제작하는데 너무
오래 걸렸음.

5. 프로젝트 일정

(진행한 작업과 진행 중인 작업 등을 표기)

업무		11/3	11/10	11/17	12/10	12/24
제안서 작성		완료				
기능1	세부기능1		완료			
기능2	세부기능				----->	
기능3	세부기능1				----->	